

## BAB I PENDAHULUAN

### 1.1. Latar Belakang

Ketinggian air merupakan suatu istilah yang digunakan untuk menyatakan posisi atau keberadaan air dari suatu dasar. Dimana dengan mengetahui ketinggian air, maka bisa dijadikan sebuah prediksi apa ada kemungkinan air akan meluap dan terjadi banjir. Banjir adalah peristiwa bencana alam yang terjadi ketika aliran air yang berlebihan merendam daratan. Banjir diakibatkan oleh volume air di suatu badan air seperti sungai meluap dari bendungan sehingga air keluar dari badan air tersebut. Ukuran Badan air akan terus berubah seiring perubahan curah hujan sehingga banjir dapat terjadi sewaktu – waktu. Fenomena banjir telah meresahkan dan mengakibatkan kerugian material pada korban seperti alat elektronik yang rusak, barang – barang terendam dalam air, dan sebagainya.

Peranan teknologi sangat dibutuhkan untuk membantu mengantisipasi warga yang tinggal berdekatan dengan aliran air. Meskipun dampak yang ditimbulkan cukup merugikan tetapi, bencana banjir ini dapat dihindari jika masyarakat telah memiliki persiapan sebelumnya sehingga dapat meminimalisir kerugian yang disebabkan banjir. Oleh sebab itu, para masyarakat perlu mengetahui ketinggian air dari badan air tersebut untuk segera mengungsi jika ketinggian hampir mencapai batas tinggi badan air tersebut.

Ada berbagai cara untuk mengukur ketinggian air dan salah satu cara tersebut adalah menggunakan sensor ultrasonik. Dibandingkan alat pengukuran lainnya, sensor ultrasonik jauh lebih efektif. Selain itu, sensor ultrasonik juga mudah didapatkan dan harganya yang terjangkau. Sensor ultrasonik jika diaplikasikan dengan baik bisa mencatat apapun itu setiap detiknya. Berbeda dengan barang pengukur lainnya yang harus dilakukan secara manual.

Oleh sebab itu, dalam proyek kali ini, kelompok kami akan menggunakan sensor HCSR04 yang merupakan sensor yang memanfaatkan prinsip gelombang ultrasonik. Dimana dengan prinsip ini, kita dapat mengetahui ketinggian level air yang mempermudah manusia dalam pengukuran dan segera melakukan pengungsian sebelum terjadinya banjir.

## 1.2. Tujuan

Adapun tujuan dari dibuatnya proyek ini adalah untuk mencari tahu level ketinggian air sebagai Sensor Peringatan Dini banjir menggunakan sensor HCSR04 dan nodeMCU ESP8266.

## 1.3. Manfaat

Adapun manfaat yang dihasilkan proyek ini adalah dapat mengetahui ketinggian banjir secara presisi untuk mencegah datangnya banjir yang lebih parah dengan data yang disimpan di dalam firebase sebagai data.

## 1.4. Deskripsi Inovasi Alat

Proyek kali ini mengalami perubahan yang cukup signifikan setelah UTS diantaranya:

- Dapat menyimpan data dalam firebase

Pada UTS, data disimpan dalam google spreadsheet namun, terlihat beberapa kekurangan sangat melakukan uji coba. Data yang dikirim akan menjadi lambat ketika total data mencapai 1000 yang berarti google spreadsheet kurang bisa digunakan untuk big data.

Dengan firebase, data bisa disimpan lebih banyak dari sebelumnya. Meskipun ada beberapa kelambatan ketika data menjadi sangat besar, namun firebase masih jauh lebih memadai untuk penyimpanan data

- Melakukan Visualisasi data dengan ThinkSpeak

Pada UTS, data yang dibaca oleh sensor hanya bersifat menyimpan tanpa ada visualisasi data. Namun, dalam proyek perkembangan ini, data bisa divisualisasikan berupa Line chart yang akan tertampil dalam web ThinkSpeak.

- Terdapat Buzzer

Pada UTS, proyek tidak memiliki buzzer, proyek sebelumnya hanya mengandalkan sensor HCSR-04 saja. Namun, dalam perkembangan proyek ini, kami menambahkan buzzer yang akan berbunyi ketika ketinggian melewati ketinggian tertentu dengan tujuan sebagai peringatan dini.

- Penataan proyek lebih stabil dan lebih rapi

Pada UTS, proyek belum tersusun secara rapi, proyek hanya dipasangkan oleh dua sumpit agar menjadi stabil. Namun, dalam perkembangan proyek ini, proyek disusun

diatas triplek sehingga kabel tidak akan menghalangi pembacaan serta pembacaan sensor jauh lebih stabil.

- Pembacaan sensor lebih detail

Pada UTS, data yang diberikan sensor selalu dibulatkan ke atas atau ke bawah. Namun, dalam perkembangan proyek ini, sensor bisa membaca hingga dua angka di belakang koma.

- Proyek menjadi portable

Pada UTS, proyek harus selalu disambungkan ke laptop agar bisa sensor bisa mendapatkan data. Namun, dalam perkembangan proyek ini, sensor hanya membutuhkan powerbank yang bisa diisi kembali dan bersifat portable.

## BAB II TINJAUAN PUSTAKA

### 2.1 Rancang Bangun Sistem Peringatan Dini Banjir Berbasis Sensor Ultrasonik Dan Mikrokontroler Sebagai Upaya Penanggulangan Banjir.” (Umari, C., Anggraini, E., & Zainul Muttaqinm. Rofif, 2017)

Pada penelitian ini membahas tentang membangun suatu alat yang dapat dijadikan sebagai alat peringatan dini bencana banjir. Sensor ultrasonik yang berfungsi sebagai pembaca ketinggian permukaan air dan kemudian diteruskan ke perangkat selanjutnya yang berupa sistem minimum mikrokontrol yang berfungsi untuk memproses masukan dengan basis pengetahuan yang sudah ditanamkan oleh programmer kemudian memerintahkan perangkat keluaran untuk menghidupkan buzzer. Adapun kesimpulan dari penelitian ini adalah:

- Alat dapat dioperasikan dengan baik sesuai instruksi program yang dimasukkan kedalam mikrokontrol oleh programmer, dimana sensor ultrasonik dapat membaca jarak permukaan air, buzzer berbunyi sesuai keadaan yang ditentukan.
- Ketinggian permukaan air dapat diukur secara otomatis dengan memanfaatkan sensor ultrasonik dan mikrokontroler.
- Mode-mode pemantauan ketinggian permukaan air yang bekerja sesuai jadwal pewartuannya

### 2.2 Perancangan Sistem Pendeteksi Banjir dengan Menggunakan Sensor HC-SR04 Berbasis Aduino Uno” (Bernandus, Johnshon Tarigan, & Tanseib, Jehunias Leonidas, 2019)

Pada sistem ini dibuat sistem secara umum sehingga diperoleh gambaran rangkaian sistem secara keseluruhan. Sistem mendeteksi banjir secara otomatis ini memiliki prinsip kerja, yaitu sensor ultrasonik sebagai input yaitu membaca serta mengukur level ketinggian air dan mikrokontroler yang dipakai ialah Mikrokontroller arduino uno yang akan memproses input dari sensor ketinggian yang selanjutnya akan memberikan perintah on atau off kepada Buzzer. Berdasarkan hasil penelitian yang dilakukan, dapat ditarik kesimpulan sebagai berikut:

- Sensor ultrasonik dapat digunakan sebagai alat mendeteksi banjir secara otomatis dengan mengukur level ketinggian air.
- Data pengukuran level ketinggian air 2 – 5 cm, status darurat banjir dan Buzzer ON sampai ketinggian 10 cm, sedangkan ketinggian air 6-20 cm status siaga, dan di atas

20cm status aman , sedangkan buzzer (Alarm) OFF mulai 11 cm.

### 2.3 Analisis Cara Kerja Sensor Ultrasonik Menggunakan Mikrokontroler Arduino Uno Untuk Merancang Alat Deteksi Banjir Secara Otomatis.” (Risdiandi, Rahmat, 2020)

Di dalam pembahasan program ini, peneliti hanya membahas alur kerja pada sensor Ultrasonik menggunakan program yang akan dikontrol dengan arduino uno dan dijalankan sesuai sesuai perancangan yang sebelumnya telah di rancang oleh peneliti. Oleh karena itu sebelum dilakukannya pengujian pada sensor Ultrasonik, peneliti mencoba memastikan untuk menjalankan program yang sudah di buat menggunakan *software* IDE arduino pada rangkaian modul yang telah dirancang sebelumnya. Pengujian ini bertujuan untuk mengetahui cara kerja sensor ultrasonik untuk mendeteksi ketinggian air dan memberikan alarm peringatan saat sensor mendeteksi level ketinggian air, ketinggian air terbagi menjadi 4 bagian yaitu level aman, siaga, awas, dan bahaya. Dari analisa dan pembahasan diatas yang telah dibuat, memperoleh kesimpulan sebagai berikut:

- Penelitian ini berhasil membuat model atau prototype peringatan dini banjir secara otomatis yang dapat bekerja sesuai dengan rancangan peneliti.
- Sensor Ultrasonik dapat mendeteksi sangat baik pada media air, dimana naik dan turunnya air dapat terdeteksi.
- Sensor Ultrasonik memiliki titik buta , yaitu di saat jarak antara ultrasonic dan air kurang dari 1cm.

### 2.4 Sensor Ultrasonik HC-SR04

Sensor Ultrasonik adalah sensor yang bekerja dengan prinsip pantulan gelombang suara. Sensor akan menghasilkan gelombang suara yang kemudian ditangkap kembali. Perbedaan waktu akan ditangkap dan dimanfaatkan untuk mengukur celah antara ketinggian wadah dengan ketinggian air. Yang akhirnya, ketinggian air dapat diketahui.

### 2.5 Arduino UNO

Software arduino yang digunakan adalah driver dan IDE. IDE atau Integrated Development Environment merupakan suatu program khusus untuk suatu komputer agar dapat membuat suatu rancangan atau sketsa program untuk papan Arduino IDE. Arduino merupakan software yang sangat canggih ditulis dengan menggunakan java

## 2.6 ThingSpeak

ThingSpeak adalah platform informasi sumber IoT API terbuka berbasis web [04, 05, 06] yang komprehensif dalam menyimpan data sensor dari beragam 'aplikasi IoT' dan menggabungkan keluaran data yang dirasakan dalam bentuk grafis di tingkat web. ThingSpeak mengambil, menyimpan/menyimpan, menganalisis, mengamati, dan mengerjakan data yang dirasakan dari sensor yang terhubung ke mikrokontroler host seperti 'Arduino, modul TI CC3200, Raspberry-pi dan sebagainya. ThingSpeak adalah istilah 'Saluran' yang memiliki bidang untuk data, bidang untuk lokasi, bidang untuk status untuk data penginderaan yang bervariasi. Setelah saluran dibuat di 'ThingSpeak', data dapat diimplementasikan\_ dan secara bergantian seseorang dapat memproses dan memvisualisasikan informasi menggunakan MATLAB dan menanggapi data dengan tweet dan bentuk peringatan lainnya.

## 2.7 Google Firebase

Firebase adalah kerangka kerja yang berguna untuk membangun aplikasi portabel dan web untuk bisnis yang membutuhkan basis data waktu nyata yang menyiratkan ketika satu pengguna memperbarui catatan dalam basis data, pembaruan harus disampaikan kepada setiap pengguna secara instan. Ini memberikan platform dasar dan terpadu untuk banyak aplikasi bersama dengan sejumlah fitur Google lainnya yang dikemas dengan layanan.

Pelayanan Firebase yang digunakan adalah Real-Time Database yang merupakan basis data yang dihosting di cloud. Data disimpan sebagai JSON dan disinkronkan secara terus-menerus ke setiap klien terkait. Saat aplikasi lintas platform dikembangkan dengan iOS, Android, dan JavaScript SDK, sebagian besar permintaan pengguna didasarkan pada satu instans database waktu nyata dan instans ini diperbarui dengan setiap data baru.

## 2.8 Sensor Peringatan Dini

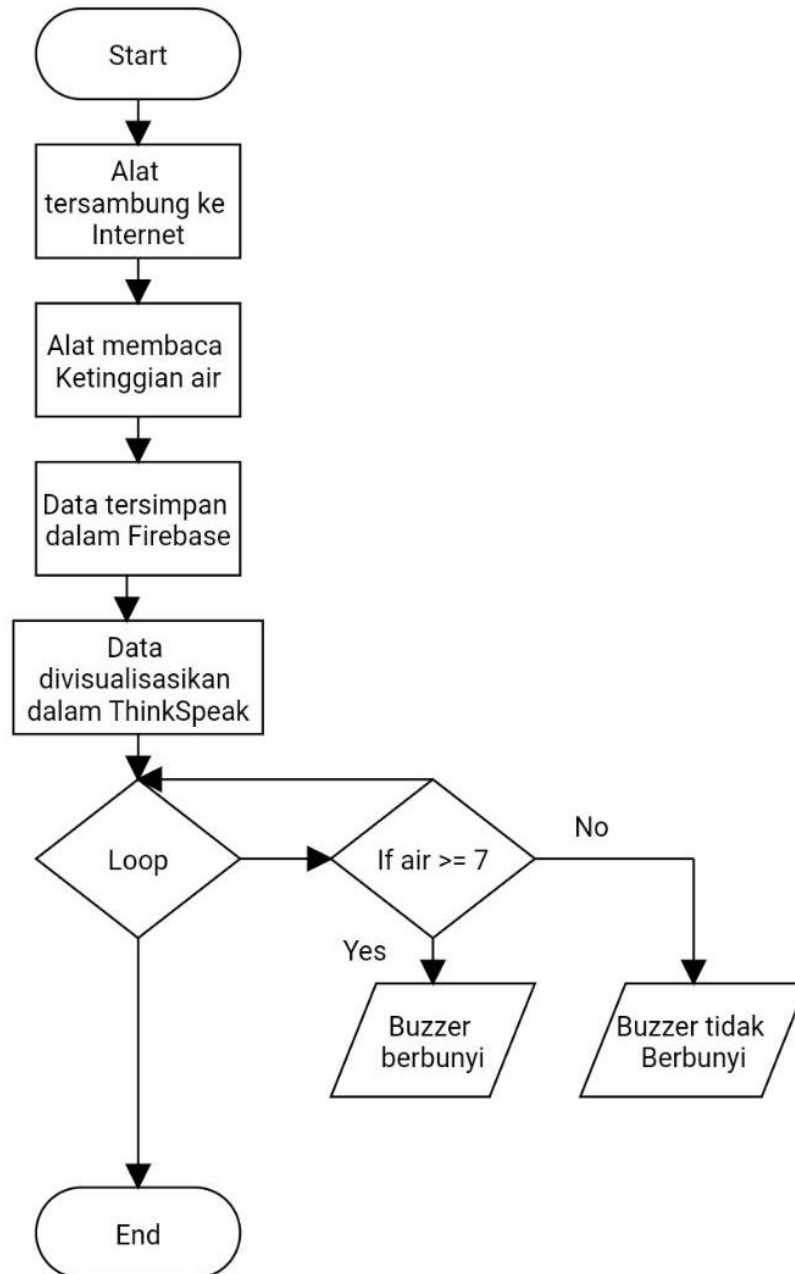
Sistem peringatan banjir merupakan sistem peringatan dimana jika ketinggian air menghasilkan nilai tertentu, maka akan menentukan status peringatan banjir. Tujuan dari sistem peringatan dini adalah untuk meminimalisir kerugian yang disebabkan oleh banjir dan agar masyarakat yang tinggal di daerah pesisir bisa segera mengungsi sebelum banjir datang.

## 2.9 Mikrokontroler

Sebuah chip IC sebagai penerima sinyal input atau biasa di sebut dengan Mikrokontroler, di olah dan mengirimkan sinyal output sama dengan program yang dimasukkan.

## BAB III METODE PENELITIAN

### 3.1 Desain Sistem Secara Umum



*Gambar 1: Desain Sistem Secara Umum*



### 3.1.1 Alat dan Bahan

#### A. Software

No.	Software	Spesifikasi
1.	Arduino IDE	Arduino
2.	Firebase	Google

*Tabel 1: Software*

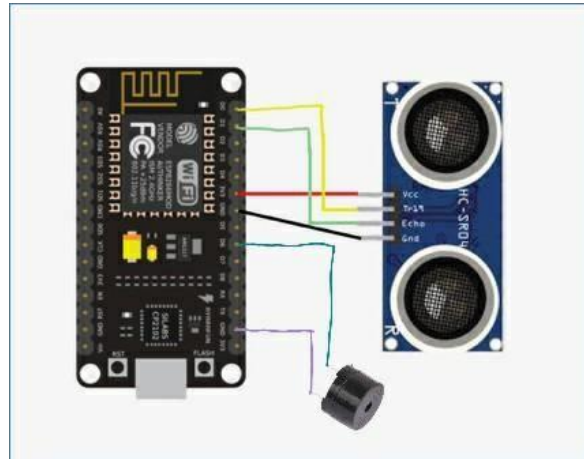
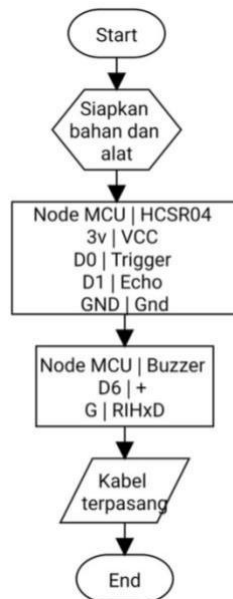
#### B. Hardware

No.	Bahan	Spesifikasi
1.	Esp8266	Tegangan: 3.3 v Frequency: 2.4 GHZ
2.	4 Kabel jumper (female to female)	Panjang: 30 cm
3.	2 Kabel jumper (male to female)	Panjang: 20 cm
4.	Bread Board Mini	Board kecil SYB-170
5.	Sensor Ultrasonic	Jenis: HC-SR04 Tegangan: 5v
6.	Buzzer Mini Active	Tegangan: 5v Dimensi: 9mm x 12mm Arus maks: 3mA
7.	Kabel USB	Panjang: 100 cm
8.	Powerbank	Merek: Robot Capacity: 5200 mAh Input: 5v/1A Output: 5v/1A
9.	2 Botol Air Keruh	Konsentrasi: 0.0285 g/ml Volume: 350 ml

*Tabel 2: Hardware*

### 3.1.2 Diagram Alir Dan Prosedur Kerja

#### a. Merangkai ESP8266 dengan HCSR04 dan Buzzer



*Gambar 2: Rangkaian*

Gambar di atas merupakan rangkaian dari sistem proyek ini, di mana sensor ultrasonic terdapat 4 sub kategori yaitu VCC, trigg, echo, dan ground (Gnd). VCC akan dipasangkan dengan 3v, trigg akan dipasangan dengan D0, echo akan dipasangkan dengan D1, dan ground (Gnd) akan dipasangkan dengan ground (GND).

Lalu tanda (+) dalam buzzer dipasangkan dengan D6 dan tanda RIHxD dipasangkan dengan Gnd. Setelah semuanya dipasangkan, sambungkan ESP8266 ke laptop menggunakan kabel USB.

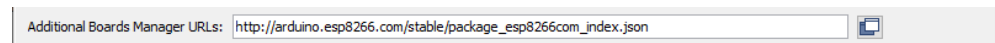
**b. Membuat *Coding* yang Sesuai Menggunakan Arduino**



*Gambar 3: Flowchart Coding*

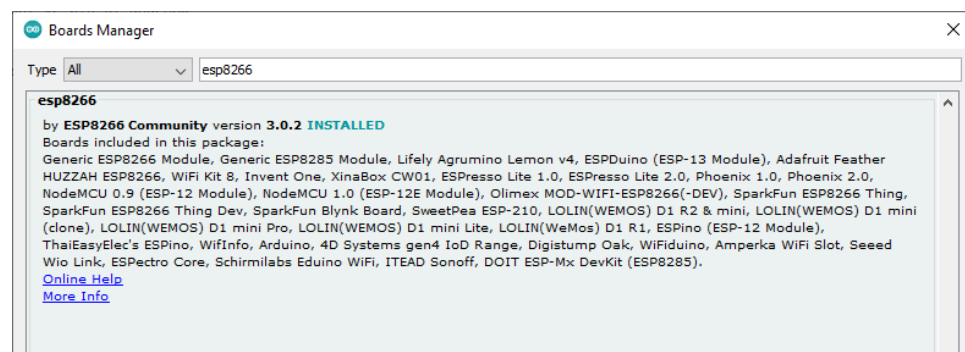
- 1) Buka aplikasi arduino (jika belum punya, download melalui microsoft store atau apk bawaan lainnya).
- 2) Klik menu arduino (file) → preferences → tulis link di bawah di bagian “additional boards manager URLs” —→klik OK.

Link: [http://arduino.esp8266.com/stable/package\\_esp8266com\\_index.json](http://arduino.esp8266.com/stable/package_esp8266com_index.json)



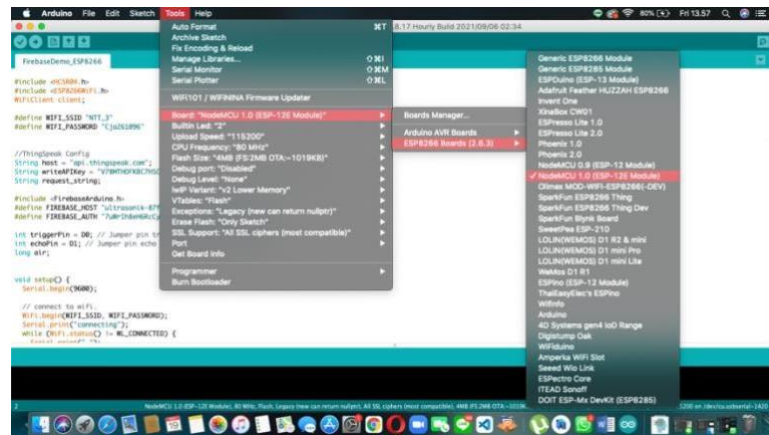
*Gambar 4: Coding 1*

- 3) Klik menu tools dan pilih boards → boards manager → cari nama esp dan muncul seperti dibawah → klik install dan tunggu hingga tertampil installed.



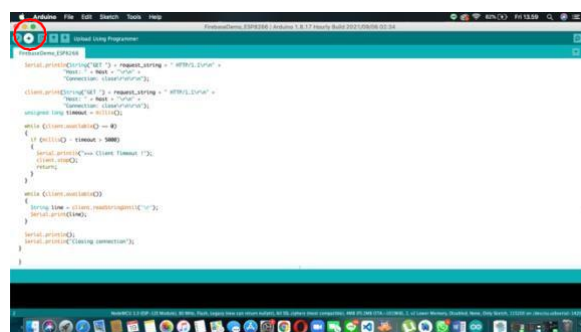
*Gambar 5: Coding 2*

- 4) Lalu klik menu tools dan pilih boards → klik ESP8266 Boards (3.0.2) → NodeMCU 1.0 (ESP-12E Module).



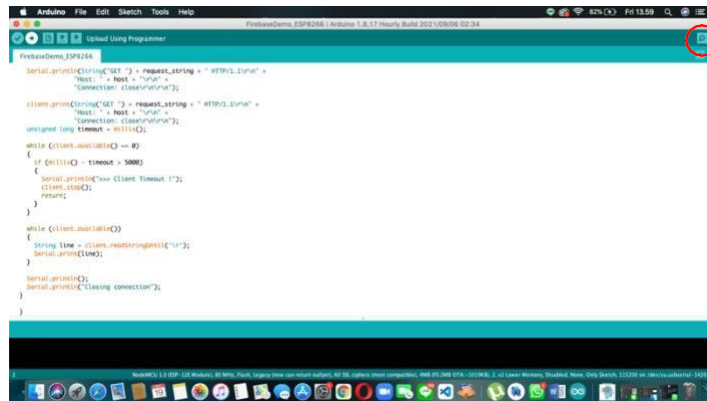
Gambar 6: Coding 3

- 5) Masukkan coding yang berada di link berikut:  
(Note: - WIFI\_SSID dan WIFI\_PASSWORD menyesuaikan  
- FIREBASE\_HOST dan FIREBASE\_AUTH menyesuaikan  
- Jika lampu led pada esp8266 menyala kedap – kedip itu berarti esp8266 telah tersambung ke wifi sehingga data yang diberikan sensor bisa disimpan
- 6) Klik upload bentuk tanda panah) → lalu tunggu hingga selesai.



Gambar 7: Coding 4

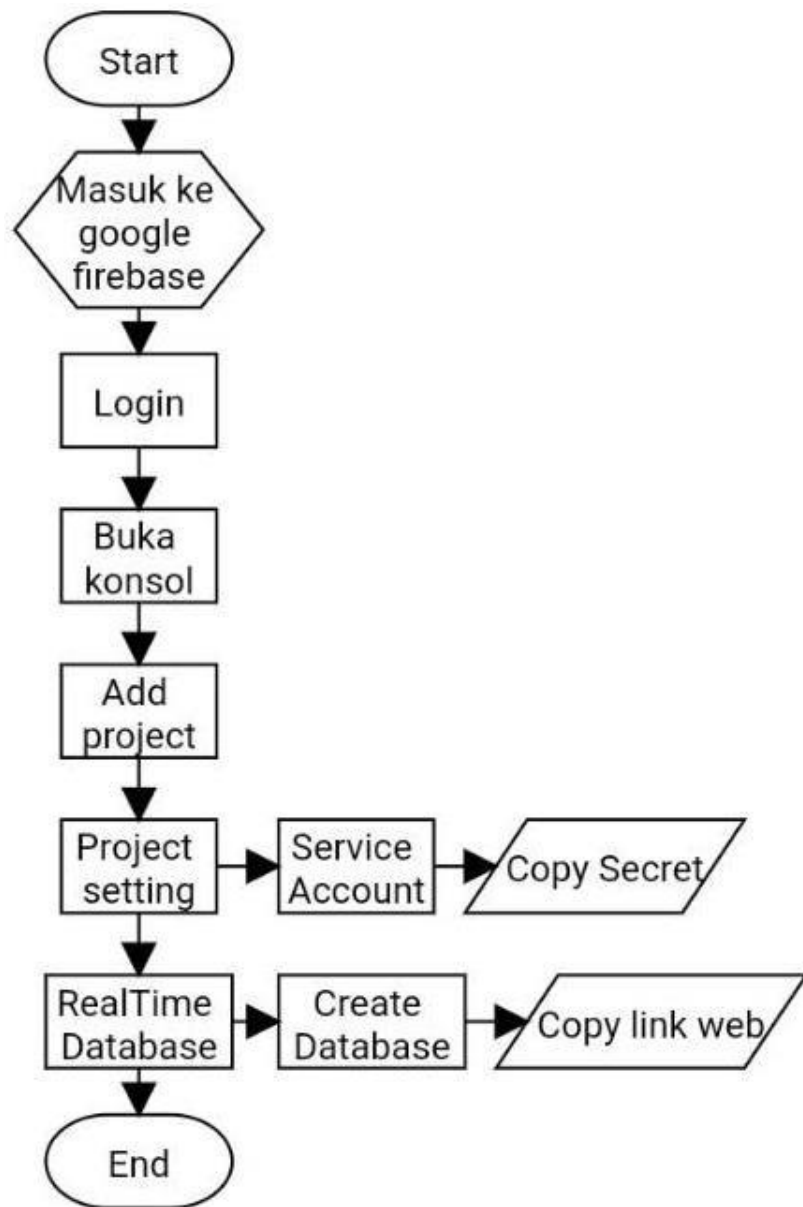
- 7) Buka serial monitor (bentuk kaca pembesar) untuk melihat monitor hasil yang dihasilkan sensor.



*Gambar 8: Coding 5*

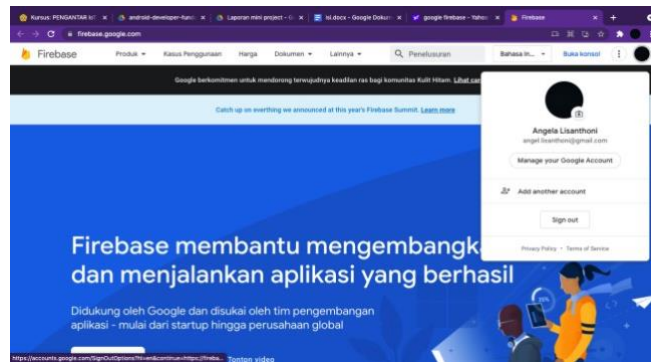
- 8) Lakukan pengecekan dengan mengukur ketinggian suatu air di dalam wadah yang mana jika berhasil, maka setiap kali penambahan air, ketinggian juga akan bertambah

c. Membuat Google Firebase



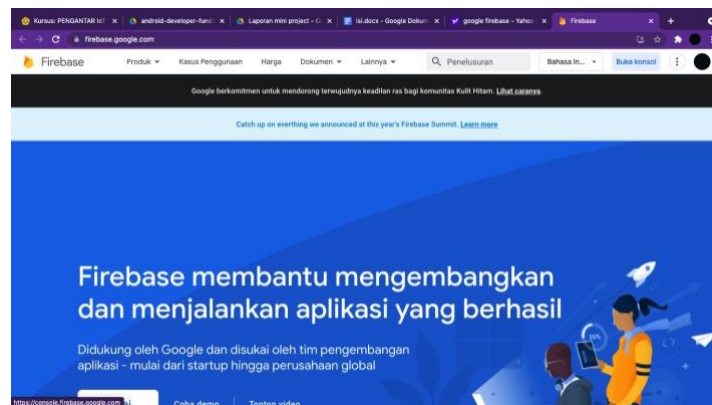
*Gambar 9: Flowchart Google Firebase*

1) Masuk ke Web Firebase dan Login ke halaman Firebase Google



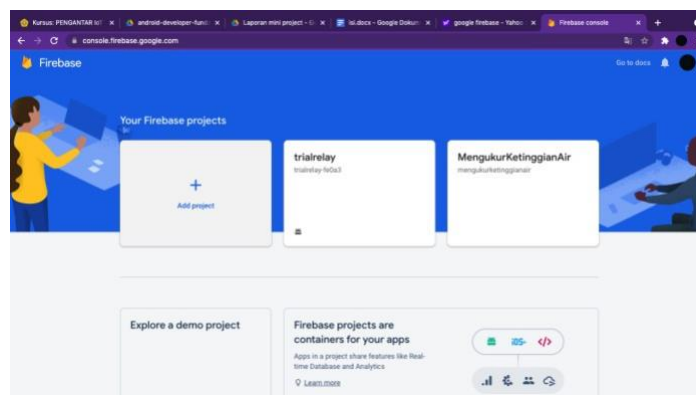
*Gambar 10: Firebase 1*

2) Klik 'Buka Konsol'



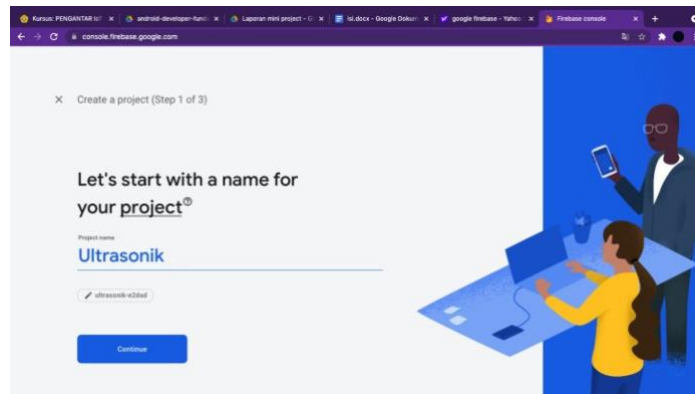
*Gambar 11: Firebase 2*

3) Klik 'add project'



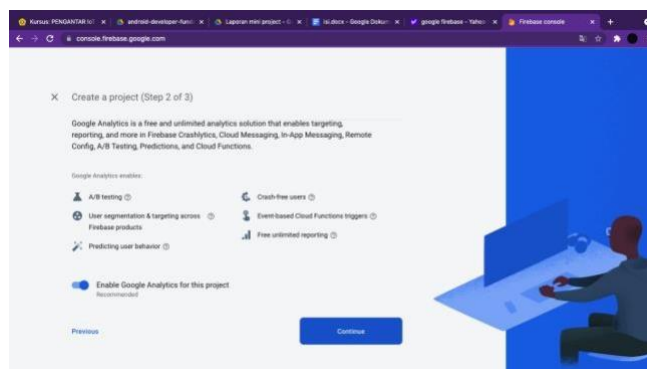
*Gambar 12: Firebase 3*

4) Masukkan Nama project dan klik continue



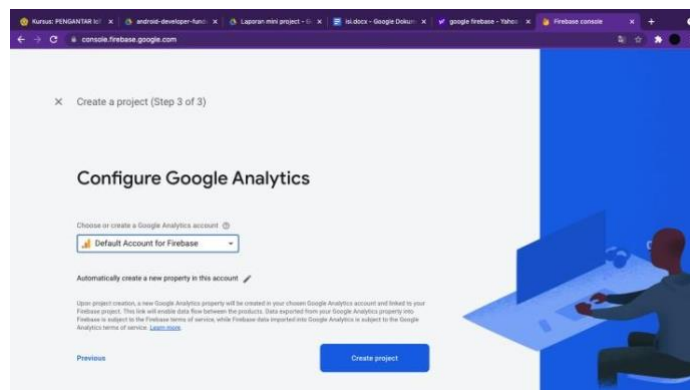
*Gambar 13: Firebase 4*

5) Klik Continue pada step ke-2



*Gambar 14: Firebase 5*

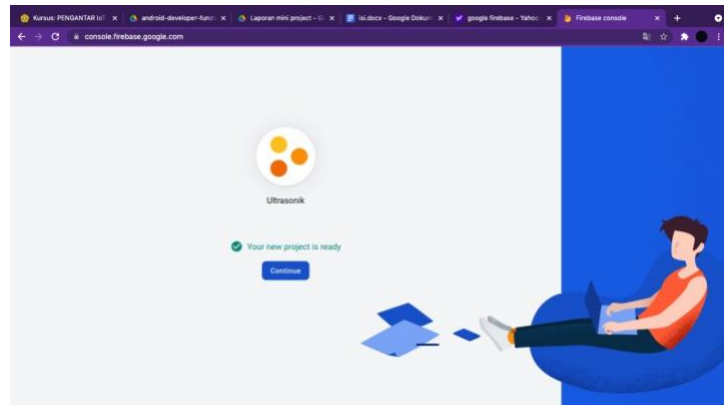
6) Pilih 'Default Account For Firebase' lalu 'create project'



*Gambar 15: Firebase 6*

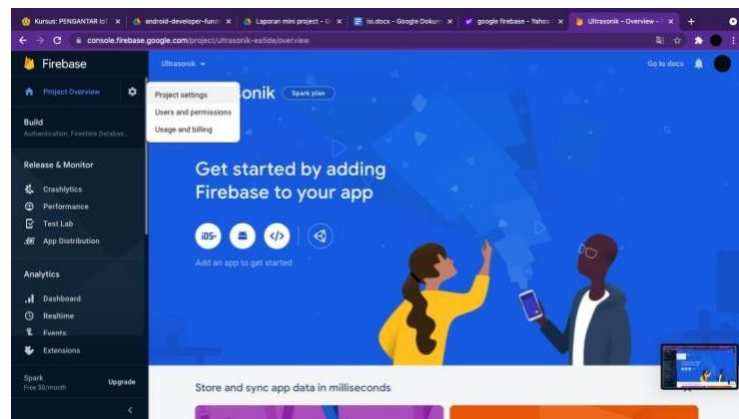


7) Tunggu Sebentar hingga muncul seperti gambar



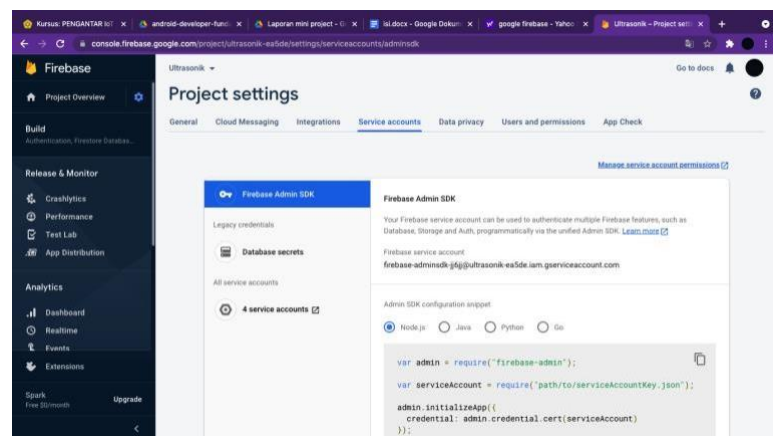
Gambar 16: Firebase 7

8) Klik kanan pada gambar gigi beroda dan klik 'Project Setting'



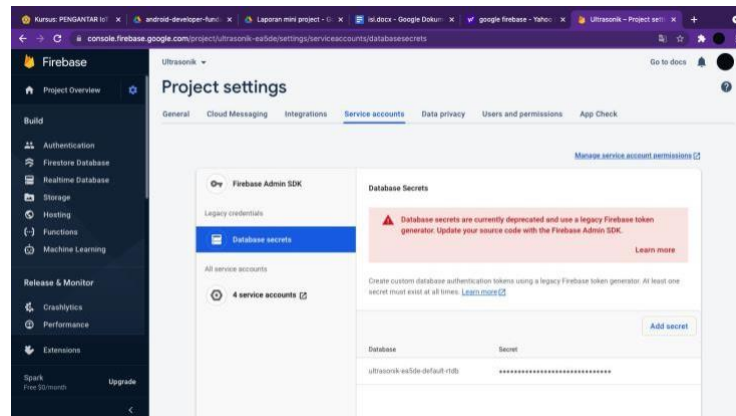
Gambar 17: Firebase 8

9) Pilih menu 'Service Accounts'



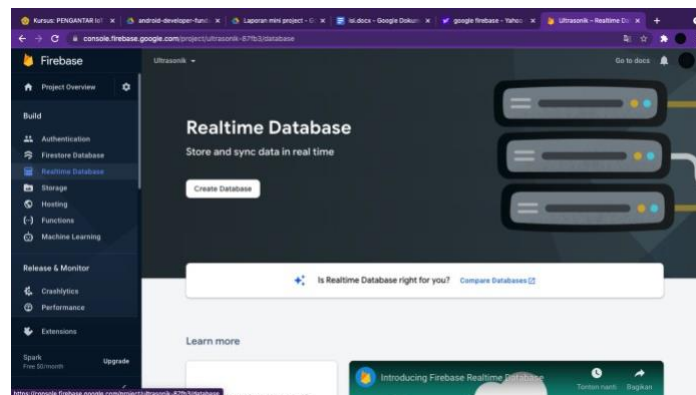
Gambar 18: Firebase 9

- 10) Pilih ‘Database Secrets’ dan tertampil nama Database dan Secret. Simpan Secret karena akan digunakan untuk menyambungkan ke arduino



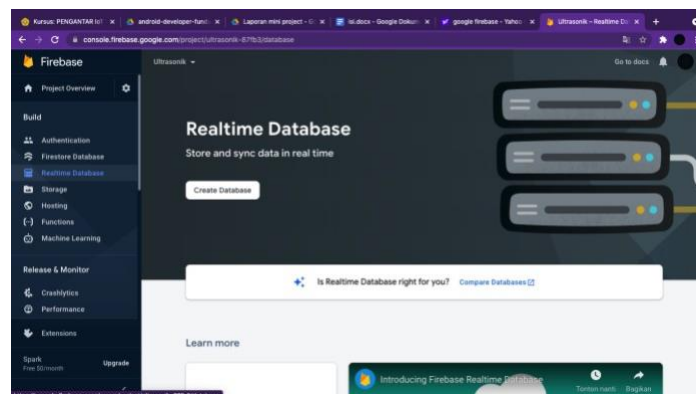
*Gambar 19: Firebase 10*

- 11) Pada bagian ‘build’ di sisi kiri, pilih realtime Database



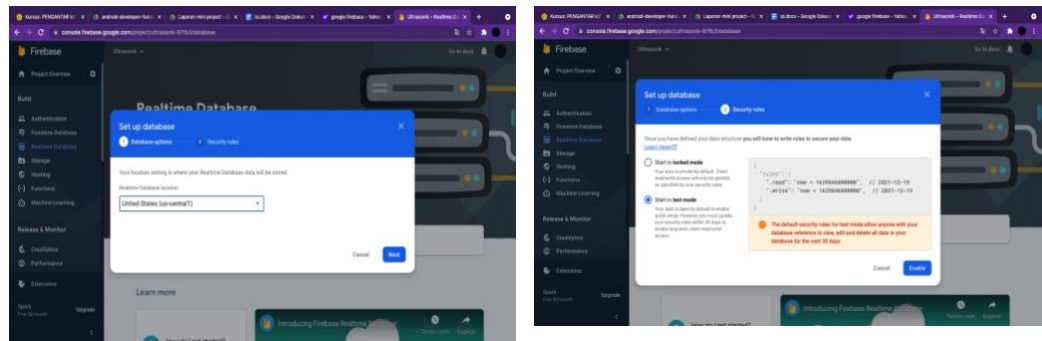
*Gambar 20: Firebase 11*

- 12) Pilih ‘Create Database’



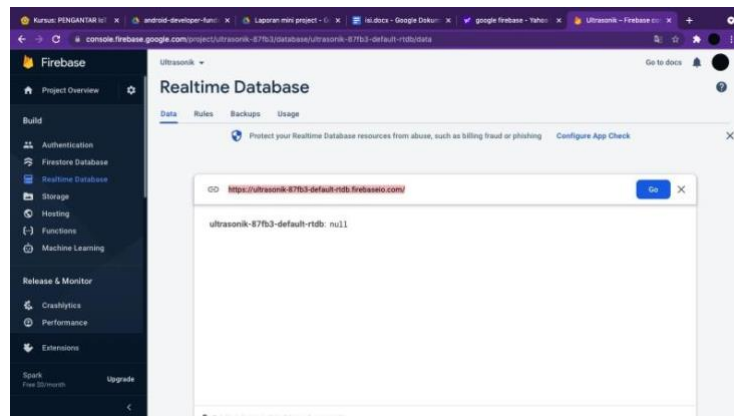
*13) Gambar 21: Firebase 12*

13) Pilih Realtime Database Location dan pilih yang 'start in test mode', lalu klik enable



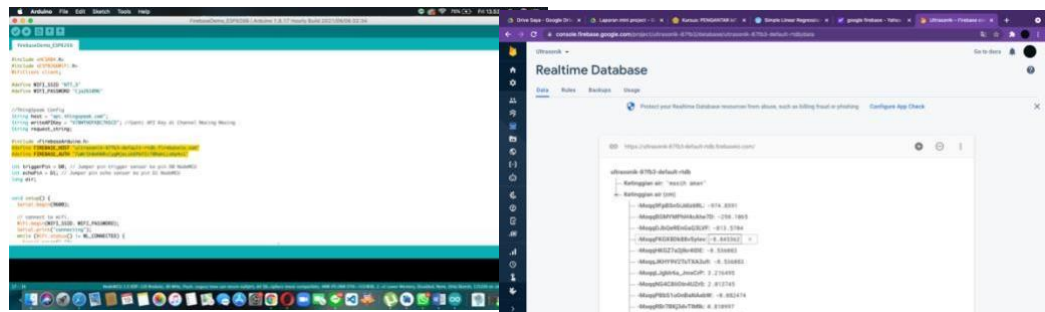
Gambar 22: Firebase 13

14) Akan muncul seperti gambar dan simpan link karena akan digunakan untuk menyambungkan ke arduino



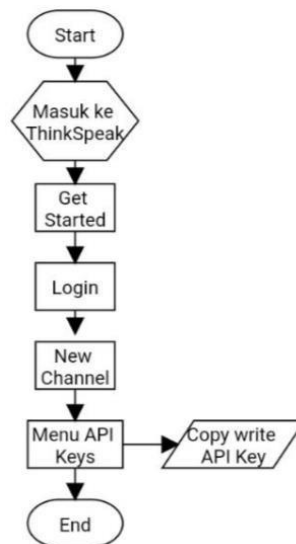
Gambar 23: Firebase 14

15) Salin Keduanya dalam coding di bagian yang di-blok dan jika berhasil maka akan muncul seperti gambar.



Gambar 24: Firebase 15

d. Menyambungkan Ke ThinkSpeak



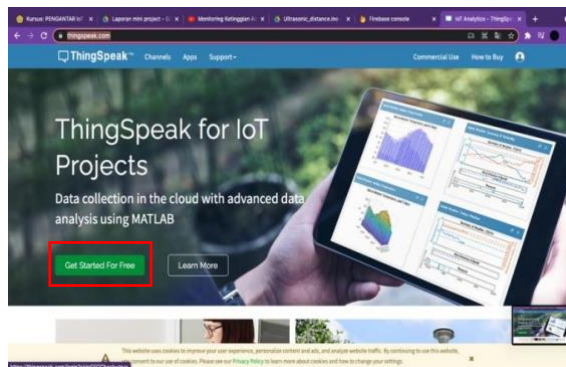
Gambar 25: Flowchart ThinkSpeak

- 1) Buka web <https://thingspeak.com/>



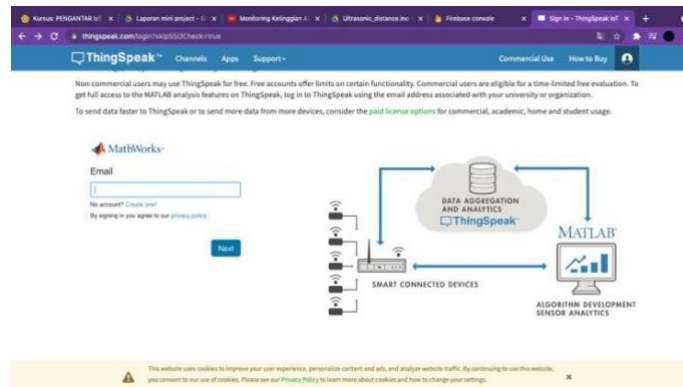
Gambar 26: ThinkSpeak 1

- 2) Klik ‘Get Started For Free’



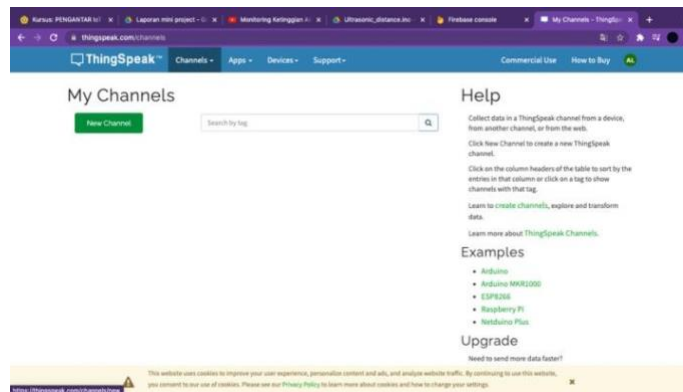
Gambar 27: ThinkSpeak 2

### 3) Login ke ThingSpeak



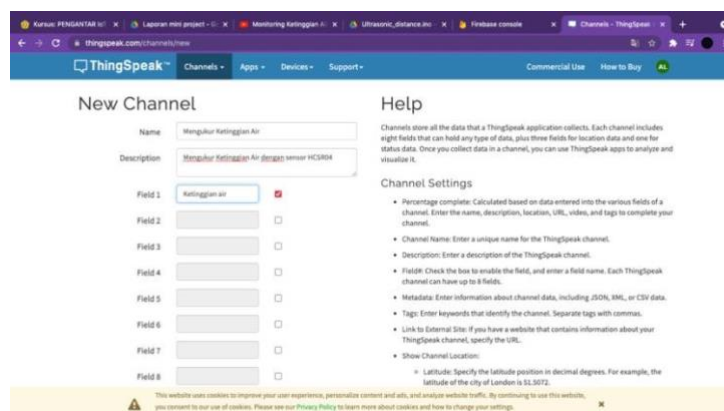
Gambar 28: ThinkSpeak 3

### 4) Klik 'New Channel'



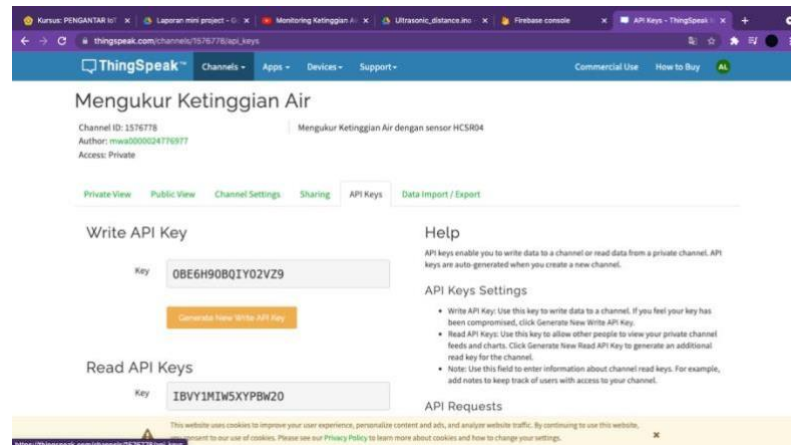
Gambar 29: ThinkSpeak 4

### 5) Masukkan nama proyek, deskriptor, dan juga label yang ingin di-data



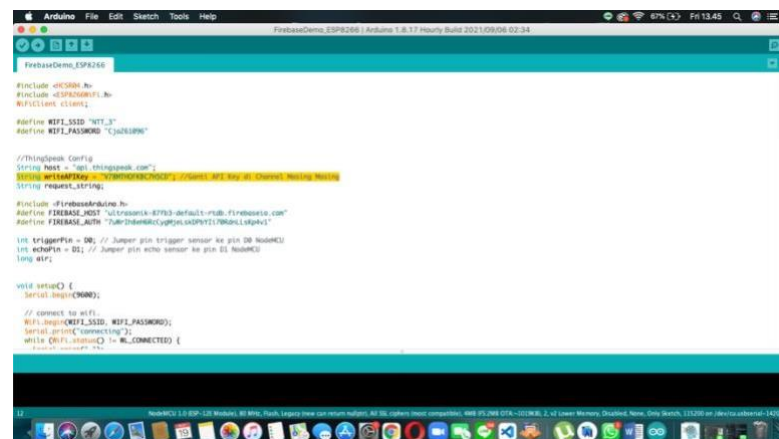
Gambar 30: ThinkSpeak 5

- 6) Masuk ke menu API Keys dan copy Key yang di Write API Key karena akan digunakan untuk menyambungkan ThingSpeak dan arduino



Gambar 31: ThinkSpeak 6

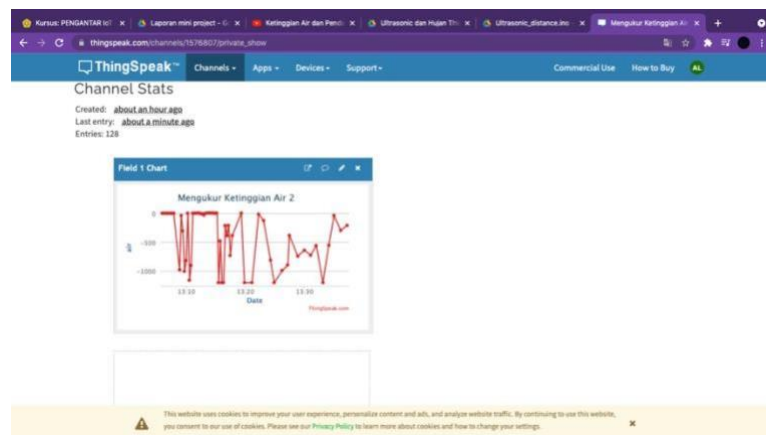
- 7) Paste Key dalam coding di bagian yang di blok pada gambar



Gambar 32: ThinkSpeak 7



8) Jika berhasil, maka data akan divisualisasikan seperti gambar



*Gambar 33: ThinkSpeak 8*

e. Membuat Proyek menjadi Portable

- 1) Sambungkan kabel usb dari esp8266 ke power bank atau baterai atau sumber listrik lainnya. Jika data pembacaan sensor telah tersimpan di google firebase, maka alat proyek telah menjadi portable



*Gambar 34: Portable*

### 3.2 Kategori Data Uji Coba

#### 3.2.1 Kategori 1: Ketika Air Dalam Keadaan Stabil

- a. Ketinggian Air (Diukur Penggaris)
- b. Ketinggian Air (Diukur Sensor)
- c. Apa Buzzer Berbunyi?
- d. Aman atau Bahaya? (google firebase)

#### 3.2.2 Kategori 2: Ketika Air Dalam Keadaan Bergoyang

- a. Ketinggian Air (Diukur Penggaris)
- b. Ketinggian Air (Diukur Sensor)

- c. Apa Buzzer Berbunyi?
- d. Aman atau Bahaya? (google firebase)



#### BAB IV RINCIAN BIAYA

***Tabel 3 Rincian Biaya untuk Pembuatan Sensor Ultrasonic***

No	Nama	Jumlah Satuan	Harga	Nama Toko / Link Toko Online
1	ESP8266	1	Rp36.390	<a href="https://shopee.co.id/product/10630793/2655511030?smtt=0.362482928-1638290179.9">https://shopee.co.id/product/10630793/2655511030?smtt=0.362482928-1638290179.9</a>
2	Kabel Jumper F-F	4	Rp 1.960	<a href="https://shopee.co.id/product/57206583/3652370871?smtt=0.362482928-1638289714.9">https://shopee.co.id/product/57206583/3652370871?smtt=0.362482928-1638289714.9</a>
3	Kabel Jumper M-F	2	Rp 1.000	<a href="https://shopee.co.id/product/55970586/1796468433?smtt=0.362482928-1638289904.9">https://shopee.co.id/product/55970586/1796468433?smtt=0.362482928-1638289904.9</a>
4	Bread Board Mini	1	Rp 4.500	<a href="https://shopee.co.id/product/14183026/6377379076?smtt=0.362482928-1638289865.9">https://shopee.co.id/product/14183026/6377379076?smtt=0.362482928-1638289865.9</a>

5	Sensor Ultrasonik HC-SR04	1	Rp14.500	<a href="https://shopee.co.id/product/57206583/4656130248?smtt=0.362482928-1638285219.9">https://shopee.co.id/product/57206583/4656130248?smtt=0.362482928-1638285219.9</a>
6	Buzzer Mini Active	1	Rp 2.000	<a href="https://shopee.co.id/product/14183026/5815556332?smtt=0.362482928-1638289823.9">https://shopee.co.id/product/14183026/5815556332?smtt=0.362482928-1638289823.9</a>

## BAB V HASIL DAN PEMBAHASAN

### 5.1 Data Pengamatan

#### 5.1.1 Kategori 1: Ketika Air Dalam Keadaan Stabil

*Tabel 4 Percobaan 1 Dengan Air Stabil*

No	Ketinggian air/cm (diukur penggaris)	Ketinggian air/cm (diukur sensor)	Apa buzzer berbunyi?	Aman atau bahaya? (google Firebase)
1.	2	1.41	Tidak	Aman
2.	2.8	2	Tidak	Aman
3.	4.6	4.98	Tidak	Aman
4.	5.7	5.58	Tidak	Aman
5.	7.2	7.34	Ya	Bahaya
6.	8.8	8.86	Ya	Bahaya
7.	9.2	8.26	Ya	Bahaya

*Tabel 5 Percobaan 2 Dengan Air Stabil*

No	Ketinggian air/cm (diukur penggaris)	Ketinggian air.cm (diukur sensor)	Apa buzzer berbunyi?	Aman atau bahaya? (google Firebase)
1.	2	1.7	Tidak	Aman
2.	3.8	3.2	Tidak	Aman
3.	4.8	4.67	Tidak	Aman
4.	5.7	5.27	Tidak	Aman
5.	6.8	7.04	Ya	Bahaya
6.	8.2	7.35	Ya	Bahaya
7.	9.2	8.26	Ya	Bahaya
8.	9.9	7.9	Ya	Bahaya

***Tabel 6 Percobaan 3 Dengan Air Stabil***

<b>No</b>	<b>Ketinggian air/cm (diukur penggaris)</b>	<b>Ketinggian air/cm (diukur sensor)</b>	<b>Apa buzzer berbunyi?</b>	<b>Aman atau bahaya? (google Firebase)</b>
1.	2.1	2.0	Tidak	Aman
2.	3.2	2.9	Tidak	Aman
3.	4.2	4.67	Tidak	Aman
4.	5.8	6.18	Tidak	Aman
5.	6.4	6.48	Tidak	Aman
6.	7.8	7.06	Ya	Bahaya
7.	9.3	6.48	Tidak	Aman

5.1.2. Kategori 2: Ketika Air Dalam Keadaan Bergoyang

***Tabel 7 Percobaan 1 Dengan Air yang Bergoyang***

<b>No</b>	<b>Ketinggian air/cm (diukur penggaris)</b>	<b>Ketinggian air/cm (diukur sensor)</b>	<b>Apa buzzer berbunyi?</b>	<b>Aman atau bahaya? (google Firebase)</b>
1.	2.2	1.41 – 1.99	Tidak	Aman
2.	3.5	2.3 – 3.5	Tidak	Aman
3.	4.5	3.8	Tidak	Aman
4.	6.1	5.62 – 5.8	Tidak	Aman
5.	7	6.7	Tidak	Aman
6.	8	6.18 – 6.4	Tidak	Aman
7.	9.2	7.6 – 7.8	Ya	Bahaya

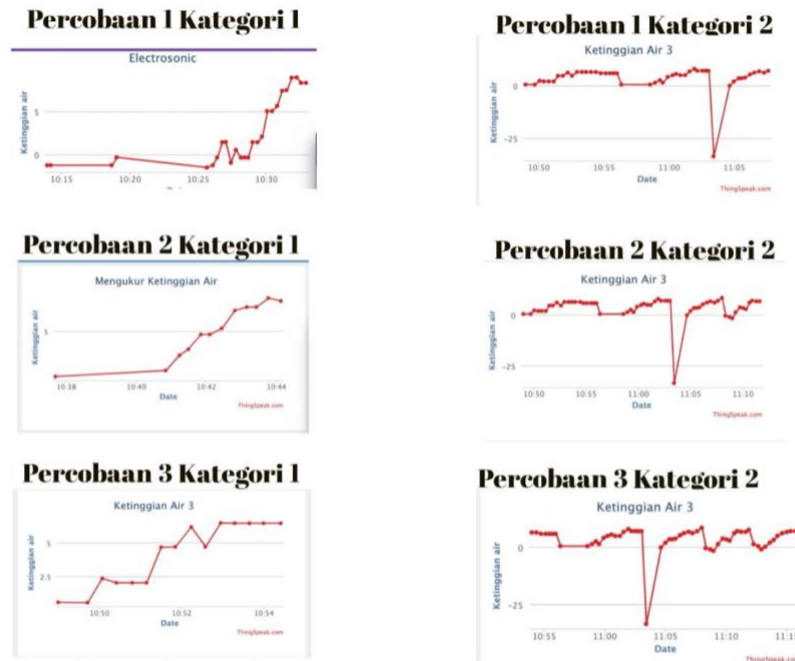
***Tabel 8 Percobaan 2 Dengan Air yang Bergoyang***

<b>No</b>	<b>Ketinggian air/cm (diukur penggaris)</b>	<b>Ketinggian air/cm (diukur sensor)</b>	<b>Apa buzzer berbunyi?</b>	<b>Aman atau bahaya? (google Firebase)</b>
1.	2.2	1.4 – 2.3	Tidak	Aman
2.	4.2	3.8	Tidak	Aman
3.	5.1	4.3 – 4.9	Tidak	Aman
4.	6.8	6.1 – 7.09	Tidak/Ya	Aman/bahaya
5.	8.5	6.79	Tidak	Aman
6.	9.7	7.6 – 7.9	Ya	Bahaya

***Tabel 9 Percobaan 3 Dengan Air yang Bergoyang***

<b>No</b>	<b>Ketinggian air/cm (diukur penggaris)</b>	<b>Ketinggian air/cm (diukur sensor)</b>	<b>Apa buzzer berbunyi?</b>	<b>Aman atau bahaya? (google Firebase)</b>
1.	2.1	2.83 – 3.2	Tidak	Aman
2.	3.4	4.98 – 5.0	Tidak	Aman
3.	4.5	6.1 – 6.7	Tidak	Aman
4.	7	7.04 – 7.08	Ya	Bahaya
5.	8	7.8 – 8.2	Ya	Bahaya
6.	9.5	7.6 – 7.7	Ya	Bahaya

## 5.2 Visualisasi Data Pengamatan



*Gambar 35 : Visualisasi Data Pengamatan*

## 5.3 Analisis Data dan Insight

### 5.3.1 Sistem Peringatan Dini

Buzzer akan berbunyi jika ketinggian mencapai lebih dari 7 cm dan suara akan terus dibunyikan selama ketinggian berada di atas 7 cm. Namun, karena sensor memiliki blinspot yaitu tidak bisa membaca kurang dari 2 cm, maka buzzer berhenti pada ketinggian 9 cm. Pada google firebase juga akan tertampil golongan aman atau bahaya tergantung pada ketinggian airnya. Jika ketinggian air berada dalam rentang dibawah 7 cm maka dikatakan golongan aman. Jika ketinggian air berada dalam ketinggian diatas 7 cm maka dikatakan golongan bahaya yang menandakan ketinggian air sudah mendekati ketinggian wadah.

Dengan penggunaan Buzzer dan muncul tulisan “Bahaya” di google Firebase sebagai peringatan dini ketika ketinggian air berada lebih dari 7 cm diharapkan agar masyarakat sekitar daerah perairan segera mengungsi sebelum banjir melanda.

### 5.3.2 Keakuratan

$$Keakuratan (persen) = 100 - |galat|$$

$$galat = \frac{Selisih\ Nilai}{Nilai\ aktual} \times 100$$

Dari ketiga tabel kategori pertama yaitu keadaan air stabil, tabel pertama memiliki rata – rata keakuratan 88.38%. Tabel kedua memiliki rata – rata keakuratan 78.08% dan tabel ketiga memiliki rata – rata keakuratan 89.58% Sehingga dari rata – rata tersebut menghasilkan rata – rata keakuratan 85.34 %.

Sementara dari dari ketiga tabel kategori kedua yaitu keadaan air bergoyang, tabel pertama memiliki rata – rata keakuratan 85.17% Tabel kedua memiliki rata – rata keakuratan... dan tabel ketiga memiliki rata – rata keakuratan 86.92% Sehingga dari rata – rata tersebut menghasilkan rata – rata keakuratan 74.51% yang berarti menunjukkan keakuratannya sebesar 82.2 %.

Terlihat ketika air dalam suatu wadah bergerak maka keakuratan pembacaan sensor berkurang meskipun tidak terlalu signifikan. Selain itu, sensor tidak bisa menghasilkan nilai yang stabil dan menghasilkan nilai interval. Berdasarkan data pengamatan, pembacaan selisih yang dihasilkan  $\pm 0.3 - 0.8$  cm dan blindspot sensor mulai pada ketinggian lebih dari 9 cm.

### 5.3.3 Kelebihan

Terdapat berbagai kelebihan dalam menggunakan alat ini dalam pembacaan ketinggian air, diantaranya:

- Pembacaan dilakukan secara otomatis oleh alat sehingga tidak memerlukan bantuan manual dari manusia.
- Pembacaan dilakukan 24 jam dengan syarat memiliki daya yang memadai.
- Data bisa disimpan secara otomatis dalam google firebase serta otomatis dapat ditampilkan dalam grafik di ThinkSpeak
- Terdapat buzzer dan tulisan “bahaya” yang menjadi peringatan dini jika ketinggian sudah mendekati ketinggian wadah.

### 5.3.4 Kekurangan

Namun, tentu saja alat ini belum sempurna sehingga memiliki beberapa kekurangan, diantaranya:

- Sensor memiliki blindspot pada jarak 1 – 2 cm sehingga ketika mencapai ketinggian tertentu, pembacaan sensor tidak dapat dilakukan.
- Meskipun tidak besar, namun alat ini membutuhkan daya dan internet untuk pengoperasian sensor sehingga membutuhkan biaya tambahan. Selain itu, untuk daerah yang kurang menjangkau internet, belum bisa memanfaatkan teknologi IoT ini.
- Jika air dalam wadah tidak stabil maka pembacaan ketinggian juga kurang sesuai. Hasil data yang stabil membutuhkan proses dan waktu.



#### 5.3.5 Perbandingan dengan Jurnal Lainnya.

Adapun berbagai perbedaan signifikan dari proyek jurnal yang menjadi tinjauan pustaka, diantaranya:

- Proyek kami melakukan pembacaan data secara portable, sementara pada proyek jurnal belum dilaksanakan.
- Proyek kami melakukan penyimpanan data otomatis dan disimpan dalam google firebase, sementara pada proyek jurnal hanya ditampilkan menggunakan LCD.
- Proyek kami melakukan visualisasi data dalam bentuk linechart di website ThinkSpeak, sementara pada proyek jurnal tidak dilakukan.
- Kami menggunakan air keruh dengan konsentrasi 0.025 g/ml sebagai simulasi air banjir namun, dalam proyek jurnal menggunakan air bersih. Diharapkan dengan air keruh sebagai simulasi akan menghasilkan keakuratan yang lebih.
- Proyek kami hanya menggolongkan ketinggian dalam kategori aman atau bahaya namun, pada proyek jurnal menggolongkan ketinggian dalam kategori aman, siaga, awas, dan bahaya.

## BAB VI PENUTUP

### 6.1.Simpulan

Sensor ultrasonik terbukti bisa mengukur ketinggian air yang mana sensor ini bisa diterapkan sebagai langkah preventif banjir. Ketika suatu level ketinggian air mendekati ketinggian wadah, langkah preventif segera dilakukan agar banjir bisa dicegah dan melihat apakah langkah preventif yang dilakukan telah berhasil menurunkan level ketinggian air. Selain itu dengan pencatatan data banjir, bisa terlihat daerah mana yang lebih rawan banjir sehingga bisa lebih difokuskan untuk pencegahan di daerah tersebut.

Keakuratan pembacaan sensor adalah sebesar 85.34 % meskipun ini hanya perkiraan sementara. Buzzer yang digunakan bisa menjadi penanda bahwa ketinggian air hampir mendekati ketinggian wadah sehingga warga sekitar daerah perairan bisa segera mengungsi dan meminimalisir dampak banjir. Meskipun begitu, masih ada beberapa kekurangan yang perlu ditingkatkan kembali.

### 6.2.Saran

Berdasarkan hasil analisis yang dilakukan, peneliti menyarankan untuk menempatkan sensor ultrasonik dengan keadaan yang stabil dan tepat karena sensor ini memiliki titik buta. Dengan penempatan yang stabil dan tepat maka akan membuat pembacaan ketinggian air berjalan dengan baik dan menghindari resiko yang tidak diinginkan.

### 6.3.Potensi Pengembangan dan Penelitian Alat di Masa Depan

Alat ini memiliki potensi untuk dikembangkan lebih lanjut lagi di Masa Depan diantaranya bisa menambahkan LED yang akan menyala pada setiap ketinggian tertentu sebagai sistem peringatan dini, pembacaan sensor lebih diakuratkan lagi untuk mendapatkan hasil yang maksimal, dan menggolongkan ketinggian air ke dalam empat kategori untuk semakin akurat.

## DAFTAR PUSTAKA

- 1) Bernandus, Johnshon Tarigan, & Tanseib, Jehunias Leonidas. 2019. *Perancangan Sistem Pendeteksi Banjir Menggunakan Sensor HC-SR04 Berbasis Arduino Uno*. Jurnal Biotropikal Sains. 6-8.
- 2) Chatterjee, Nilanjan dkk. *Real-Time Communication Application Based on Android Using Google Firebase*. International Journal of Advance Research in Computer Science and Management Studies. 74.
- 3) Diy ALC. (2019). "Cara Menyimpan Data Sensor Secara Real Time ke Database Firebase Menggunakan ESP8266". Indonesia: [www.youtube.com](http://www.youtube.com). Diambil Kembali dari: [https://youtu.be/bzy\\_NuDm0UY](https://youtu.be/bzy_NuDm0UY).
- 4) Domi Burman. (2021). Pembuatan Alat Monitoring Ketinggian Air Berbasis IOT | Internet of Things. Indonesia: [www.youtube.com](http://www.youtube.com). Retrieved from: <https://youtu.be/wGgQJpHUszM>.
- 5) Hiyang Zelika. (2021). Mengukur Ketinggian Air dengan Sensor Ultrasonik Menggunakan ESP8266. Indonesia: [www.youtube.com](http://www.youtube.com). Retrieved from: <https://youtu.be/EixJE8nYCI4>.
- 6) Kelas Robot. (2021). #LangsungPraktek menghubungkan NodeMCU ESP8266 ke Firebase – Kirim Terima Data. Indonesia: [www.youtube.com](http://www.youtube.com). Retrieved from: <https://youtu.be/W4Hn3pi4mNc>.
- 7) LawLaw ELC. (2021). Ketinggian Air dan Pendeteksi Hujan Untuk Alarm Banjir Berbasis Thingspeak IoT || Tutorial Full Code. Indonesia: [www.youtube.com](http://www.youtube.com). Retrieved from: <https://youtu.be/NHWugl5NCH8>.
- 8) OTW Skripsi. (2021). Skripsi Rancang Bangun Monitoring dan Deteksi Banjir Berbasis Iot – nodeMCU Ultrasonik HCSR04 Blynk. Indonesia: [www.youtube.com](http://www.youtube.com). Retrieved from: <https://youtu.be/Lv3iRN-GXhI>.
- 9) Pasha, Sharmad. (2016). *ThinkSpeak Based Sensing and Monitoring System for IoT with Matlab Analysis*. International Journal of New Technology and Research. 19
- 10) Risdiandi, Rahmat. (2017). *Analisis Cara Kerja Sensor Ultrasonik Menggunakan Mikrokontroler Arduino Uno Untuk Merancang Alat Deteksi Banjir Secara Otomatis*. ResearchGate.

- 11) Tech Vegan. (2020). How to Interface IR Sensor with NodeMCU ESP8266 | Obstacle Detector With Buzzer & Arduino Code. Indonesia: [www.youtube.com](https://www.youtube.com/watch?v=kyuo0JpZtBE). Retrieved from: <https://youtu.be/kyuo0JpZtBE>
- 12) Umari, C., Anggraini, E., & Zainul Muttaqinm Rofif. (2017). *Rancang Bangun Sistem Peringatan Dini Banjir Berbasis Sensor Ultrasonik Dan Mikrokontroler Sebagai Upaya Penanggulangan Banjir*. Jurnal Meteorologi Klimatologi Dan Geofisika, 4(2), 35–42.

## LAMPIRAN

### Lampiran 1. Dokumentasi Kegiatan



*Gambar 36 : Proyek Dari Depan*



*Gambar 37 : Proyek Dari Samping*

## Lampiran 2. Source Code Program

```
#include <HCSR04.h>
#include <ESP8266WiFi.h>
WiFiClient client;

#define WIFI_SSID "NTT_3" // nama wifi/hotspot
#define WIFI_PASSWORD "Cja261096" // password wifi/hotspot

//ThingSpeak Config
String host = "api.thingspeak.com"; // connect ke ThingSpeak
String writeAPIKey = "4S5R789LENOHWOTL"; //Ganti API Key di
Channel Masing Masing
String request_string;

#include <FirebaseArduino.h>
#define FIREBASE_HOST "ultrasonik-ea5de-default-
rtadb.firebaseio.com" //web sesuai masing - masing
#define FIREBASE_AUTH
"2u4z7Faip99B69dsj6T8HnOmRZRiV3MIKOxu075c" //secret sesuai
masing - masing

int triggerPin = D0; // Jumper pin trigger sensor ke pin D0
NodeMCU
int echoPin = D1; // Jumper pin echo sensor ke pin D1 NodeMCU
int buzzer = D6; //Jumper pin buzzer ke pin D6 NodeMCU
long air;

void setup() {
  Serial.begin(9600);

  // connect to wifi.
  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  Serial.print("connecting");
  while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");
    delay(500);
  }
  Serial.println();
  Serial.print("connected: ");
  Serial.println(WiFi.localIP());

  //connect to firebase
```

```

    //pinmode
    pinMode(triggerPin, OUTPUT);
    pinMode(echoPin, INPUT);
    pinMode(buzzer, OUTPUT);}

void loop() {

    // Menghitung ketinggian air
    float duration, ketinggian, air;
    digitalWrite(triggerPin, LOW);
    delayMicroseconds(2);          // delay tiap pengukuran (bisa
    diset sendiri)
    digitalWrite(triggerPin, HIGH);
    delayMicroseconds(10);         // delay tiap pengukuran (bisa
    diset sendiri)
    digitalWrite(triggerPin, LOW);
    duration = pulseIn(echoPin, HIGH);
    ketinggian = (duration/2) / 29.1;
    air = 11.00-ketinggian;
    Serial.println("ketinggian air :"); //kirim tulisan
    'Ketinggian air :' ke serial monitor
    Serial.print(air); //kirim tulisan 'air' ke serial monitor
    Serial.println(" cm"); //kirim tulisan 'cm' ke serial
    monitor
    delay(400);

    // Mengirim pembacaan sensor ke firebase
    Firebase.pushFloat("/Ketinggian air (cm)/", air);

    if (air < 0) {
        Firebase.setString("Ketinggian air/", "ERROR");
    }
    if (air <= 7.00) {
        Firebase.setString("Ketinggian air/", "masih aman");
    }
    if (air > 7.00) {
        Firebase.setString("Ketinggian air/", "TERLALU BERLEBIH,
        BAHAYA!!!");
    }
}

```

```

// handle error
if (Firebase.failed()) {
    Serial.print("setting /number failed:");
    Serial.println(Firebase.error());
    return;
}
delay(100);

// BUZZER
if (air <= 7.00){
    digitalWrite(buzzer,LOW);
}
else if (air>7.00){
    digitalWrite(buzzer,HIGH);
}

// Mengirim ke ThinkSpeak
if (client.connect(host, 80))
{
    request_string = "/update?key=" + writeAPIKey
                    + "&field1=" + air;

    Serial.println(String("GET ") + request_string + "
HTTP/1.1\r\n" +
                    "Host: " + host + "\r\n" +
                    "Connection: close\r\n\r\n");

    client.print(String("GET ") + request_string + "
HTTP/1.1\r\n" +
                    "Host: " + host + "\r\n" +
                    "Connection: close\r\n\r\n");
    unsigned long timeout = millis();

    while (client.available() == 0)
    {
        if (millis() - timeout > 5000)
        {
            Serial.println(">>> Client Timeout !");
            client.stop();
            return;
        }
    }
}

```



```
while (client.available())
{
    String line = client.readStringUntil('\r');
    Serial.print(line);
}

Serial.println();
Serial.println("Closing connection");
}

}
```