

Monitoring Kualitas Air Tambak Udang Menggunakan Metode *Data Logging*

<http://dx.doi.org/10.28932/jutisi.v6i1.2365>

Harry Pratama Ramadhan^{✉ #1}, Condro Kartiko^{*2}, Agi Prasetiadi^{#3}

[#]Jurusan SI Teknik Informatika, Institut Teknologi Telkom Purwokerto

Jl. DI Panjaitan No.128, Pancurawis, Purwokerto Kidul, Kec. Purwokerto Sel., Kabupaten Banyumas, Jawa Tengah

¹16102052@ittelkom-pwt.ac.id

³agi@ittelkom-pwt.ac.id

^{*}Jurusan SI Rekayasa Perangkat Lunak, Institut Teknologi Telkom Purwokerto

Jl. DI Panjaitan No.128, Pancurawis, Purwokerto Kidul, Kec. Purwokerto Sel., Kabupaten Banyumas, Jawa Tengah

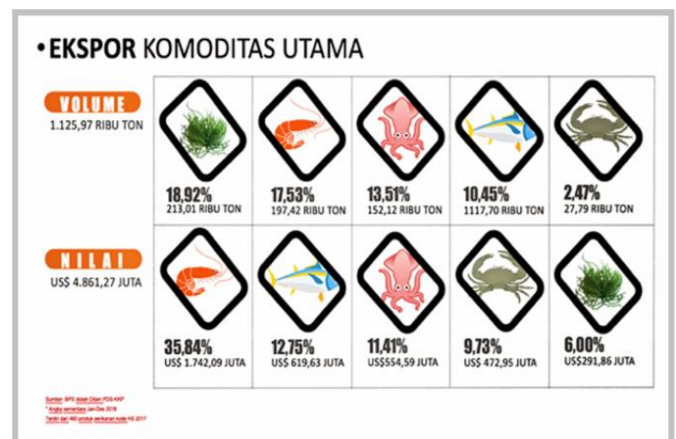
²condro.kartiko@ittelkom-pwt.ac.id

Abstract — Based on the prior study, some shrimp ponds went bankrupt due to pond water quality monitoring is still not good. Many shrimps get sick and die for water quality monitoring still relies on laboratory checks and is rarely done because of financial problems. The purpose of this study is to develop a monitoring system of shrimp pond water quality especially for vannamei shrimp using an Internet of Things (IoT)-based device with a data logging method. The system role is to monitor the water condition, record sensor data, and provide water quality status of shrimp ponds based on water movement, turbidity of water, and water temperature. The data logger device uses a microcontroller named NodeMCU ESP8266 and two sensors namely the LDR sensor and the water temperature sensor dallas 18b20. The devices are connected to the internet and send all water quality monitoring data to Google's database service called Firebase. The results of the water quality monitoring can be accessed through an Android-based monitoring application that is built using Flutter framework which contains information.

Keywords— Flutter Android; Internet of Things; Monitoring System; Water Quality

I. PENDAHULUAN

Kegiatan budi daya tambak udang merupakan sebuah peluang usaha yang memiliki potensi yaitu keuntungan yang sangat besar. Hal tersebut didukung dari data kenaikan jumlah ekspor komoditas udang yang semakin meningkat setiap tahunnya, berdasarkan data dari laporan tahunan milik Kementerian Kelautan dan Perikanan Republik Indonesia tahun 2018, komoditas udang secara volume menempati posisi kedua dari komoditas ekspor utama produk perikanan yaitu mencapai angka 197,42 ribu ton dan untuk nilai yang didapatkan dari hasil ekspor keseluruhan komoditas, udang menempati posisi pertama dengan jumlah sebanyak 1.742,09 juta US Dollar. Hal tersebut ditunjukkan dalam Gambar 1.



Gambar 1 Nilai dan Volume Ekspor Produk Perikanan dan Kelautan Tahun 2018 [1]

Namun, banyak para pengusaha budi daya tambak udang yang mengalami kerugian hingga bangkrut dikarenakan udang merupakan komoditas yang mudah terserang penyakit. Virus yang kerap menyerang udang yaitu *white feces* (WFD) dan *white spot virus* (WSSV) dimana virus tersebut dapat mengurangi tingkat ketahanan hidup udang hingga mencapai 30% [2]. Adapun parameter yang digunakan untuk mengukur kualitas air yaitu pH, suhu, kecerahan, dan organisme [3].

A. Rumusan Masalah

Rumusan masalah pada penelitian ini berdasarkan latar belakang permasalahan yaitu :

Bagaimana tingkat keberhasilan implementasi metode *data logging* pada penelitian ini dapat meningkatkan produksi komoditas udang di Indonesia ?

B. Tujuan

Membuat sistem pemantauan untuk menganalisis kualitas air tambak udang secara *real-time* dengan

metode *data logging* menggunakan NodeMCU yang berperan sebagai *data logger*. Diharapkan dengan adanya alat *monitoring* kualitas air tersebut, hasil produksi komoditas udang selanjutnya semakin meningkat.

C. Batasan Masalah

Adapun batasan masalah dari penelitian ini yaitu :

1. Sistem *Monitoring* hanya dapat dilakukan pada siang hari, karena sensor LDR memanfaatkan sinar matahari.
2. Sistem ini hanya dapat berjalan ketika cuaca tidak hujan, karena perangkat *data logger* merupakan perangkat elektronik yang mudah rusak apabila terkena air.
3. Sistem deteksi pergerakan air sangat tergantung pada tingkat kekeruhan air, karena tingkat kekeruhan air memiliki nilai refraksi cahaya matahari yang berbeda.

Penelitian ini berkorelasi dengan penelitian sebelumnya yaitu penelitian milik Apri Junaidi dan Condro Kartiko tahun 2020[4], sistem *monitoring* kualitas air tambak udang dirancang dalam bentuk perangkat berbasis IoT yang dapat digunakan untuk menganalisis data kualitas air. *Data logging* merupakan metode yang digunakan untuk mendapatkan data dari sensor secara *real time*[5]. Perangkat *data logger* menggunakan mikrokontroler NodeMCU ESP8266, sensor suhu air DS18B20, dan sensor LDR. Semua data sensor akan dicatat dan diproses oleh NodeMCU ESP8266 dan hasil pemrosesan dan pencatatan data akan dikirim ke Firebase, kemudian aplikasi Android akan mengakses data yang terdapat pada Firebase untuk menunjukkan informasi kualitas air tambak udang.

II. LANDASAN TEORI

A. Kualitas Air

Kualitas air merupakan sebuah parameter yang digunakan untuk mengukur serta mengetahui bagaimana kondisi air dari sebuah tambak udang dengan cara melakukan pengujian. Beberapa parameter yang digunakan untuk menguji kualitas air yaitu [3]:

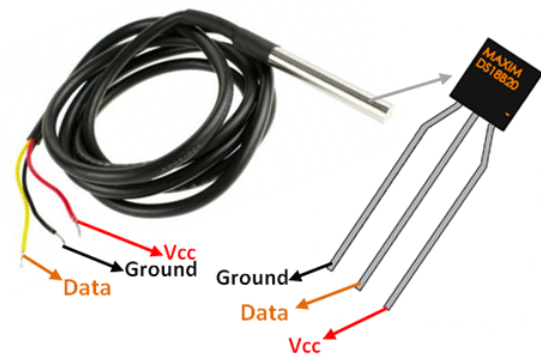
1. Tingkat Keasaman : pH merupakan tingkat keasaman air, pengujian pH pada air sangat menentukan apakah air tersebut memiliki kualitas air yang baik atau buruk.
2. Suhu : suhu merupakan parameter kualitas air yang dapat mempengaruhi kehidupan organisme di dalamnya. Faktor yang mempengaruhi perubahan suhu air yaitu musim, ketinggian wilayah, perairan dan permukaan air laut, kedalaman serta aliran air. Suhu air yang ideal bagi kehidupan organisme adalah suhu yang perubahannya tidak signifikan antara siang dan malam hari.

3. Kecerahan : kecerahan merupakan parameter yang sangat membantu proses fotosintesis dari organisme yang hidup di perairan. Air yang baik pada umumnya tidak terlalu cerah dan tidak terlalu keruh.
4. Organisme : organisme merupakan faktor biologis yang mempengaruhi kualitas air. Pertumbuhan plankton membantu perkembangbiakan ikan dan udang.

Pada penelitian ini digunakan dua parameter pengujian kualitas air tambak udang yaitu menggunakan parameter suhu dan kecerahan air, dimana untuk kecerahan air dibantu menggunakan sensor LDR untuk menentukan gerakan udang dan kekeruhan air dengan memanfaatkan refraksi sinar matahari di dalam air, dan untuk suhu menggunakan sensor suhu air DS1820 untuk mengetahui kondisi suhu air tambak. Suhu air tambak yang baik berkisar antara 20° C sampai 31° C [6].

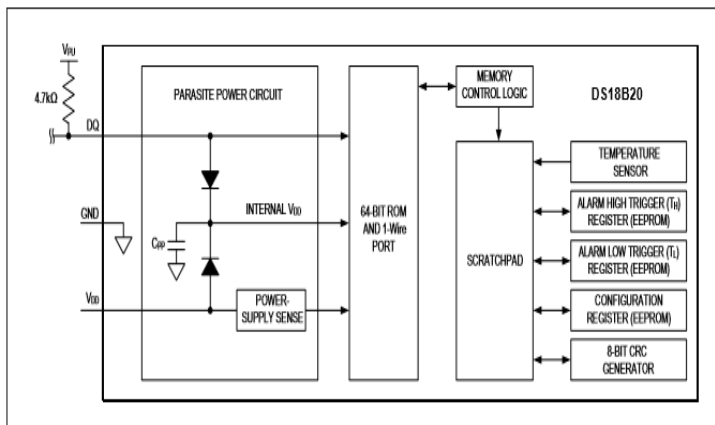
B. Sensor Suhu (Dallas DS1820)

Sensor Suhu (Dallas DS18B20) adalah sebuah sensor suhu digital dari Maxim IC (Dulu bernama Dallas Semiconductor, lalu diakuisi oleh Maxim Integrated Products). Sensor Dallas DS18B20 adalah sensor yang mampu membaca suhu dengan tingkat ketelitian 9 hingga 12 bit, rentang suhu yang dapat dibaca oleh sensor tersebut yaitu antara -55°C hingga 125°C dengan akurasi sebesar kurang lebih 0.5°C [7]. Adapun untuk bentuk sensor DS1820 yaitu sensor tersebut memiliki kabel sepanjang kurang lebih satu meter yang didalamnya terdapat tiga kabel kecil berwarna merah, kuning, dan hitam (Gambar 2). Fungsi masing-masing kabel yaitu merah untuk tegangan positif, kuning untuk keluaran nilai suhu sensor yang nantinya akan dibaca oleh NodeMCU, dan kabel hitam untuk tegangan negatif.



Gambar 2. Bentuk Sensor DS18B20 [8]

Sensor DS18B20 memiliki blok diagram (Gambar 3) dimana sensor tersebut di dalamnya memiliki beberapa komponen.



Gambar 3. Blok Diagram DS18B20 [9]

C. NODEMCU ESP8266

NodeMCU ESP8266 merupakan *development board* ESP-12E yang didukung oleh chip ESP8266. Pada penelitian ini, NodeMCU ESP8266 yang digunakan memiliki mikroprosesor RISC LX106 RXC Tensilica 32 bit yang dapat bekerja pada kecepatan *clock* 80 sampai 160 MHz. Memiliki RAM sebesar 128 KB dan *flash memory* 4 MB dan konektivitas *wifi transceiver* dengan standar 802.11 b/g/n yang beroperasi pada frekuensi 2.4 GHz dan tegangan operasi sebesar 5 volt, NodeMCU ESP8266 cocok digunakan untuk proyek berbasis IoT[10]. NodeMCU ESP8266 memiliki beberapa model dan versi, pada penelitian ini digunakan versi 1.0 (Gambar 4) dari vendor LoLin yang memiliki pin GPIO sebanyak 13 pin dan pin ADC sebanyak 1 pin yang digunakan untuk memasukkan data dari sensor.



Gambar 4. Bentuk Mikrokontroler NodeMCU ESP8266 Versi 1.0

Dalam proses penulisan program ke NodeMCU ESP8266, diperlukan beberapa *library* pendukung supaya program dapat berjalan dengan baik, karena proses pengolahan data sensor sepenuhnya berada pada NodeMCU ESP8266. Tabel I merupakan rincian dari penggunaan *library* pada NodeMCU ESP8266.

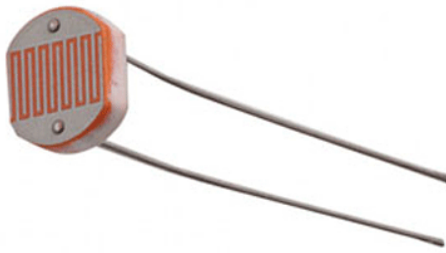
TABEL I
LIBRARY PENDUKUNG PROGRAM NODEMCU ESP8266

Library	Fungsi
iostream	Singkatan dari <i>input output stream</i> , header yang digunakan untuk memanggil fungsi standar dari pemrograman bahasa C
FirestoreESP8266.h	<i>Library</i> yang berfungsi untuk menghubungkan NodeMCU ESP8266 ke <i>platform</i> firebase
ESP8266WiFi.h	<i>Library</i> yang berfungsi untuk mengaktifkan fungsi <i>wifi transceiver</i> pada NodeMCU ESP8266
ESP8266HTTPClient.h	<i>Library</i> yang berfungsi untuk melakukan GET maupun POST data ke <i>http web server</i>
SimpleTimer.h	<i>Library</i> yang berfungsi untuk mengatur interval waktu, menggantikan fungsi <i>delay</i> pada pemrograman di Arduino IDE
DallasTemperature.h	<i>Library</i> yang berfungsi untuk mengakses data dari sensor suhu digital DS18B20
OneWire.h	<i>Library</i> yang berfungsi untuk mengakses data dari sensor suhu digital DS18B20
ArduinoJson.h	<i>Library</i> yang berfungsi untuk mengubah kumpulan data yang akan dikirimkan ke firebase ke dalam format JSON
NTPClient.h	<i>Library</i> yang berfungsi untuk mendapatkan waktu dari NTP server
WiFiUdp.h	<i>Library</i> yang berfungsi untuk melakukan komunikasi UDP

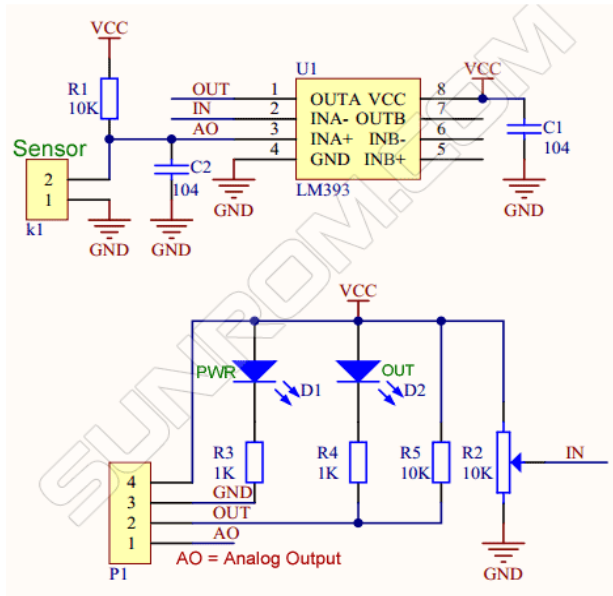
D. Sensor Cahaya (Light Dependent Resistor)

Sensor cahaya merupakan sensor yang komponennya berbentuk resistor yang nilai resistansinya berubah-ubah sesuai dengan intensitas cahaya yang mengenai sensor tersebut[11]. Sensor LDR menggunakan data analog dengan nilai maksimum yang dapat diterima sensor LDR adalah 1023, data disimpan menggunakan variabel *resistance* yang artinya cahaya yang mengenai sensor LDR mempengaruhi nilai resistensi yang terbaca oleh mikrokontroler Sensor LDR memiliki bentuk seperti lampu LED dan memiliki dua kaki (Gambar 5). Sensor LDR tidak memancarkan cahaya, melainkan menerima cahaya yang mengenai sensor LDR. Sensor LDR yang digunakan pada penelitian ini

menggunakan modul yang memiliki pin sebanyak 4 pin dan memiliki IC bernama LM393 (Gambar 6).



Gambar 5. Bentuk Sensor LDR[12]



Gambar 6. Skematik sensor cahaya [11]

Rumus 1 merupakan perhitungan nilai dari cahaya yang masuk berdasarkan kondisi refraksi cahaya di dalam air.

$$resistance = \frac{read_adc(1) \times 100}{1023} \quad (1)$$

Keterangan :

1. *Resistance*: variabel nilai resistensi
2. *Read_adc* : proses sensor membaca nilai pada sensor secara analog
3. *x100* : mengalikan nilai yang terbaca oleh sensor dengan nilai 100
4. *1023* : membagi hasil pembacaan sensor yang dikalikan dengan nilai 100 untuk mendapatkan rentang nilai 0 sampai 100

Kalibrasi sensor cahaya diperlukan untuk mendapatkan standar deviasi yang akurat untuk memberikan status kondisi kualitas air tambak udang. Rumus 2 merupakan *array* untuk mendapatkan nilai dari sensor cahaya.

$$s_arr[30] = \{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0\}; \quad (2)$$

Kemudian digunakan metode shifting untuk menggeser nilai *array* kekanan dengan data yang baru, geser ke kanan sama dengan perulangan 0 kurang / sama dengan 28 untuk *array* panjang *array* 30 elemen. Nilai di *array* 28 akan diganti dan data yang ada di *array* sebelumnya digeser ke kanan, dan untuk nilai *array* di paling kiri diisi oleh data baru.

Kemudian rumus 3 merupakan rumus *variance* untuk menghitung data yaitu :

$$S^2 = \frac{n \sum_{i=1}^n x_i^2 - (\sum_{i=1}^n x_i)^2}{n(n-1)} \quad (3)$$

Dan rumus standar deviasi untuk menentukan pergerakan air, apabila gerakan air pasif maka nilai akan mendekati 0 dan apabila aktif akan memiliki nilai yang bervariasi. Rumus 4 merupakan rumus untuk standar deviasi.

$$S = \sqrt{\frac{n \sum_{i=1}^n x_i^2 - (\sum_{i=1}^n x_i)^2}{n(n-1)}} \quad (4)$$

Keterangan:

1. x_i = nilai *x* ke-*i*
2. \bar{x} = rata-rata
3. *n* = ukuran sampel

E. Firebase

Firebase merupakan layanan dari Google yang digunakan untuk mempermudah para *developer* perangkat lunak untuk mengembangkan aplikasinya. Berfungsi sebagai BaaS (*Backend as a Service*) merupakan solusi yang ditawarkan oleh Google agar *developer* dapat fokus mengembangkan keseluruhan aplikasi [13].

F. Flutter

Flutter merupakan standard development kit (SDK) yang dibuat oleh Google untuk pengembangan aplikasi *mobile* yang dapat dipublikasi di *platform* Android maupun IOS. Flutter menggunakan bahasa pemrograman *.dart* dan dapat berjalan pada sistem operasi Android versi 4.1 dan IOS versi 8 atau yang lebih tinggi. Flutter memiliki mesin render bawaan untuk menampilkan *widget*, sehingga UI yang ditampilkan akan terlihat lebih konsisten dan unik. Pada Android, aplikasi dikompilasi dengan mesin C / C++ dengan menggunakan Android NDK, kerangka kode dibuat *native* dan dikompilasi kembali menggunakan *dart compiler*. Pada IOS, kode dikompilasi dengan LLVM dan dijalankan dengan kumpulan instruksi *native* tanpa interpreter [14].

G. Android

Android merupakan sistem operasi yang dikembangkan oleh Android Inc, Google yaitu sistem operasi gratis yang dapat dikembangkan oleh siapa saja (*open source*). Basis

kernel yang terdapat pada Android dapat digunakan untuk perangkat layar sentuh seperti *smartphone* / tablet. Penamaan sistem operasi Android menggunakan alfabet yang berurutan, untuk saat ini versi Android sudah mencapai versi Android Q / Android 10 [15].

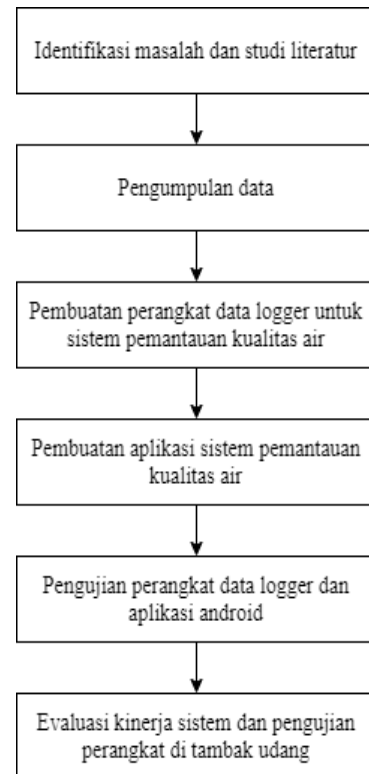
Android dikembangkan menggunakan sistem operasi yang memiliki komponen arsitektur. Adapun komponen arsitektur yang terdapat pada sistem operasi Android ditampilkan pada gambar 7.



Gambar 7. Arsitektur Android [16]

III. METODOLOGI PENELITIAN

Pada penelitian ini, terdapat enam tahapan penelitian seperti pada Gambar 8. Tahap pertama penelitian yaitu dilakukan sebuah identifikasi masalah dan studi literatur. Identifikasi masalah dilakukan dengan cara melakukan survey di tempat tambak udang dan mengamati perilaku pengelola tambak udang dalam mengelola tambak udang, dan untuk studi literatur yaitu membaca beberapa jurnal dan pustaka *online*. Tahap kedua penelitian yaitu dilakukan pengumpulan data dengan melakukan wawancara kepada pengelola tambak udang untuk mendapatkan informasi yang dibutuhkan dalam penelitian serta melakukan observasi ketempat tambak udang yang akan dijadikan obyek penelitian. Kemudian di tahap ketiga merupakan tahap pembuatan sistem *monitoring* berbasis IoT yaitu berupa alat *data logger*. Tahap keempat yaitu pembuatan aplikasi Android yang digunakan untuk menampilkan *dashboard* informasi pemantauan kualitas air. Tahap kelima dilakukan pengujian terhadap alat yang sudah dibuat. Tahap keenam dilakukan evaluasi kinerja alat dan mencoba melakukan pemantauan kualitas air secara *real time*.



Gambar 8. Metodologi Penelitian

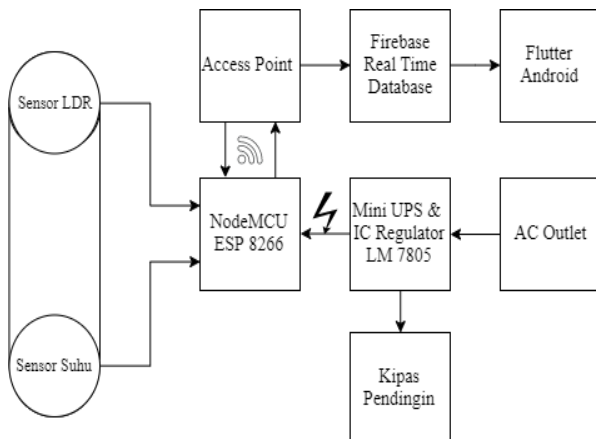
IV. DESAIN SISTEM

Pada bagian ini, dijelaskan mengenai perancangan dan realisasi perangkat *data logger* dan perangkat lunak di Android untuk memantau dan mencatat kualitas air tambak udang. Untuk perancangan perangkat *data logger* meliputi *wiring diagram*. Sedangkan untuk perangkat lunak meliputi diagram alir program pada NodeMCU ESP8266 dan perangkat lunak Android meliputi antarmuka pada aplikasi serta diagram alir pada aplikasi Android.

A. Diagram Blok Sistem

Cara kerja dari perangkat *monitoring* kualitas air tambak udang berdasarkan diagram blok sistem (Gambar 9) yaitu perangkat *data logger* memiliki dua sensor yaitu sensor suhu dan sensor LDR, masing-masing sensor memiliki fungsi yaitu sensor LDR memiliki fungsi mendeteksi pergerakan dan kekeruhan air dengan memanfaatkan cahaya matahari, dan untuk sensor suhu berfungsi untuk membaca suhu air tambak yang baik untuk tambak udang. Kedua sensor tersebut dihubungkan ke mikrokontroler NodeMCU ESP8266 dan data yang didapatkan dari kedua sensor akan diolah menggunakan beberapa rumus seperti *array*, *variance*, *standard deviation*, dan *average / mean*. Setelah data berhasil diolah, di dalam NodeMCU ESP8266 terdapat fungsi yang didalamnya terdapat percabangan untuk menentukan kondisi air berdasarkan nilai yang didapatkan dari sensor. Selanjutnya NodeMCU ESP266 akan mengirim

semua data sensor tersebut melalui jaringan internet ke Firebase *database*, dan data dapat diakses pada *smartphone* Android.



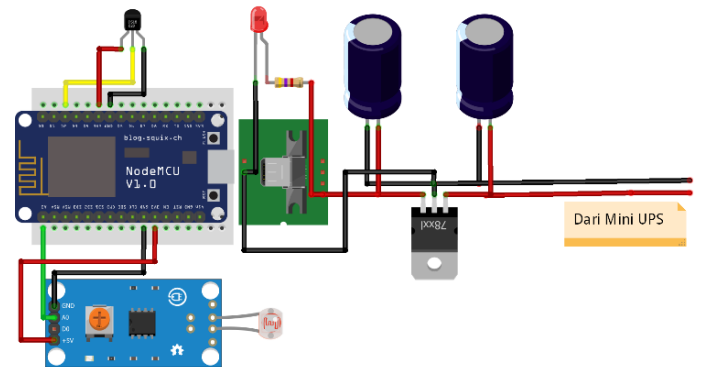
Gambar 9. Diagram Blok Sistem

B. Perancangan Perangkat Keras Data Logger

Perancangan perangkat *data logger* yaitu berisi tentang penjelasan tentang *wiring diagram* pada rangkaian data logger. Perangkat *data logger* dirancang untuk memudahkan pengujian saat berada di lokasi tambak udang dan pengambilan data kualitas air. *Wiring diagram* (Gambar 10) menunjukkan beberapa komponen yang digunakan pada penelitian ini serta pin apa saja yang terhubung antar komponen yang ada.

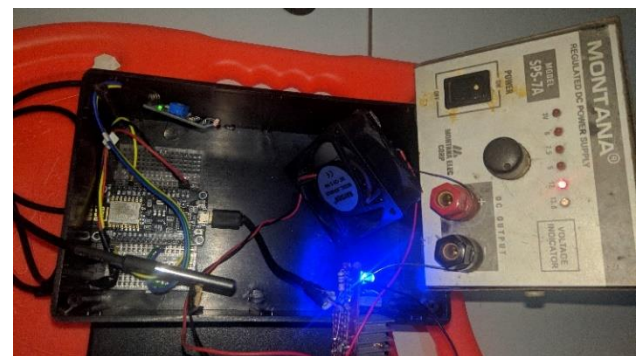
Pada penelitian ini terdapat beberapa komponen yang digunakan yaitu sensor LDR, sensor suhu air ds18b20, *access point*, mini ups dan IC regulator LM 7805 serta kipas pendingin. Beberapa komponen lainnya seperti PCB, LED indikator, saklar, dan *cord AC outlet* disematkan di dalam perangkat keras data logger. Mini ups adalah sebuah alat yang digunakan untuk menghidupkan perangkat *data logger*. Berperan sebagai baterai, mini ups memiliki bentuk yang kecil dan dapat dimasukkan kedalam kotak perangkat *data logger*, sehingga perangkat *data logger* akan lebih ringan apabila dibandingkan dengan baterai 12 volt lainnya seperti aki motor. Mini ups sudah dilengkapi dengan *cord AC outlet*, dimana ketika mengisi ulang daya baterai bisa dilakukan secara langsung dengan menghubungkan mini ups ke listrik PLN. Output daya sebesar 12 volt yang dihasilkan dari mini ups akan dibuat dua jalur, jalur pertama untuk kipas pendingin yang akan langsung menerima daya 12 volt, sedangkan jalur kedua akan masuk ke rangkaian penurun tegangan dari 12 volt menjadi 5 volt dengan IC regulator LM7805. Rangkaian penurun tegangan memiliki beberapa komponen yaitu diantaranya dua buah elco (kapasitor) DC, dimana elco pertama berukuran 25 volt 1000 mikrofarad yang artinya elco tersebut mampu menyimpan tegangan maksimal 25 volt 1000 mikrofarad, elco pertama digunakan sebagai penstabil tegangan 12 volt dari mini ups dan elco kedua berukuran 25 volt 470 mikrofarad yang artinya elco

tersebut mampu menerima tegangan maksimal 25 volt dengan daya simpan sebesar 470 mikrofarad yang digunakan sebagai penstabil tegangan keluaran dari hasil penurunan tegangan IC regulator LM7805 untuk mensuplai NodeMCU dan lampu indikator rangkaian penurun tegangan. Untuk lampu indikator digunakan resistor sebesar 1.6 kilo ohm untuk menurunkan tegangan keluaran dari mini ups yaitu dari 5 volt ke 3 volt. Tujuan dari penurunan yaitu supaya daya yang di suplai ke NodeMCU tidak berlebihan dan tetap stabil, karena NodeMCU hanya mampu menerima daya sebesar 5 volt.



Gambar 10. Wiring Perangkat Data Logger

fritzing



Gambar 11. Realisasi Rangkaian Data Logger

Gambar 11 merupakan realisasi dari rangkaian perangkat *data logger*. Dikarenakan perangkat masih dalam tahap pengembangan, maka mini ups digantikan dengan *power supply* dengan output tegangan yang sama yaitu 12 volt. Perangkat *data logger* dapat bekerja dengan baik. Tabel II dibawah merupakan rincian dari penggunaan pin pada NodeMCU.

TABEL II
PIN KOMPONEN

Pin NodeMCU	Komponen
3.3 V	Sumber Tegangan Positif
GND	Sumber Tegangan Negatif
A0	Pin Data Sensor LDR
D2	Pin Data Sensor DS18B20

C. Perancangan Perangkat Lunak Data Logger

Pada bagian ini akan dijelaskan perancangan perangkat lunak pada perangkat data logger (Gambar 14). Ketika NodeMCU ESP8266 mendapatkan tegangan dari mini ups, NodeMCU ESP8266 akan mengaktifkan sensor ds18b20 dan sensor LDR sehingga mendapatkan data nilai dari kedua sensor tersebut. Kemudian NodeMCU ESP8266 akan melakukan perhitungan dan kalibrasi menggunakan rumus dari *standard deviation*, *variance*, dan *average / mean*. Rumus perhitungan tersebut di modifikasi menjadi fungsi dalam bahasa pemrograman C pada Arduino IDE.

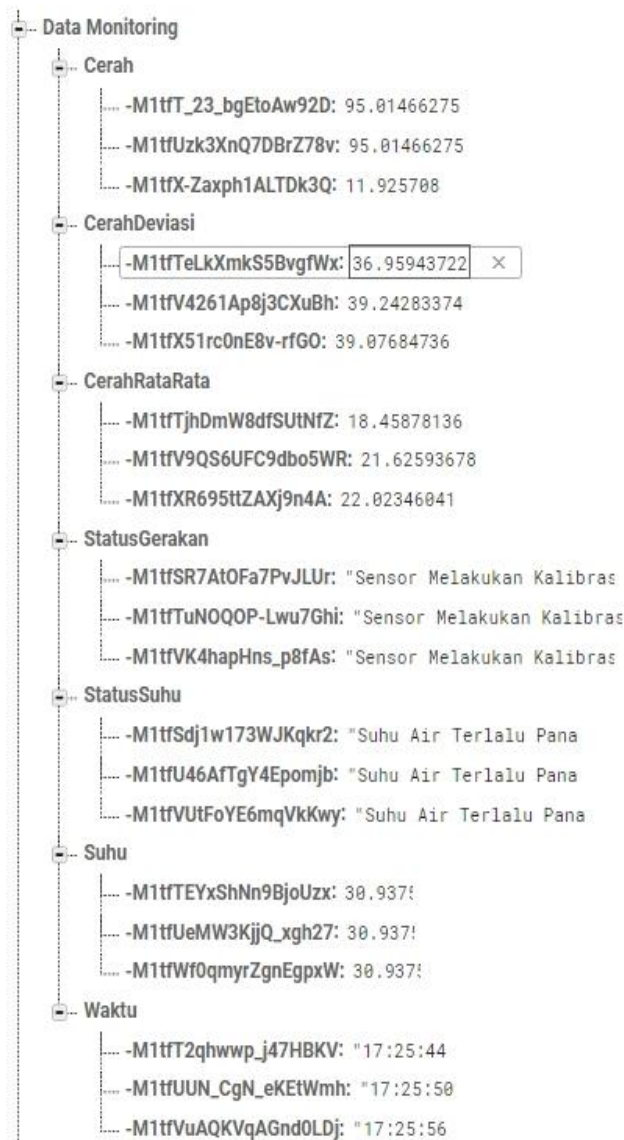
Data yang telah dihitung dengan fungsi akan dikirimkan ke Firebase dan terlihat data sensor yang terbaca tersebut akan melakukan pembaruan secara *realtime* selama perangkat *data logger* masih terkoneksi ke internet. Gambar 12 berikut ini merupakan struktur rancangan skema data pada Firebase.



Gambar 12. Rancangan Skema Data Tabel *Monitoring*

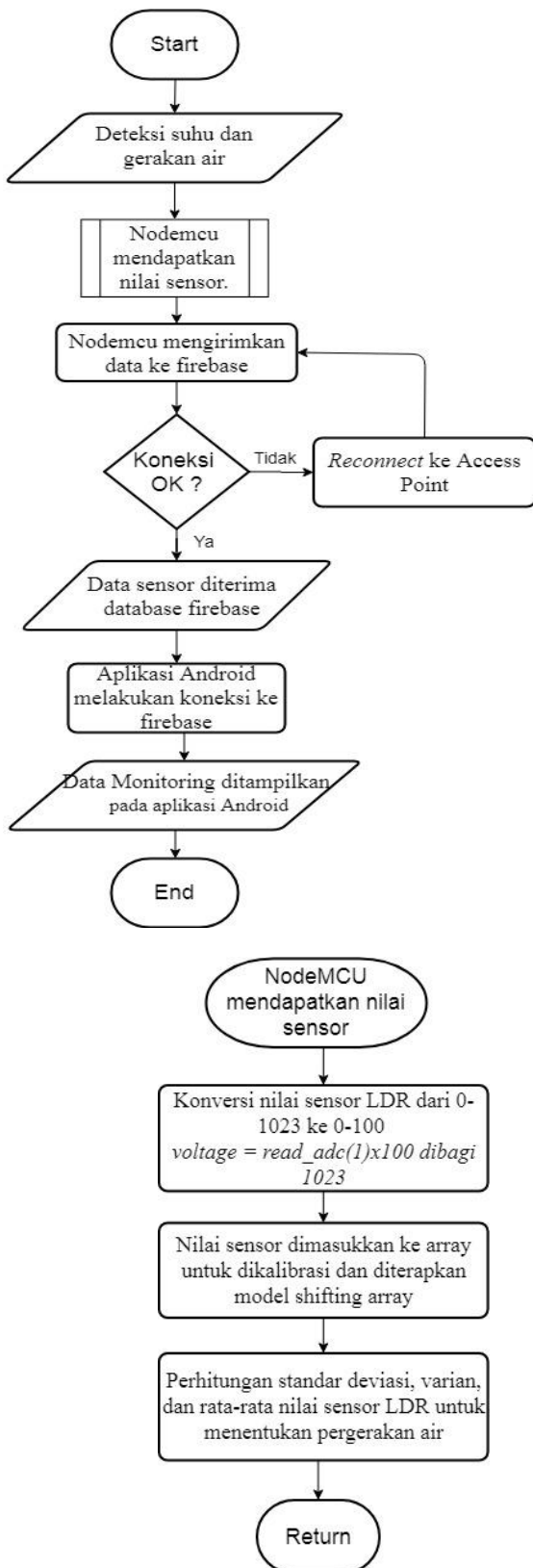
Data yang terdapat pada tabel *monitoring* merupakan tabel yang datanya akan ditampilkan pada aplikasi Android berbasis flutter. Untuk penjelasan mengenai aplikasi Android dapat dilihat pada sub bab perancangan perangkat

lunak pada *smartphone*. Gambar 13 merupakan struktur rancangan skema data pada Firebase untuk tujuan pengumpulan data.



Gambar 13. Rancangan Skema Data Tabel *Data Monitoring*

Tabel data *monitoring* digunakan untuk mengumpulkan data yang dikirimkan oleh NodeMCU. Data tersebut digunakan untuk pengujian pada nilai kedua sensor sehingga dihasilkan nilai yang sesuai dengan kondisi kualitas air pada tambak udang. Untuk tipe data yang digunakan yaitu string dan double. Semua proses diatas dijelaskan secara berurutan melalui *flowchart* pada Gambar 14.



Gambar 14. Flowchart Program Perangkat Data Logger

Algoritma 1 merupakan algoritma program dari NodeMCU ESP8266 untuk *array* dan *shifting array*.

Algoritma 1 Algoritma program *array* dan *shifting array*.

1. NodeMCU membaca nilai pada sensor suhu dan sensor LDR
2. Nilai yang terbaca dimasukkan ke array dengan panjang elemen 30 dengan tipe data double
3. Terdapat prosedur bernama *geser_ke_kanan* yang didalamnya terdapat perulangan yang berfungsi untuk menggeser nilai yang ada pada elemen 28 ke elemen 29
4. Proses *shifting* berlangsung secara terus menerus selama NodeMCU masih membaca nilai pada sensor
5. Selesai

Algoritma 2 adalah algoritma program dari NodeMCU ESP8266 untuk menghitung *variance*, *standar deviation*, dan *average / mean* dari nilai sensor.

Algoritma 2 Menghitung *Variance*, *Standar Deviation*, dan *Average*

1. Nilai sensor dari hasil *array shifting* dicatat pada prosedur *catat_suhu* dan *catat_cahaya* dari kode program 1 dibawah
2. Kemudian sebelum dikirim ke firebase, fungsi *standard deviation* melakukan perhitungan deviasi dibantu dengan fungsi *variance*
3. Disaat yang bersamaan, fungsi *mean / average* menghitung rata-rata nilai sensor sebelum dikirim ke sensor
4. Semua data akan akan di simpan pada firebase setelah proses perhitungan berhasil dilakukan
5. Selesai

Kode program 1 adalah kode program dari NodeMCU ESP8266 untuk mencatat suhu dan cahaya dan kode program untuk mengirim data ke Firebase.

Kode Program 1 Mencatat Data Sensor dan Mengirim Ke Firebase

```

void catat_suhu(double ini)
{
    geser_ke_kanan_suhu();
    t_arr[0] = ini;
}

void catat_cahaya(double ini)
{
    geser_ke_kanan_cahaya();
    s_arr[0]=ini;
}

void Temp ()
  
```



```

{
  tdata = sensors.getTempCByIndex(0);
  myDouble = double(tdata);
  catat_suhu(myDouble);
  Firebase.setDouble(firebaseData, path +
"/Suhu/Data",myDouble);
  Firebase.pushDouble(firebaseData,
datapath + "/Suhu" ,myDouble);
}

void Light ()
{
  sdata = analogRead(vr);

  convert = sdata * (100.0 /1023);
  myDouble2 = double(convert);
  catat_cahaya(myDouble2);
  cahaya_mean = myDouble2 / 30;
  Firebase.setDouble(firebaseData, path +
"/Cerah/Data",myDouble2);
  Firebase.setDouble(firebaseData, path +
"/Cerah_stdv/Data",standardDeviation(s_arr,
30));
  Firebase.setDouble(firebaseData, path +
"/Cerah_mean/Data",mean(s_arr));
  Firebase.pushDouble(firebaseData,
datapath + "/Cerah" ,myDouble2);
  Firebase.pushDouble(firebaseData,
datapath +
"/CerahDeviiasi" ,standardDeviation(s_arr,
30));
  Firebase.pushDouble(firebaseData,
datapath + "/CerahRataRata" ,mean(s_arr));
}

```

Kode program 2 adalah kode program dari NodeMCU ESP8266 untuk memberikan status gerakan udang dan status suhu air. Dikarenakan alat masih dalam tahap pengujian, maka pemberian status masih dilakukan beberapa kalibrasi dan pengujian.

Kode Program 2 Informasi Status Gerakan Udang dan Suhu Air

```

String status1 = "Sensor Melakukan Kalibrasi";
String status2 = "Gerak Air Pasif";
String status3 = "Gerak Air Aktif";
String status4 = "Suhu Air Terlalu Dingin";
String status5 = "Suhu Air Baik";
String status6 = "Suhu Air Terlalu Panas";

void StatusGerakan()
{
  if (standardDeviation(s_arr,30) > 10 ){
    Firebase.setString(firebaseData, path +
"/StatusGerakan/Data",status1);
    Firebase.pushString(firebaseData, datapath
+ "/StatusGerakan" ,status1);
  }

  if (standardDeviation(s_arr,30) < 1){
    Firebase.setString(firebaseData, path +
"/StatusGerakan/Data",status2);
    Firebase.pushString(firebaseData, datapath
+ "/StatusGerakan" ,status2);
  }

  if (standardDeviation(s_arr,30) > 1 &&
standardDeviation(s_arr,30) < 10){

```

```

    Firebase.setString(firebaseData, path +
"/StatusGerakan/Data",status3);
    Firebase.pushString(firebaseData, datapath
+ "/StatusGerakan" ,status3);
  }
}

void StatusSuhu ()
{
  if (myDouble < 25 ){
    Firebase.setString(firebaseData, path +
"/StatusSuhu/Data",status4);
    Firebase.pushString(firebaseData, datapath
+ "/StatusSuhu" ,status4);
  }

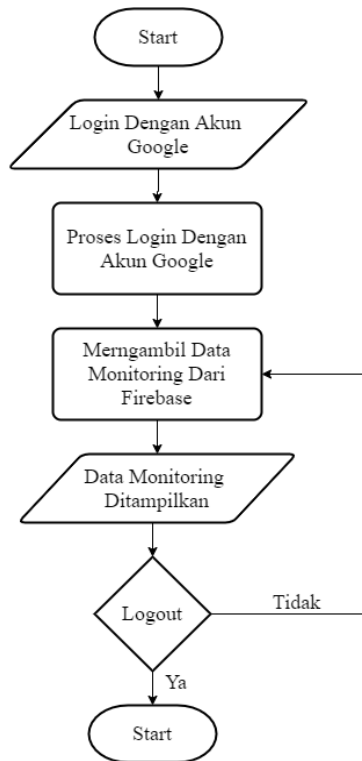
  if (myDouble > 25 && myDouble < 30){
    Firebase.setString(firebaseData, path +
"/StatusSuhu/Data",status5);
    Firebase.pushString(firebaseData, datapath
+ "/StatusSuhu" ,status5);
  }

  if (myDouble > 30){
    Firebase.setString(firebaseData, path +
"/StatusSuhu/Data",status6);
    Firebase.pushString(firebaseData, datapath
+ "/StatusSuhu" ,status6);
  }
}

```

D. Perancangan Perangkat Lunak Smartphone

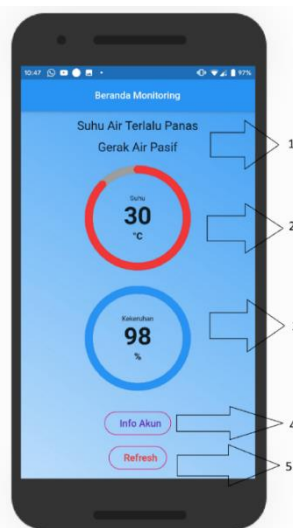
Pada bagian ini, dijelaskan perancangan perangkat lunak yaitu diagram alir untuk aplikasi Android (Gambar 15) dan perancangan GUI (Gambar 16). Aplikasi yang dirancang untuk *smartphone* Android dibuat menggunakan *framework* Flutter. Dalam merancang aplikasi Android, digunakan aplikasi VSCode untuk membangun aplikasi Android. Cara kerja aplikasi Android yaitu menampilkan halaman login menggunakan akun google, kemudian setelah berhasil login akan ditampilkan dua *widget* data sensor, status pergerakan air dan suhu air, serta dua button info akun dan *refresh*. Aplikasi Android hanya dapat berjalan menggunakan koneksi internet dan data akan diperbaharui apabila perangkat *data logger* diaktifkan dan terhubung ke internet.



Gambar 15. Flowchart Aplikasi Android

E. Perancangan GUI Aplikasi Android

Perancangan GUI untuk aplikasi pada Android terdiri dari penempatan widget yang digunakan (Gambar 16).



Gambar 16. Flowchart Aplikasi Android

Berdasarkan Gambar 16 terdapat komponen *widget* yang sudah ditandai dengan angka, untuk penjelasan fungsi dan nama *widget* terdapat pada Tabel III.

TABEL III
WIDGET YANG DIGUNAKAN DALAM GUI

No	Nama Komponen / Widget	Fungsi
1	Text	Menampilkan status keadaan air dan suhu air
2	Custom Painter	Menampilkan nilai suhu kedalam gauge yang memiliki animasi
3	Custom Painter	Menampilkan kekeruhan air kedalam gauge yang memiliki animasi
4	Button	Membuka tampilan info akun
5	Button	Melakukan refresh pada dashboard <i>monitoring</i> untuk mendapatkan data baru

Kode program 3 merupakan program dari aplikasi Android untuk mendapatkan data dari *database* Firebase.

Kode Program 3 Mendapatkan Data dari *Database*

```

@Override
void initState() {
    super.initState();

    databaseReference
        .child('Monitoring')
        .once()
        .then((DataSnapshot snapshot) {
            double suhu =
            snapshot.value['Suhu']['Data'];
            double cerah =
            snapshot.value['Cerah']['Data'];

            isLoading = true;
            _dashboardInit(suhu, cerah);
        });
}

new StreamBuilder<Event>(
    stream:
    FirebaseDatabase.instance
        .reference()
        .child('Monitoring')
        .onValue,
    builder:
    (BuildContext context,
    AsyncSnapshot<Event> event) {
        Map<dynamic, dynamic>
        data = event.data.snapshot.value;
        return Column(children:
        [
            new
            Text('${data['StatusSuhu']['Data']}',
                style: new
                TextStyle(fontSize: 25.0)),

            ],
        ),
    ),
    new StreamBuilder<Event>(
    stream:
    FirebaseDatabase.instance
        .reference()
        .child('Monitoring')
    
```

```

        .onValue,
        builder:
          (BuildContext context,
          AsyncSnapshot<Event> event) {
            Map<dynamic, dynamic>
            data = event.data.snapshot.value;
            return Column(children:
            [
              new
              Text('${data['StatusGerakan']]['Data']}',
              style: new
              TextStyle(fontSize: 25.0)),
            ]);
          },
        ),
    ),

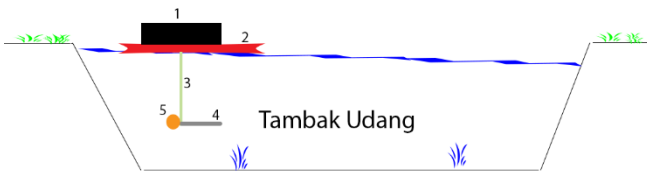
```

V. IMPLEMENTASI SISTEM

Saat ini realisasi sistem masih pada tahap pengujian dan pengembangan, adapun untuk realisasi sistem terdapat dua bagian yaitu realisasi pada perangkat *data logger* dan realisasi pada aplikasi Android.

A. Realisasi perangkat *data logger*

Pada realisasi perangkat *data logger* terdapat implementasi di tambak udang (Gambar 17) dan rancangan alat yang sedang proses perancangan, terdapat dua sensor yang masih dikalibrasi yaitu sensor suhu dan sensor LDR (Gambar 18), perangkat akan dipasang pada pelampung agar alat *data logger* tidak terkena air tambak.



Gambar 17. Implementasi Alat Ketika di Tambak

Berdasarkan Gambar 17 terdapat beberapa komponen yang sudah diberikan angka yang penjelasannya dapat dilihat pada Tabel IV.

TABEL IV
WIDGET YANG DIGUNAKAN DALAM GUI

No	Nama	Fungsi
1	Black Box	Merupakan tempat dimana semua perangkat keras <i>data logger</i> terintegrasi menjadi satu
2	Pelampung	Untuk membantu perangkat keras data logger selalu diatas permukaan air

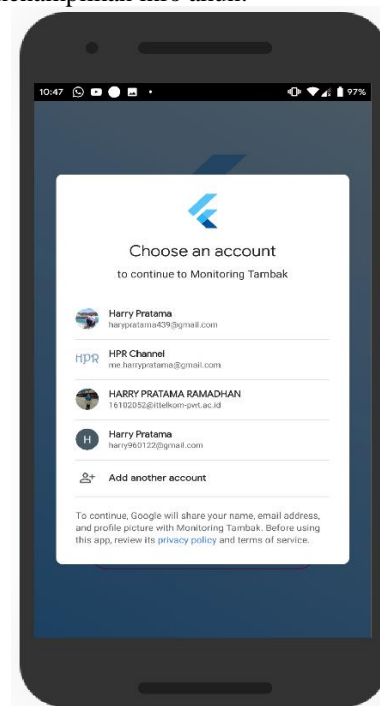
		(tidak tenggelam)
3	Pipa Kabel Listrik	Tempat untuk meletakkan kabel sensor
4	Sensor Suhu	Sensor yang digunakan mengukur suhu air
5	Sensor LDR	Sensor yang digunakan untuk mengukur aktivitas pergerakan air dan udang dan mengukur kekeruhan air



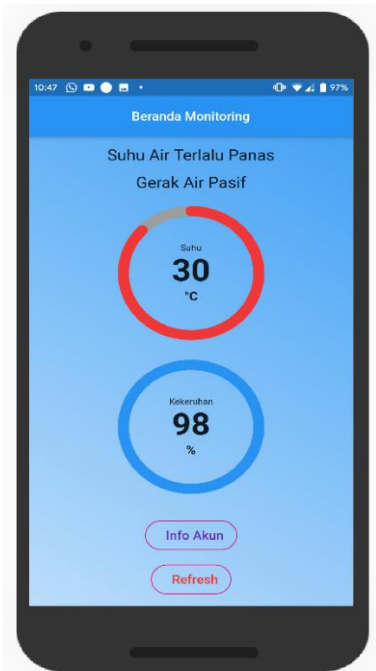
Gambar 18. Rancangan Alat

B. Realisasi aplikasi Android

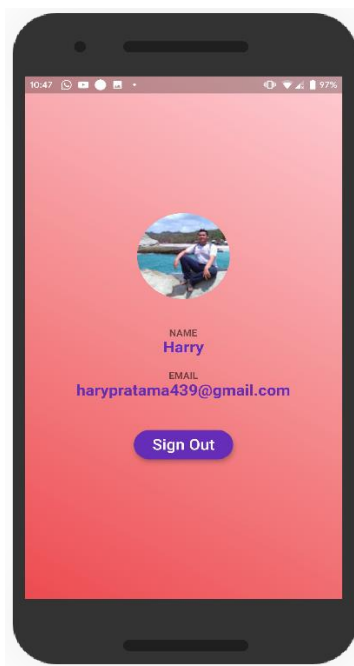
Realisasi aplikasi pada Android ditunjukkan dengan beberapa menu yaitu Gambar 19 menampilkan halaman login, Gambar 20 menampilkan beranda *monitoring* dan Gambar 21 menampilkan info akun.



Gambar 19. Tampilan Login



Gambar 20. Tampilan Dashboard Monitoring



Gambar 21. Tampilan Info Akun

VI. HASIL PENGUJIAN

A. Pengujian Kalibrasi

Pada bagian ini dijelaskan pengujian kalibrasi pada sensor dengan menggunakan *array* sepanjang 30 elemen.

Setiap perangkat *data logger* dimatikan, maka *array* tersebut akan kembali ke nilai 0. Proses kalibrasi sensor untuk saat ini masih memiliki rentang nilai standar deviasi diatas 10.

TABEL V
PENGUJIAN KALIBRASI

Jam	Rata-rata Kekeruhan	Standar Deviasi Cerah	Status Gerak	Status Suhu
01:50:30	76.39622	2.0	Sensor Melakukan Kalibrasi	Suhu Air Baik
01:50:36	79.34506	2.0	Sensor Melakukan Kalibrasi	Suhu Air Baik
01:50:42	82.29390	1.73205	Sensor Melakukan Kalibrasi	Suhu Air Baik
01:50:48	85.24275	1.41421	Sensor Melakukan Kalibrasi	Suhu Air Baik
01:50:56	88.19485	1.0	Gerak Air Pasif	Suhu Air Baik

Dari Tabel V didapatkan pengujian kalibrasi dari sensor LDR dan didapatkan standar deviasi yang bervariasi, dan sensor mendeteksi gerak air pasif. Lalu untuk status suhu dinyatakan dengan status suhu air baik karena berdasarkan artikel [6] Suhu air tambak yang baik berkisar antara 20° C sampai 31° C. Rata-rata kekeruhan semakin tinggi mendekati nilai kekeruhan pada Tabel VI.

TABEL VI
PENGUJIAN DATA SENSOR

Jam	Kekeruhan	Suhu	Status Gerak	Status Suhu
01:50:30	88.46529	28.937	Sensor Melakukan Kalibrasi	Suhu Air Baik
01:50:36	88.46529	28.937	Sensor Melakukan Kalibrasi	Suhu Air Baik
01:50:42	88.46529	28.937	Sensor Melakukan Kalibrasi	Suhu Air Baik
01:50:48	88.46529	28.937	Sensor Melakukan Kalibrasi	Suhu Air Baik
01:50:56	88.56304	28.937	Gerak Air Pasif	Suhu Air Baik

VII. KESIMPULAN

Sistem *monitoring* kualitas air tambak udang berbasis IoT dan Android yang terdiri dari perangkat *data logger* dan aplikasi Android untuk saat ini masih dalam tahap pengembangan. Data yang dikirim dari *data logger* ke Firebase sudah dapat berjalan dengan baik melalui jaringan internet. Untuk mendapatkan status kualitas air tambak

udang, perangkat *data logger* dan Android harus terhubung ke internet kemudian letakkan perangkat *data logger* di atas permukaan air tambak udang, lalu tunggu selama beberapa menit hingga sensor selesai melakukan kalibrasi. Hasil data yang didapatkan oleh kedua sensor yaitu berupa kekeruhan, suhu, standar deviasi dan nilai rata-rata pergerakan air dan udang, serta kondisi kualitas air dan suhu air dapat diakses melalui aplikasi Android. Adapun kendala dari penelitian ini yaitu dua tempat tambak yang dijadikan tempat untuk melakukan penelitian mengalami bangkrut dan adanya pandemi saat ini membuat kegiatan penelitian seperti pengambilan data di tempat tambak belum bisa dilakukan karena tempat tambak berada di luar kota. Diharapkan sistem *monitoring* tersebut dapat membantu pengelola tambak udang dalam mendapatkan informasi mengenai kualitas air tambak udang.

VIII. SARAN

Untuk pengembangan sistem *monitoring* kualitas air tambak udang berbasis IoT dan Android selanjutnya yaitu untuk mengambil data akan dibuatkan menu khusus di aplikasi Android, karena saat ini masih manual dari Firebase dengan cara *export to JSON*.

DAFTAR PUSTAKA

- [1] Kementerian Kelautan dan Perikanan, "Laporan kinerja kementerian kelautan dan perikanan," *Annual Rep.*, p. 32 - 117, 2018.
- [2] S. Adiwijaya. (2016) Indonesia berpotensi kembangkan industri akuakultur. [Online]. Tersedia: <https://bisnis.tempo.co/read/799904/indonesia-berpeluang-kembangkan-industri-akuakultur/full&view=ok>.
- [3] F. Gheost. (2018) Parameter kualitas air berdasarkan faktor kimia, fisika dan biologinya. [Online]. Tersedia: <https://www.geolognesia.com/2018/08/parameter-kualitas-air.html>.
- [4] A. Junaidi & C. Kartiko, "Design of pond water quality monitoring system based on internet of things and pond fish market in real-time to support the industrial revolution 4.0," *IOP Conf. Ser. Mater. Sci. Eng.*, 2020, vol. 771, p. 1-6, doi: 10.1088/1757-899x/771/1/012034.
- [5] A. Yulianto. (2011) Data logger (bagian 1) *edutechment, technology*. [Online]. Tersedia: <https://sonoku.com/data-logger-bagian-1/>
- [6] (2017) Menjaga kestabilan parameter air tambak. [Online]. Tersedia : <https://tambakudang.com/menjaga-kestabilan-parameter-air-tambak/>
- [7] S. Puji Irawan. (2019) Pelajari tentang sensor suhu DS18B20 dan bagaimana penyambungan alat tersebut sebagai input pada perangkat raspberry pi sebagai sensor suhu sebuah ruangan. [Online]. Tersedia: <https://k1801.ilearning.me/2017/02/26/pelajari-tentang-sensor-suhu-ds18b20-dan-bagaimana-penyambungan-alat-tersebut-sebagai-input-pada-perangkat-raspberry-pi-sebagai-sensor-suhu-sebuah-ruangan/>
- [8] (2018) DS18B20 temperature sensor. [Online]. Tersedia: <https://components101.com/sensors/ds18b20-temperature-sensor>
- [9] (2019) *Water Temperature Sensor : DS18B20 data sheet*, 19-7487, Rev 6, Maxim Integrated, American. [Online]. Tersedia: <https://datasheets.maximintegrated.com/en/ds/DS18B20.pdf>
- [10] (2015) *ESP8266EX data sheet, espr. syst. datasheet*, espressif systems, Shanghai. [Online]. Tersedia: https://www.adafruit.com/images/product-files/2471/0A-ESP8266_Datasheet_EN_v4.3.pdf
- [11] (2018) Pengertian sensor ldr, fungsi dan cara kerja ldr. [Online]. Tersedia : <http://www.immersa-lab.com/pengertian-sensor-ldr-fungsi-dan-cara-kerja-ldr.html>
- [12] (2017) Light dependent resistor (ldr). [Online]. Tersedia : <https://components101.com/index.php/ldr-datasheet>
- [13] Guntoro. (2019) Memahami " apa itu firebase " hanya dalam 10 menit. [Online]. Tersedia: <https://badoystudio.com/apa-itu-firebase/>
- [14] M. Arumsari. (2019) Microsoft visual studio code: seperti apa fiturnya? [Online]. Tersedia: <https://www.dicoding.com/blog/microsoft-visual-studio-code/>
- [15] A. Islamic Hermawan. (2018) Apa itu android? penjelasan super lengkap android ada disini!. [Online]. Tersedia: <https://windowsku.com/apa-itu-android-adalah/>
- [16] (2016) Android - arsitektur sistem operasi android. [Online]. Tersedia : <https://www.kapalomen.com/2016/07/android-arsitektur-android-sistem-operasi-android.html>