

USE CASE PROJECT – GROUP C

# *Smart Fishery 2022 (Control and Monitoring)*

Digitalent Scholarship Professional Academy

[indobot.co.id](https://indobot.co.id)

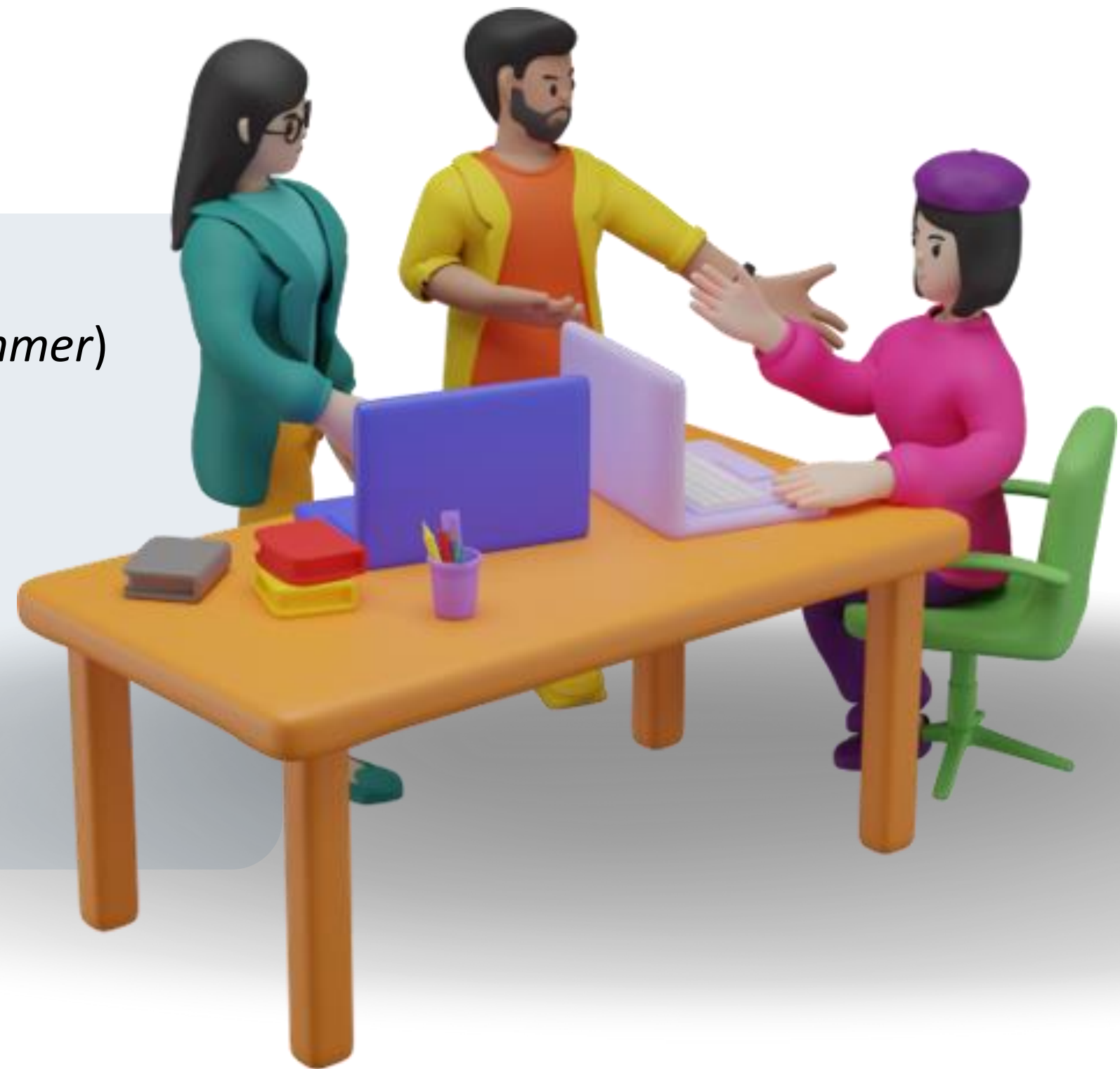
**Isi dan elemen dari video ini memiliki hak kekayaan intelektual yang dilindungi oleh undang-undang**

**Dilarang menggunakan, merubah, memperbanyak, dan mendistribusikan video ini untuk tujuan komersil.**

# ANGGOTA KELOMPOK & *JOBDESK*

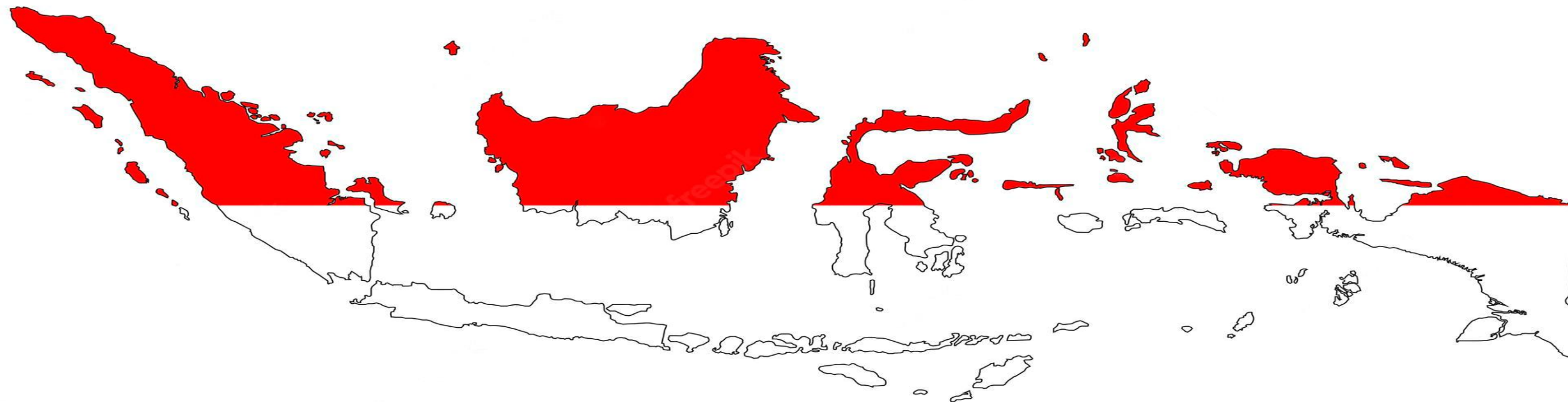
**Kelas: IoT4 - 12**

1. Teguh Pratama Nugraha (*Project Leader & Programmer*)
2. Octavianus E. W. Modami (*Project Manager*)
3. Devan Cakra Mudra Wijaya (*Programmer*)
4. Hariansyah (*Support*)
5. Rohmat Fajar Agustian (*Support*)



# Latar Belakang *Project*

Wilayah Indonesia lebih besar perairannya daripada daratan. Hal tersebut menunjukkan adanya potensi yang besar pada sektor perikanan. Bagi peternak ikan, peralatan budidaya konvensional dinilai kurang efisien, karena harus diaplikasikan secara langsung di lokasi budidaya, sedangkan yang dibutuhkan adalah kemudahan aksi. Pada penelitian ini, kami berupaya untuk memberikan inovasi berupa *control* dan *monitoring* IoT secara *real-time*, yang mana tujuannya demi menyukseskan perkembangan budidaya ikan di Indonesia. Parameter yang perlu diotomatisasi berupa: perlakuan pakan, suhu, dan kekeruhan pada air. Berdasarkan hasil penelitian, menunjukkan aplikasi yang dibuat ini telah berhasil dijalankan.





# Tujuan *Project*

- Mengotomatisasi proses kerja rutin dalam pemberian pakan ikan.
- Memantau jumlah pakan ikan yang tersisa secara *real-time*.
- Memantau kualitas air secara *real-time* : Suhu dan Kekeruhan.
- Melakukan pengendalian jarak jauh pada suhu air jika kondisinya ekstrim (terlalu rendah/tinggi).

# Manfaat *Project*

- Dapat dikerjakan dan dipantau baik kapanpun serta dimanapun berada.
- Menghemat waktu dan biaya pengoperasian.
- Data yang diperoleh bersifat aktual karena didapatkan dari sensor langsung.
- Dapat mencapai efisiensi kerja dan mendukung produktifitas yang baik.

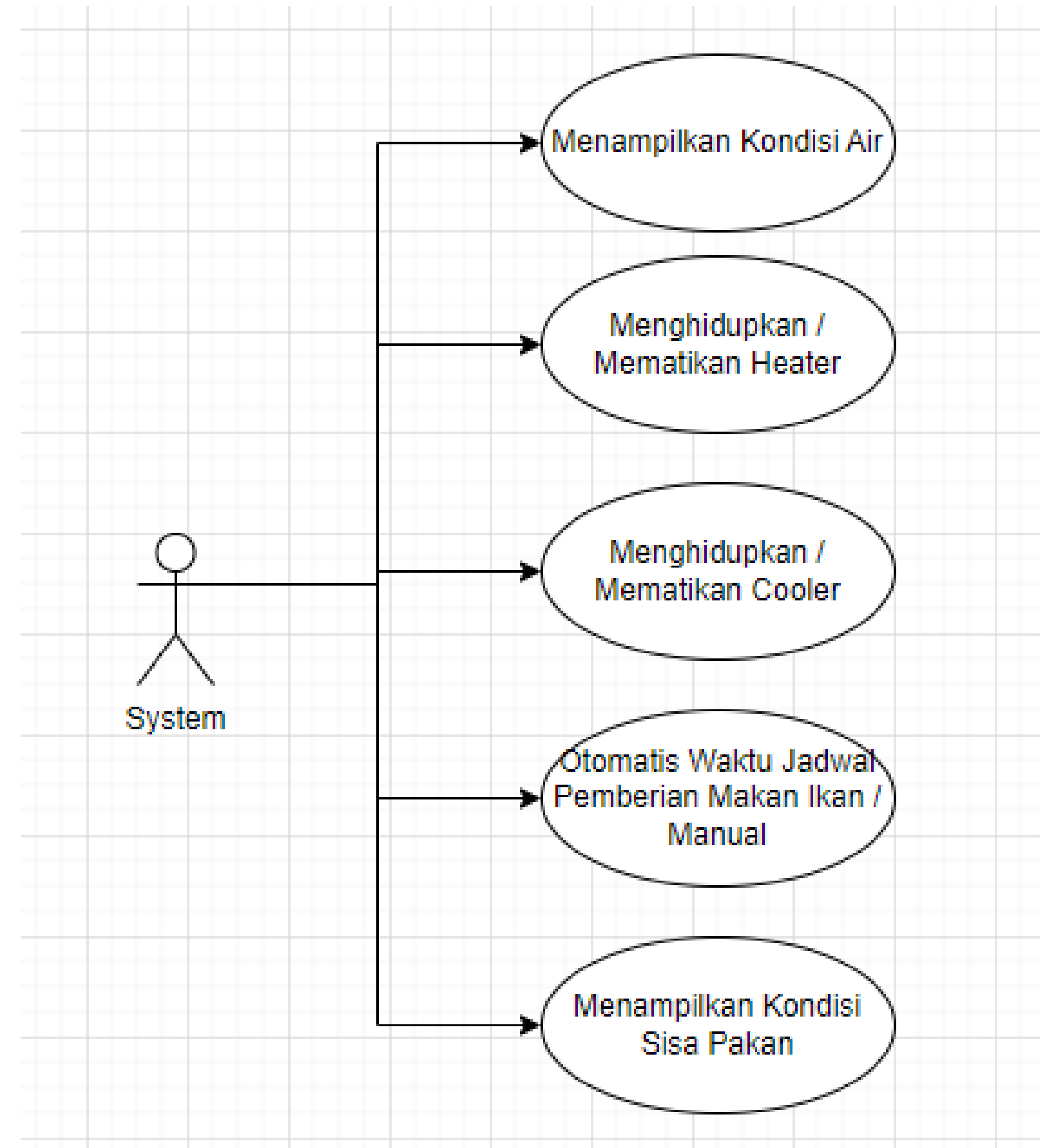
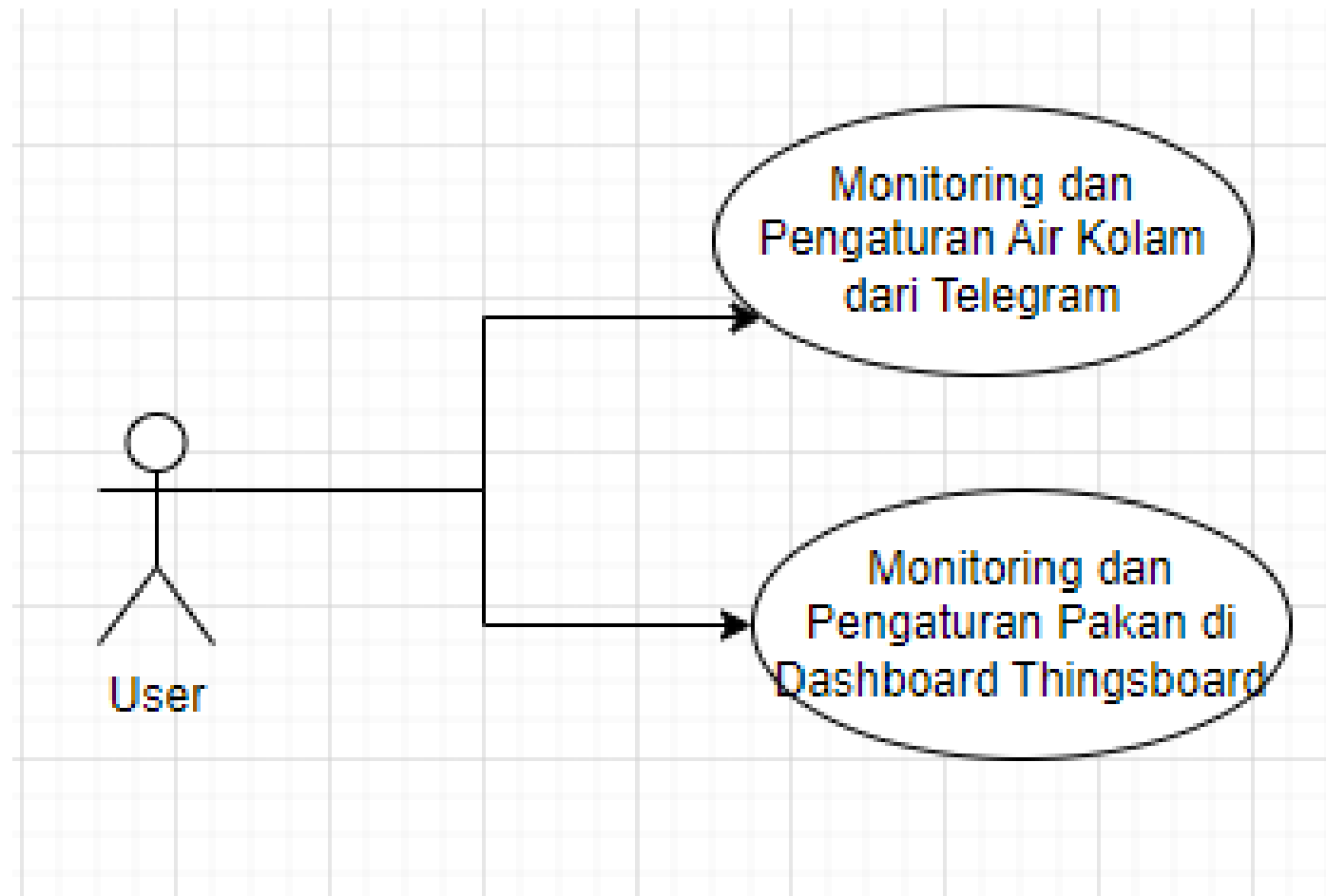
# Analisis Kebutuhan



| NO | KEBUTUHAN HARDWARE  | JUMLAH |
|----|---|--------|
| 1  | Modul ESP32 (Mikrokontroler)  | 2 Buah |
| 2  | <i>Rotatory Encoder</i> (Sensor)  | 1 Buah |
| 3  | RTC (Sensor)  | 2 Buah |
| 4  | DHT22 (Sensor)  | 1 Buah |
| 5  | <i>Ultrasonic</i> HC-SR04 (Sensor)  | 1 Buah |
| 6  | <i>Motor Servo</i> (Aktuator)   | 1 Buah |
| 7  | <i>Photoresistor</i> : Dimisalkan untuk Kejernihan Air atau <i>Turbidity</i> (Sensor) | 1 Buah |
| 8  | LED Kuning : Dimisalkan untuk <i>Heater</i> (Aktuator)                                | 1 Buah |
| 9  | LED Hijau : Dimisalkan untuk <i>Cooler</i> (Aktuator)                                 | 1 Buah |
| 10 | LED Merah : Indikator Bahaya (Aktuator)   | 1 Buah |
| NO | KEBUTUHAN SOFTWARE  |        |
| 1  | Wokwi (Aplikasi: Simulator)   |        |
| 2  | <i>Thingsboard</i> (Aplikasi: Visualisasi Data & Media Interaksi)                     |        |
| 3  | <i>Bot Telegram</i> (Aplikasi: Media Interaksi)                                       |        |

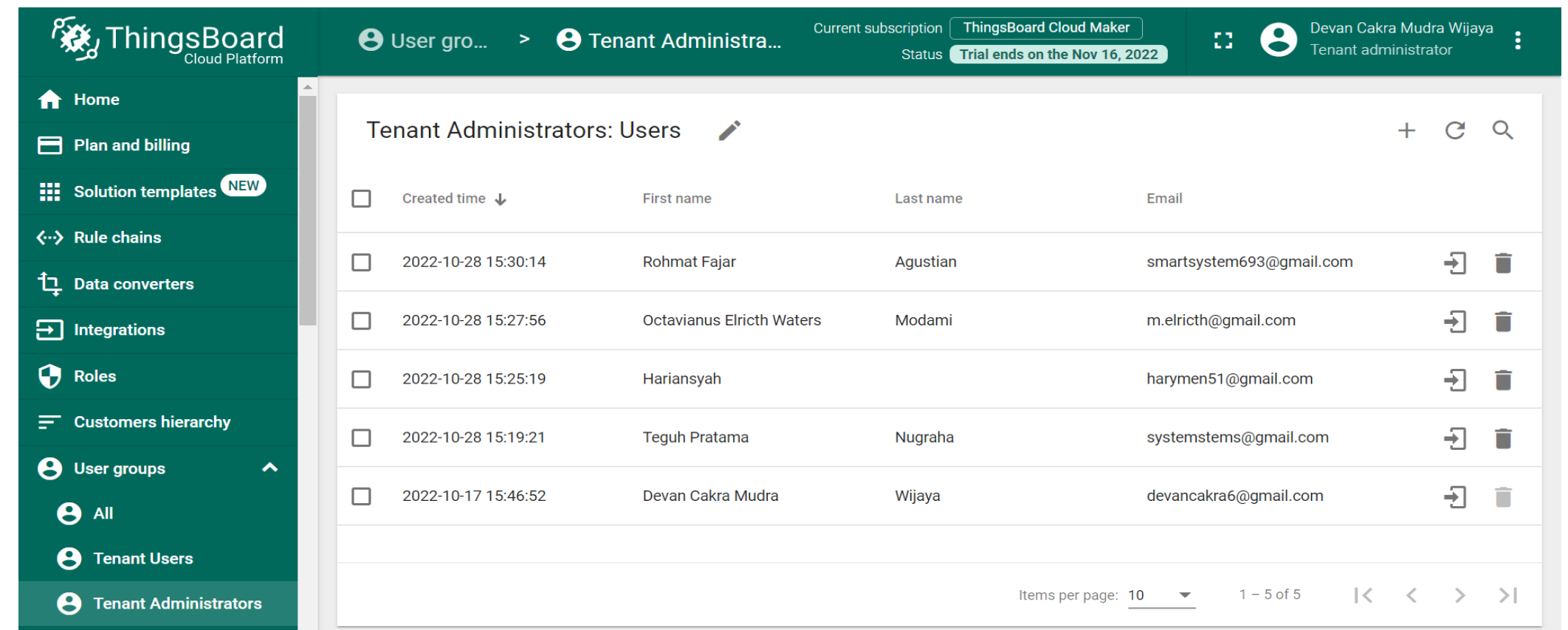


# Work Flow Diagram



# Cara Kami Kolaborasi di *ThingsBoard*

- Memilih salah satu *ThingsBoard* rekan yang ingin dijadikan tempat kolaborasi.
- Pemilik *ThingsBoard* tersebut harus *login* terlebih dahulu.
- Lalu masuk ke bagian *Users Groups* → pilih *Tenant Administrator* → *Add User* dengan *email* yang belum terdaftar di *ThingsBoard*.
- Pastikan bagian/menu telah dibuat *public* semua yaitu dengan cara → *Make entity group public*.
- Melanjutkan pengerjaan kelompok di bagian *device*, *profile*, *dashboard*, dan lain sebagainya.



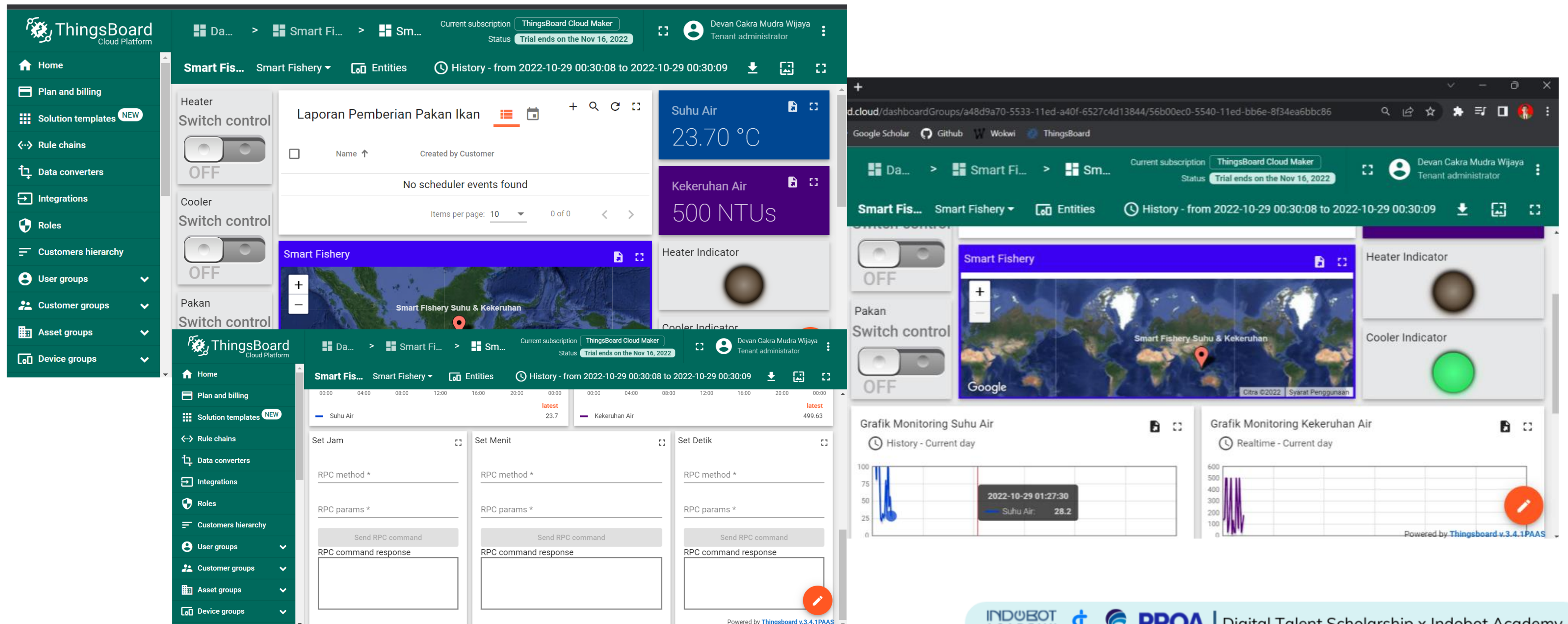
## Kategori Akses :

- Sebagai *Tenant Administrator*, anda dapat mengakses semuanya.
- Sebagai *Tenant Users*, anda dapat menindaklanjuti aksi yang dibuat oleh *Tenant Administrator*.
- Sebagai *Customer Administrator*, anda dapat mengatur hak akses pengguna dalam lingkup yang kecil.
- Sebagai *Customer Tenant*, anda hanya akan mendapatkan akses yang terbatas.



# Dokumentasi *Project #1*

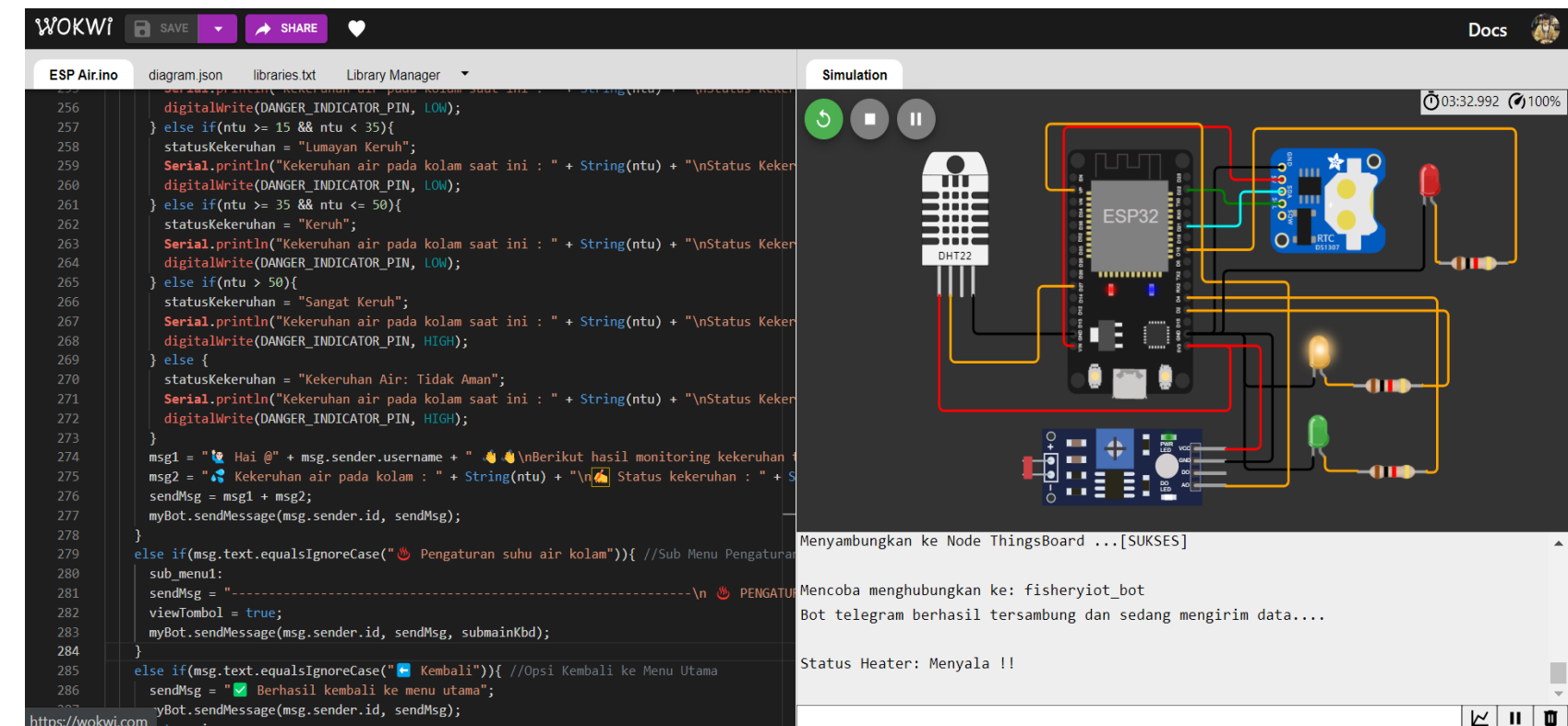
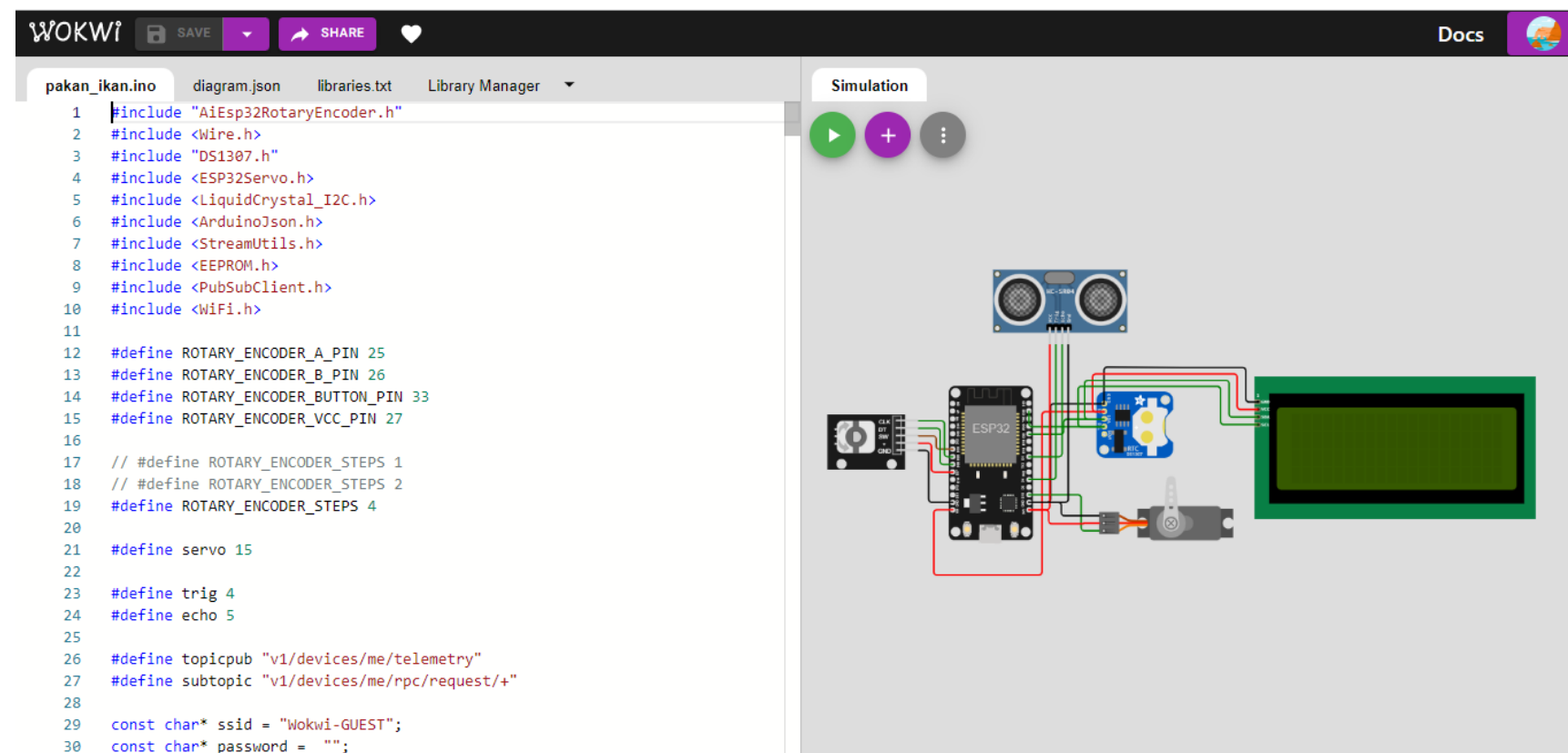
Hasil dokumentasi dari pengerjaan kami sekelompok terkait visualisasi *dashboard* yang ada di *platform ThingsBoard* dapat anda lihat sebagai berikut :



# Dokumentasi *Project #2*

Hasil dokumentasi dari pengerjaan kami sekelompok terkait wokwi 2 *device*, yaitu untuk ESP pakan dan ESP air dapat anda lihat sebagai berikut :

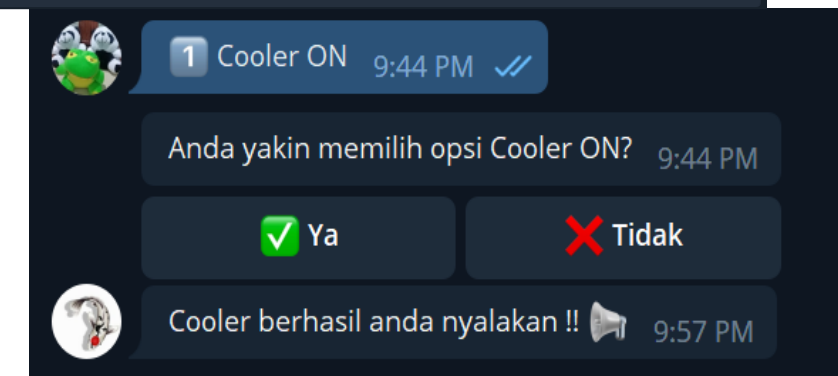
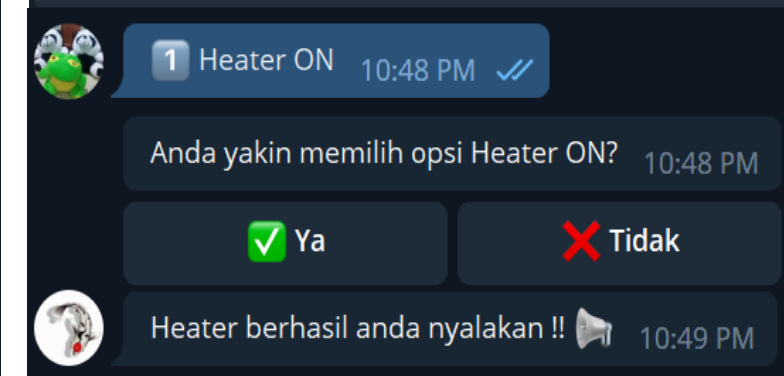
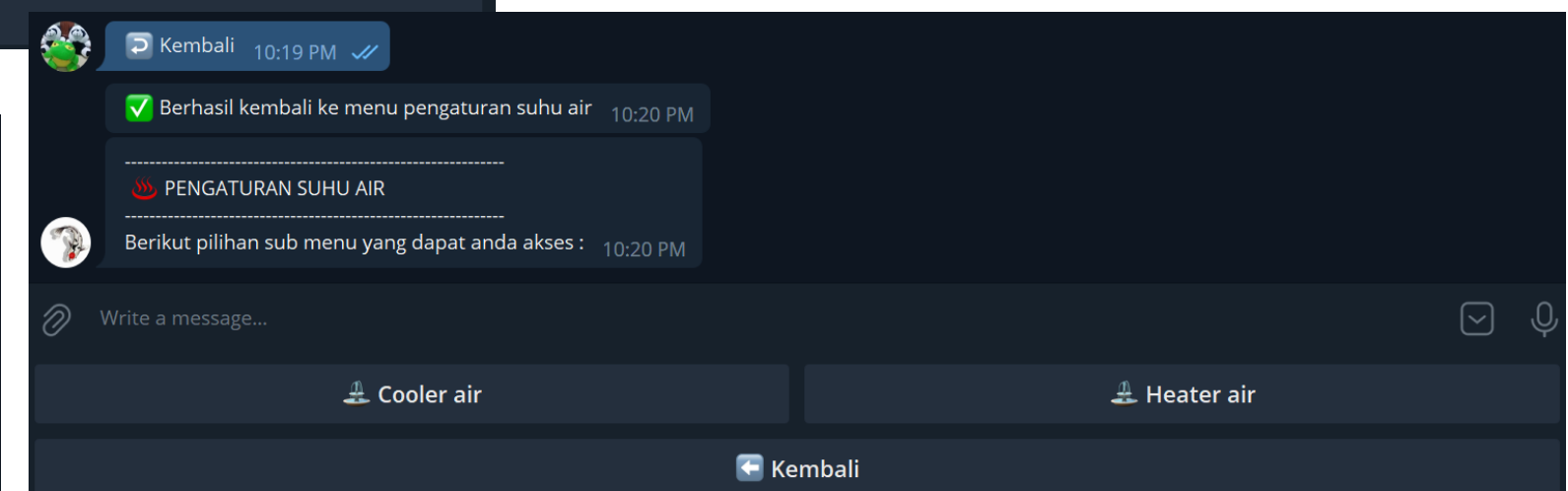
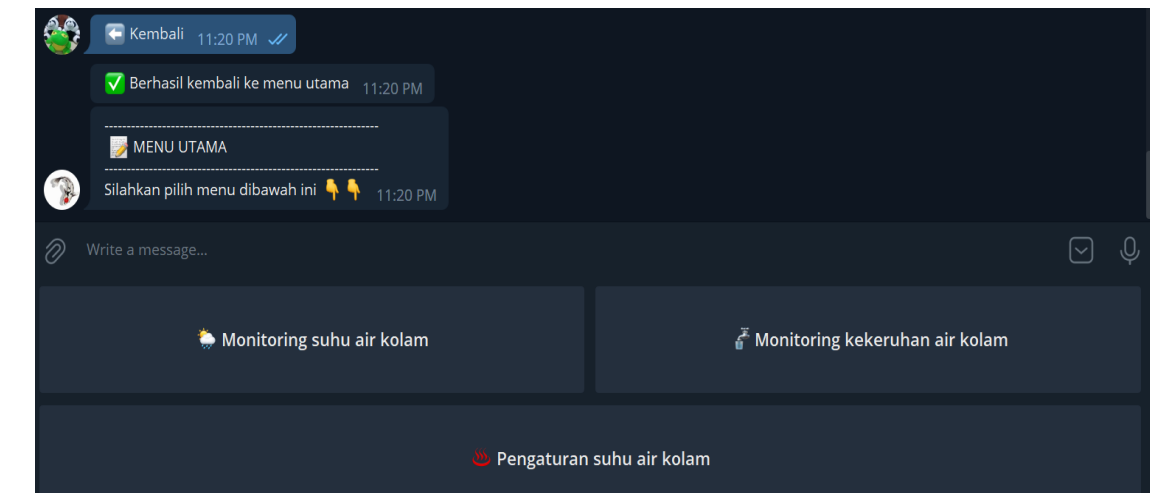
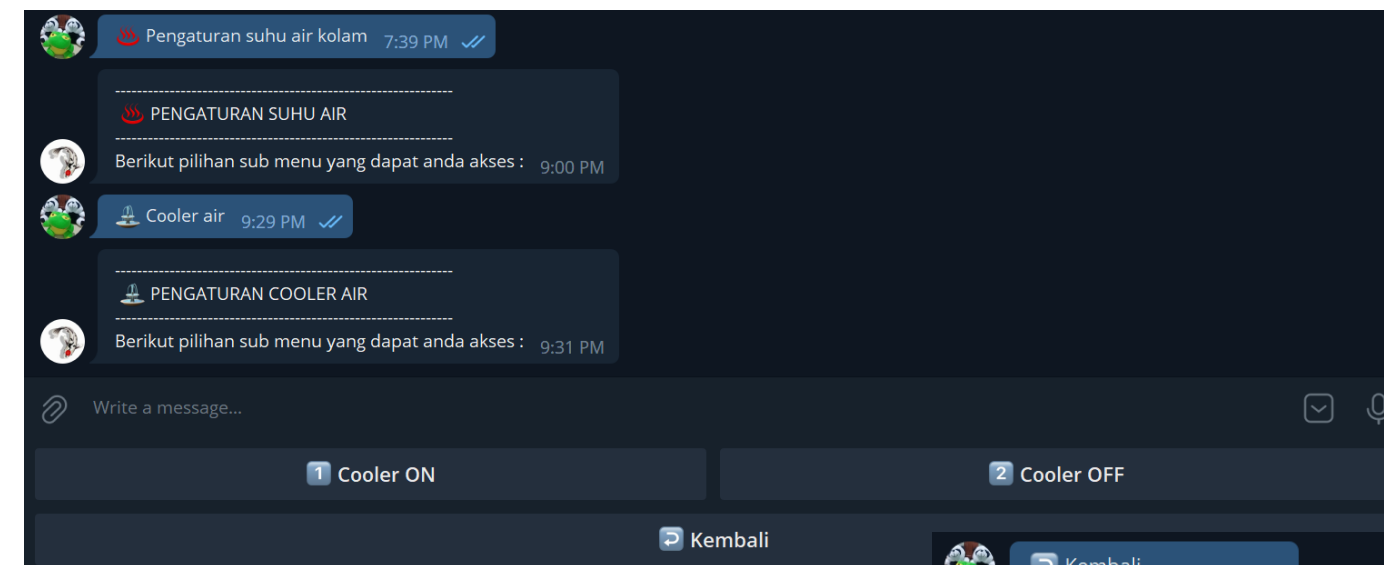
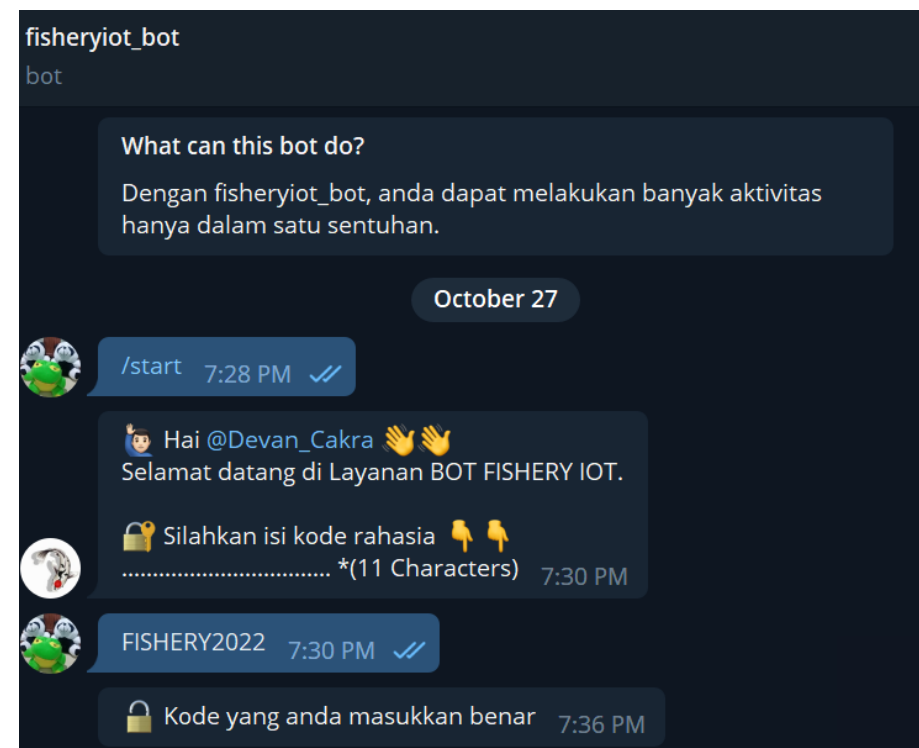
- Pada ESP Pakan, ketika penjadwalan pakan ikan sesuai dengan waktu yang kita atur, maka sistem secara otomatis aktuator servo membuka wadah makanan lalu diukur menggunakan *rotary encoder*, setelah itu akan menutup kembali.
- Pada ESP Air, ketika tombol *Heater On* pada *Bot* ditekan: Ya, maka *device heater* yang ada di Wokwi akan menyala.





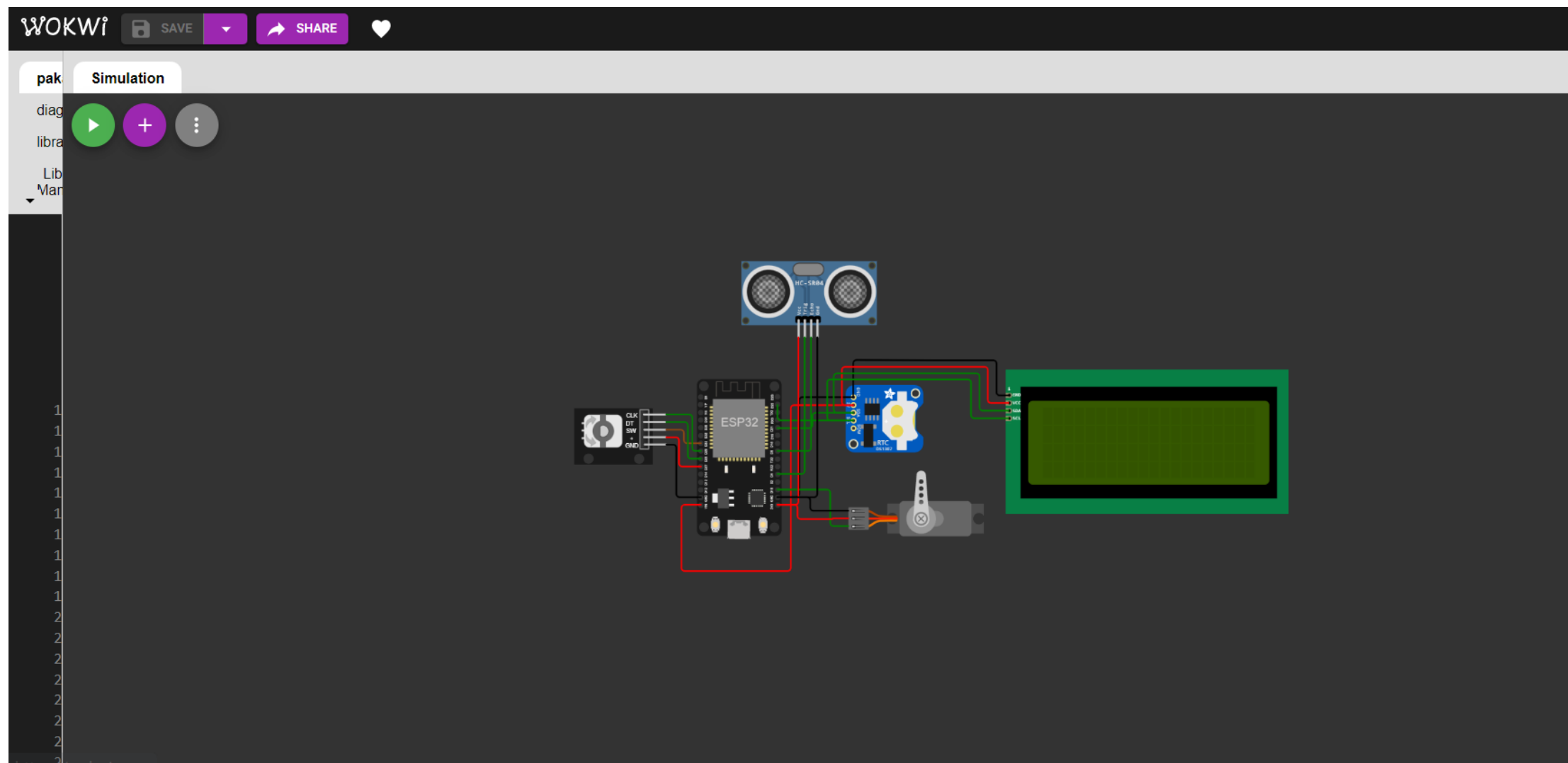
# Dokumentasi *Project #3*

Hasil dokumentasi dari pengerjaan kami sekelompok terkait *bot telegram* dapat anda lihat sebagai berikut :



# *Demo Project #1*

## *Control & Monitoring Pakan Ikan*

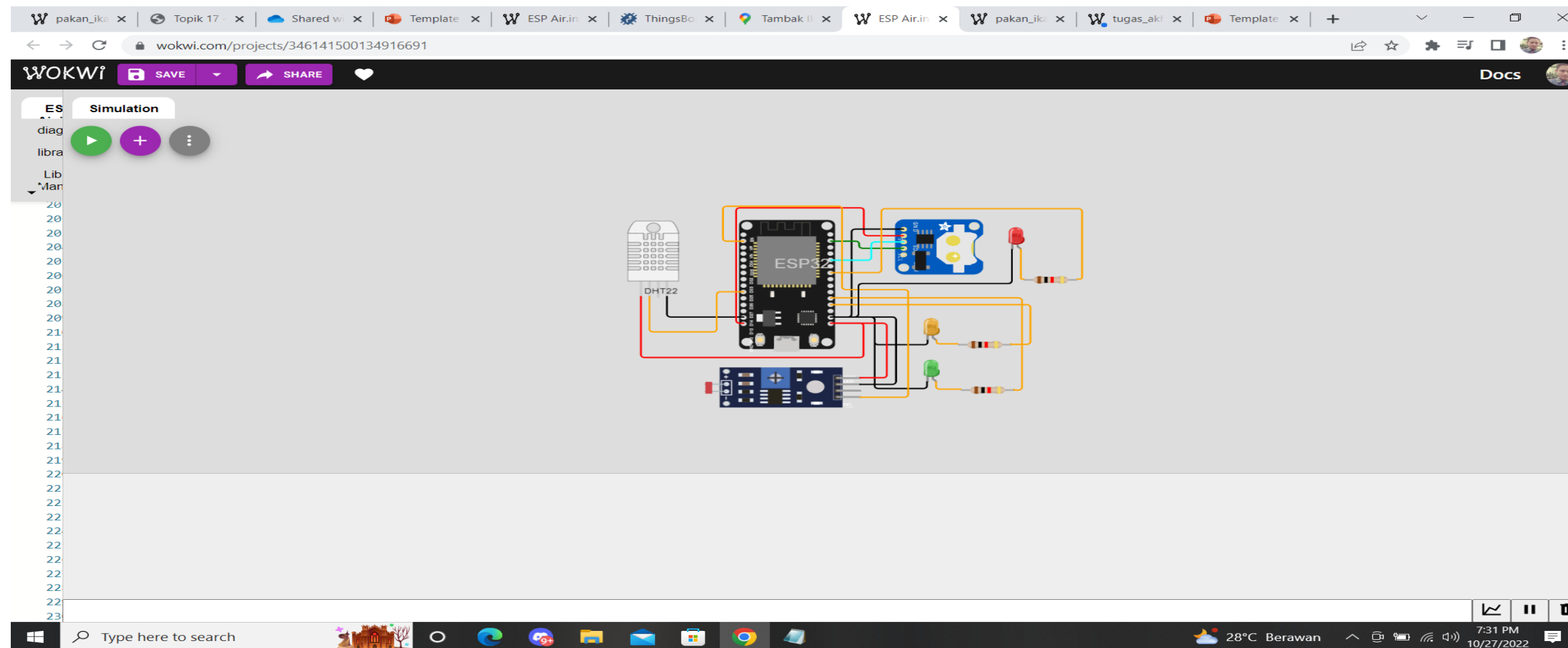


Link untuk mengakses :

<https://wokwi.com/projects/346563289358008916>

# *Demo Project #2*

## *Control & Monitoring Suhu-Kekeruhan Air*



Link untuk mengakses :

<https://wokwi.com/projects/346141500134916691>



# Kesimpulan

Berdasarkan penelitian yang telah kami lakukan didapatkan beberapa kesimpulan yang dapat kami jabarkan melalui poin-poin di bawah ini :

- Sistem aplikasi ini dapat berjalan dengan baik menggunakan protokol MQTT dan API dalam bertukar data secara *real-time*.
- Integrasi sistem yang ada di *project* kami berupa *Bot Telegram* dan *ThingsBoard*. *Bot Telegram* disini sebagai media interaksi, sedangkan *ThingsBoard* juga sama namun ada tambahan dalam bentuk visualisasi.
- Dalam *project* ini, kami melakukan kolaborasi di bagian *Device* dan *Dashboard* melalui pengaturan *Users Groups: Add User* → pilih *Tenant Administrator* → *Add User* dengan *email* yang belum terdaftar di *ThingsBoard*. Lalu atur semua bagian menjadi *public* agar dapat diakses bersama.
- Di Wokwi, jika baris kodenya banyak ketika di *compile* programnya akan ada kemungkinan *error* pada *systems* baik di Wokwi itu sendiri maupun media yang lain (*ThingsBoard* & *Bot Telegram*). Selain itu *Server Wokwi* tidak dapat bertahan lama, sehingga butuh di *refresh* secara berkala.
- Dalam proses simulasi tentu membutuhkan koneksi *internet* agar sistem yang dibuat dapat berfungsi sebagaimana mestinya. Sistem aplikasi ini dibuat harapannya untuk dapat dijadikan sebagai solusi dari permasalahan yang ada untuk peternak ikan.

# Referensi

- HobTechTV. (2021). TELEGRAM BOT INLINE KEYBOARD / BUTTON RECEIVE AND SEND DATA. Retrieved from <https://www.youtube.com/watch?v=jDvzJ8MEpv4>
- Indobot. (2022). Topic 1 - 18 For Education of IoT. In Internet of Things. DTS PROA KOMINFO.
- Stefano Ledda. (2018). CTBot. Retrieved May 2, 2022, from shurillu github website: <https://github.com/shurillu/CTBot>
- WFH20. (2020). Kontrol Lampu Menggunakan Telegram Bot, Tanpa diketik Cukup dipencet Saja | Home Automation | IoT. Retrieved from <https://www.youtube.com/watch?v=oLEW0Qlidvo&t=1092s>

**Sekian Presentasi dari Group-C**  
*Smart Fishery 2022 (Control and Monitoring)*

Kami Mengucapkan  
**TERIMA KASIH.....**

Digitalent Scholarship Professional Academy



[indobot.co.id](http://indobot.co.id)