

IoT Fish Monitoring System

Sistem monitoring ikan otomatis berbasis ESP32 untuk aquaculture modern dengan fitur monitoring suhu air, level pakan, dan feeding otomatis.

Features

- **Real-time Monitoring:** Suhu air dan level pakan dengan sensor DS18B20 dan HC-SR04
- **Automated Feeding:** Pemberian pakan terjadwal dari database MongoDB
- **Remote Control:** Kontrol manual via web interface dengan Socket.IO
- **Smart Alerts:** Notifikasi audio buzzer untuk kondisi abnormal
- **Multi-core Processing:** Task management dengan FreeRTOS pada ESP32

Hardware Requirements

Component	Model	Pin	Function
Microcontroller	ESP32 DevKit	-	Main controller
Temperature Sensor	DS18B20	GPIO 15	Water temperature
Distance Sensor	HC-SR04	GPIO 4,5	Feed level detection
Servo Motor	SG90	GPIO 18	Feed dispenser
Buzzer	Active 5V	GPIO 19	Audio alerts
RTC Module	DS3231	GPIO 21,22	Real-time clock

Installation

Arduino Libraries

- WiFi (ESP32 built-in)
 - HTTPClient (ESP32 built-in)
 - SocketIoClient by Markus Sattler
 - OneWire by Jim Studt
 - DallasTemperature by Miles Burton
 - RTCLib by Adafruit
 - ESP32Servo by Kevin Harrington

Configuration

Edit di `iot-monitoring.ino`:

cpp

```
// Network Settings
const char* ssid = "YOUR_WIFI_SSID";
const char* password = "YOUR_WIFI_PASSWORD";
const char* serverIP = "YOUR_SERVER_IP";
const int serverPort = 3000;

// Threshold Settings
const float SUHU_MIN = 20.0; // Min temperature (°C)
const float SUHU_MAX = 32.0; // Max temperature (°C)
const float PAKAN_HABIS_CM = 12.0; // Feed empty threshold (cm)
```

API Endpoints

Method	Endpoint	Description
POST	/api/sensor/kirim	Send sensor data
POST	/api/sensor/feeding-log	Log feeding activity
GET	/api/sensor/jadwal	Get feeding schedule
POST	/api/sensor/heartbeat	Device status update

Socket.IO Events

Event	Direction	Description
connect	Server → ESP32	Connection established
beri-pakan	Server → ESP32	Manual feeding trigger
jadwal-updated	Server → ESP32	Schedule update

Alert System

Condition	Response
Temperature abnormal + Feed empty	2 beeps
Manual feeding	1 beep
Auto feeding	Silent
Individual alerts	Log only

Data Format

Sensor Data

```
json
{
  "suhu": 26.5,
  "pakan_cm": 8.3,
  "waktu": "14:30"
}
```

Feeding Schedule

```
json
["07:00", "12:00", "18:00"]
```

Troubleshooting

WiFi Issues

- Check SSID and password
- Verify 2.4GHz compatibility
- Restart ESP32 and router

Sensor Issues

- **DS18B20 (-127°C):** Check wiring and 4.7kΩ pull-up resistor
- **HC-SR04 (timeout):** Verify 5V power and clear sensor path
- **RTC time wrong:** Check CR2032 battery and I2C wiring

Debug Commands

```
cpp
Serial.setDebugOutput(true);
Serial.println("Free Heap: " + String(ESP.getFreeHeap()));
WiFi.printDiag(Serial);
```

System Architecture

- **Core 0:** Auto feeding task, main loop, WiFi communication
- **Core 1:** Sensor monitoring, alert system, data collection

- **Communication:** WiFi + Socket.IO untuk real-time data
- **Database:** MongoDB untuk schedule dan data logging

Performance

- **Update Frequency:** 10s (sensor data), 5min (schedule sync)
- **Response Time:** <2 seconds (sensor to server)
- **Accuracy:** $\pm 0.5^{\circ}\text{C}$ (temperature), $\pm 3\text{mm}$ (distance)
- **Power:** 5V/2A recommended

License

MIT License - see LICENSE file for details.

Support

- Issues: [GitHub Issues](#)
- Email: support@domain.com