

BAB I PENDAHULUAN

Bab I Pendahuluan ini menjelaskan gambaran umum yang terdiri dari latar belakang topik penelitian, perumusan masalah, tujuan penelitian, batasan penelitian, dan sistematika penulisan penelitian.

I.1 Latar Belakang

Indonesia merupakan negara tropis yang kaya akan keanekaragaman hayati, termasuk di antaranya keberagaman ikan hias. Salah satu ikan hias yang sangat diminati oleh masyarakat adalah ikan koi (*Cyprinus Carpio*). Ikan koi memiliki daya tarik yang khas berkat bentuk tubuhnya yang indah, variasi warna yang menarik, pola atau motif yang unik, serta harga jual yang tinggi di pasaran. Sejak tahun 2011, ekspor ikan hias Indonesia telah menduduki peringkat kelima di dunia, dengan total ekspor mencapai US \$5,24 juta. Data ini menunjukkan kontribusi signifikan industri ikan hias Indonesia dalam pasar global (Laksono dkk., 2021).

Tabel I-1 Data Asosiasi Pcinta Koi Indonesia 2024

Nama Komunitas	Nama Komunitas	Nama Komunitas
Bogor Nishikigoi Club	Blitar Koi Club	HTP Koi Klub
Sukabumi Bersatu Koi Club	Parahyangan Koi Club	Jogja Koi Club
Makassar Koi Club	Bandung Koi Club	Jakarta Koi Club
Kediri Koi Club	Tangerang Koi Club	Bekasi Nishikigoi Club
Dewata Koi Club	Bengawan Koi Club	Keluarga Pehobi Koi Lombok (KOILO)
Malangraya Nishikigoi Club	Priangan Timur Koi Club	Tuban Koi Club
Balikpapan Koi Club	Muria Koi Club Kudus	Banua Koi Club
Galuh Nishikigoi Club	Pasuruan Untung Soeropati Koi Club	West Borneo Koi Club
Sindosui Koi Club Temanggung	Padang Koi Club	Bontang Koi Khatulistiwa

Nama Komunitas	Nama Komunitas	Nama Komunitas
Kalimantan Utara Koi Club	Gandrung Nishikigoi Club	Probolinggo Koi Club
Malang Koi Club	Sidoarjo Koi Club	Jember Koi Club
Lumajang Koi Club	Koi Lampung Club	Sultra Koi Club
Palu Nagaya Koi Club	Manado Koi Club	Mahakam Koi Club
Sandeq Koi Club Polman	Luwuk Koi Club	Cirebon Koi Community
Bondowoso Koi Club	Tabanan Koi Club Bali	Tulungagung Koi Club
Banjarmasin Koi Club	Tambun Bungai Palangkaraya Koi Club	Lubuklinggau Silampari Koi Club
Ponorogo Warok Koi Club	Wonosobo Koi Club	Selaparang Koi Club Lombok
Mangupura Koi Club	Walet Koi Kebumen	Bencoolem Koi Community

Pada Tabel I-1 ditemukan data berdasarkan dari data Asosiasi Pecinta Koi Indonesia tahun 2024, terdapat 54 komunitas atau klub anggota Asosiasi Pecinta Koi Indonesia (APKI) yang tersebar di seluruh wilayah Indonesia. Angka ini diperkirakan akan terus meningkat seiring dengan bertambahnya jumlah pecinta dan pemelihara ikan koi di Indonesia.

Kesehatan dan pertumbuhan ikan koi sangat dipengaruhi oleh kualitas air kolam. Kualitas air kolam ditentukan oleh suhu dan tingkat zat terlarut atau TDS (*Total Dissolved Solid*) dalam air. Suhu air yang tidak stabil dapat menyebabkan perubahan perilaku ikan, penurunan nafsu makan, dan memperlambat pertumbuhan ikan. Selain itu, kandungan mineral yang terlarut dalam air juga berperan dalam menentukan tingkat pertumbuhan dan warna ikan koi (Ariyanto & Kusriyanto, 2023).

Kandungan mineral terlarut dalam air merupakan faktor penting yang mempengaruhi kualitas air kolam ikan. Kadar mineral yang optimal dapat memperkuat sistem kekebalan tubuh ikan koi, meningkatkan kualitas warna dan

pola pada tubuh ikan koi. Kandungan mineral yang terlalu tinggi atau rendah dapat mengganggu pertumbuhan dan reproduksi ikan (Al Ayyubi dkk., 2023).

Perubahan suhu yang tidak stabil dapat menyebabkan kematian massal pada budidaya ikan koi. Ketika suhu air naik atau turun secara tiba-tiba, pernapasan ikan menjadi lebih aktif dan kebutuhan oksigen meningkat. Akibatnya, sistem pernapasan ikan dapat mengalami tekanan berlebihan dan mengganggu keseimbangan oksigen dalam air kolam (Damayanti dkk., 2021).

Berdasarkan dokumen SNI Nomor 7734 Tahun 2011, rentang suhu ideal untuk kolam ikan koi berada pada 20°C -28°C dan kandungan mineral yang terlarut dalam air untuk kolam ikan koi tidak lebih dari 150 ppm. Kualitas air kolam yang sesuai standar akan menciptakan lingkungan kolam yang optimal untuk pertumbuhan ikan koi.

Terdapat teknologi *Internet of Things* (IoT) yang merupakan konsep di mana berbagai perangkat fisik, seperti sensor, aktuator, dan alat-alat elektronik lainnya, yang akan terhubung ke internet dan mampu saling berkomunikasi serta berbagi data. Dalam konteks ini, IoT memungkinkan perangkat-perangkat tersebut untuk bekerja secara otomatis tanpa intervensi manusia. Teknologi ini memungkinkan pengumpulan data secara *real-time*, pemantauan jarak jauh, dan kontrol otomatis terhadap berbagai sistem dan perangkat.

Diperlukan teknologi *Internet of Things* (IoT) yang dapat membantu memantau dan mengontrol suhu air kolam ikan koi dengan menggunakan *heater* dan *chiller*. Tujuannya adalah untuk mengatur suhu air kolam ikan secara otomatis agar menjaga kondisi lingkungan yang ideal bagi pertumbuhan ikan koi. *Heater* berfungsi untuk meningkatkan suhu air kolam ikan saat suhu air turun di bawah suhu yang telah ditentukan, dan *chiller* akan menurunkan suhu air kolam apabila suhu kolam melebihi 28°C (Noer, 2021).

Penelitian dengan judul “Efektivitas Pemanfaatan Sistem Resirkulasi Akuakultur (RAS) Terhadap Kualitas Air Dalam Budidaya Ikan Koi (*Cyprinus rubrofasciatus*)” yang dilakukan oleh Lembang & Kuing (2022), berfokus pada penelitian yang dilakukan di Balai Perikanan Budidaya Air Tawar (BPBAT) dengan membuat

rangkaian Sistem Resirkulasi Akuakultur (RAS), penebaran benih ikan koi, dan pengukuran kualitas air secara deskriptif dan eksperimental. Alat yang digunakan yaitu termometer, pH meter, dan DO meter. Susunan *filter* RAS terdiri pasir, zeolite, *bioball*, dan arang. Berdasarkan hasil penelitian teknologi RAS dapat menstabilkan suhu menjadi 29,7°C, pH menjadi 7,1, dan DO menjadi 6,0 mg/L sesuai standar kualitas air yang ditentukan untuk budidaya ikan koi.

Penelitian yang dilakukan oleh Bella (2023) dengan judul “Implementasi *Smart* Akuarium Berbasis *Internet of Things* (IoT) pada Salma Akuarium Ikan Hias” berfokus pada penerapan sistem sirkulasi air. Penelitian ini menerapkan sistem *monitoring* air dengan menggabungkan sistem pengolahan air secara geologi dan biologi untuk mengukur dan memantau suhu dan kekeruhan air menggunakan aplikasi *blynk*. Komponen alat yang digunakan yaitu NodeMCU, ESP8266, sensor *turbidity*, DHT11, motor servo, dan *buzzer*. Hasil pengujian menunjukkan *dashboard blynk* berjalan secara *real-time* dan rata-rata waktu selisih yang dibutuhkan untuk melihat respon alat dan aplikasi yaitu 1,9 detik.

Penelitian yang dilakukan oleh Nurdina dkk., (2023) dengan judul “Penerapan *Internet of Things* (IoT) *Monitoring* dan *Controlling* Perawatan Anakan Ikan Koi berbasis *Website*” berfokus pada pengembangan sistem *monitoring* dan *controlling* pemeliharaan ikan koi berbasis *website* dan IoT menggunakan sensor DHT11 untuk *monitoring* suhu dan kelembapan udara dengan tingkat akurasi sensor suhu sebesar 5%, dan tingkat akurasi pembacaan suhu sebesar 2% untuk pengambilan data minimal 1Hz. Hasil pengujian yang didapatkan adalah sensor dapat melakukan *monitoring* berdasarkan data yang sudah ditentukan. Jika nilai pH berada di luar rentang 6,5-8,0 pompa air akan menyala untuk menguras dan mengganti air yang baru. Jika suhu kurang dari 20°C maka *heater* akan menyala. Jika suhu lebih dari 28°C maka kipas akan menyala. Grafik sensor, *data history* sensor pH, suhu air, dan suhu udara dapat di-*monitoring* melalui *website*.

Dalam penelitian ini penulis mengembangkan sistem pemantauan kualitas air akuarium ikan koi berbasis *Internet of Things* (IoT) dengan menggunakan mikrokontroler ESP32 dan sensor suhu. Data suhu yang diukur oleh sensor akan diolah menggunakan kontrol *Proportional Integral Derivative* (PID) untuk

mengatur suhu air akuarium ikan koi. Data suhu yang telah diolah akan digunakan untuk mengendalikan *heater* dan *chiller*, serta dikirim ke Firebase untuk disimpan dan dikelola. Data suhu kemudian ditampilkan dalam bentuk *smart dashboard*, sehingga pemantauan kualitas air akuarium ikan koi dapat dilakukan secara *real-time* dan otomatis dengan menciptakan sistem yang terintegrasi dan efektif .

I.2 Perumusan Masalah

Rumusan masalah yang mendasari penelitian ini adalah sebagai berikut.

- a. Bagaimana merancang dan mengimplementasikan *smart dashboard* berbasis *Internet of Things* (IoT) untuk memantau dan mengontrol suhu air akuarium ikan koi secara *real-time*?
- b. Bagaimana mengintegrasikan sensor suhu, *heater*, *chiller*, dan mikrokontroler agar dapat mengirimkan data melalui internet ke dalam IoT *platform* (Firebase) dan menyalakan atau mematikan *heater* atau *chiller* melalui *relay*?

I.3 Tujuan Penelitian

Tujuan penelitian ini adalah sebagai berikut.

- a. Membuat dan menerapkan sistem berbasis *Internet of Things* (IoT) yang efektif dan efisien dalam memantau suhu akuarium budidaya ikan koi. Sistem ini akan memungkinkan pemantauan suhu akuarium secara *real-time* dan memberikan informasi yang akurat tentang suhu air akuarium.
- b. Mengintegrasikan sensor suhu, *heater*, *chiller*, dan mikrokontroler ke dalam platform IoT (Firebase) sehingga memungkinkan pengiriman data suhu air secara *real-time* melalui internet dan pengendalian otomatis *heater* atau *chiller* menggunakan relay.

I.4 Batasan Penelitian

Batasan dalam penelitian tugas akhir ini adalah sebagai berikut.

- a. Penelitian ini berfokus pada pengembangan sistem *dashboard monitoring* dan pengendalian suhu akuarium ikan koi berbasis IoT, dengan menggunakan sensor suhu dan perangkat IoT lainnya.
- b. Penelitian ini dilakukan dengan memanfaatkan teknologi IoT dan *platform* Firebase untuk menghubungkan sensor suhu dengan *smart dashboard*.

- c. Sistem *monitoring* dan kontrol yang dirancang dengan menggunakan mikrokontroler ESP32 sebagai pusat sistem dan *smart dashboard* sebagai antarmuka pengguna.

I.5 Manfaat Penelitian

Setelah didapatkan berbagai unsur yang telah dipaparkan dapat dijadikan suatu manfaat di antaranya sebagai berikut.

- a. Bagi penulis, penelitian ini memberi pengalaman dalam merancang, membangun, dan mengelola sistem pemantauan suhu berbasis *Internet of Things* (IoT), yang merupakan keterampilan penting di bidang teknologi informasi.
- b. Bagi pemilik budidaya ikan koi, penelitian ini memberikan wawasan tentang pentingnya penggunaan teknologi untuk mengatur suhu air akuarium secara otomatis, sehingga dapat meningkatkan efisiensi dalam budidaya ikan koi dan menjaga kesehatan ikan dengan lebih baik.
- c. Bagi Telkom University, penelitian ini meningkatkan reputasi universitas dengan mengembangkan teknologi yang relevan dengan industri pertanian dan perikanan, khususnya dalam budidaya ikan Koi; memberikan kesempatan untuk bekerja sama dengan industri perikanan dan teknologi *Internet of Things* untuk menghasilkan solusi yang lebih baik dan inovatif.
- d. Bagi peneliti lain, penelitian ini dapat menjadi referensi untuk penelitian selanjutnya dalam bidang yang sama atau terkait. Metodologi yang dikembangkan dalam penelitian ini dapat memberikan kontribusi dalam pengembangan teknologi *Internet of Things* dalam bidang perikanan.

Adapun sistematika penulisan dari setiap bab yang akan dibahas dalam tugas akhir ini terdiri atas:

- **BAB I PENDAHULUAN**

Bab I Pendahuluan digunakan untuk menjelaskan mengenai apa yang melatar belakangi penelitian ini dilakukan, beserta maksud dan tujuannya. Bab ini digunakan untuk memberikan penjelasan secara umum kepada pembaca.

- **BAB II TINJAUAN PUSTAKA**

Bab ini digunakan untuk memberikan referensi atau landasan teori dari setiap hal yang digunakan pada penulisan tugas akhir, yang memerlukan penjelasan lebih dalam dan tidak dapat dipahami secara umum.

- **BAB III METODOLOGI PENELITIAN**

Bab ini digunakan untuk menjelaskan bagaimana penelitian ini dilakukan, diawali dengan cara menyelesaikan masalah hingga tahapan yang digunakan untuk menyelesaikan masalah yang ada.

- **BAB IV ANALISIS DAN PERANCANGAN SISTEM**

Bab ini digunakan untuk menganalisis data yang telah dikumpulkan serta merancang sistem yang akan dibangun berdasarkan hasil analisis tersebut.

- **BAB V IMPLEMENTASI DAN PENGUJIAN**

Bab ini digunakan untuk menjelaskan proses implementasi dari rancangan yang telah dibuat dan pengujian sistem untuk memastikan bahwa sistem berfungsi sesuai dengan yang diharapkan.

- **BAB VI KESIMPULAN DAN SARAN**

Bab ini menyajikan ringkasan hasil penelitian, menyoroti pencapaian tujuan penelitian dan menjawab rumusan masalah. Selain itu, bab ini memberikan rekomendasi untuk penelitian lebih lanjut, perbaikan sistem, dan aplikasi praktis dari hasil penelitian.

BAB II TINJAUAN PUSTAKA

Bab ini membahas hasil literatur dari studi pustaka yang berkaitan dengan permasalahan yang diangkat pada penelitian ini. Selain itu, bab ini juga membahas pemilihan teori beserta alasannya, kerangka kerja, dan mekanisme yang digunakan.

II.1 Ikan Koi

Ikan koi (*Cyprinus carpio*) adalah spesies ikan air tawar yang populer di seluruh dunia, terutama di Jepang, karena pola warna yang menarik dan variasi bentuk tubuh yang unik seperti pada Gambar II.1. Ikan koi memiliki sifat-sifat seperti kecerdasan, daya tahan yang kuat, dan panjang umur, yang semuanya menambah daya tariknya sebagai ikan hias. Selain itu, ikan koi juga memiliki peran ekologis yang signifikan dalam menjaga keseimbangan ekosistem perairan, karena mereka membantu mengontrol populasi serangga dan tumbuhan air.



Gambar II.1 Ikan Koi

Ikan koi memiliki sejarah panjang dalam budaya Jepang dan telah menjadi simbol keberuntungan, kesuksesan, dan keberanian. Mereka sering dikaitkan dengan nilai-nilai positif dan sering digambarkan dalam seni Jepang. Pemeliharaan ikan koi juga telah menjadi industri yang signifikan, dengan munculnya berbagai varietas baru melalui pemuliaan selektif. Selain itu, ikan koi juga memiliki peran dalam menjaga keseimbangan ekosistem perairan, karena mereka membantu mengontrol populasi serangga dan tumbuhan air (Hestukoro dkk., 2022).

II.2 Suhu Ideal untuk Budidaya Ikan Koi

Suhu air yang terlalu tinggi atau rendah dapat mempengaruhi kualitas air di kolam pembenihan ikan koi. Pada suhu air yang terlalu tinggi atau rendah, kualitas air dapat cepat menurun karena pencemaran aktivitas lingkungan dan hasil metabolisme ikan koi. Selain itu, suhu air yang tidak sesuai juga dapat mempengaruhi tingkat kematian ikan koi. Oleh karena itu, penting untuk mengontrol suhu air pada rentang yang ideal, yaitu antara 25°C dan 27°C (Lembang & Kuing, 2022).

Selain suhu, faktor lain yang berpengaruh terhadap kualitas air di kolam pembenihan ikan koi adalah pH dan kadar oksigen terlarut (DO) dari air kolam. Kualitas air yang optimal akan menyebabkan pertumbuhan ikan koi yang baik. Untuk menjaga kualitas air di kolam pembenihan ikan koi, dapat dilakukan teknik pembenihan yang meliputi persiapan kolam, pemeliharaan induk, pemijahan, penetasan telur, pemeliharaan larva, pengelolaan kualitas air, dan pengendalian hama dan penyakit. Selain itu, dapat juga dilakukan pemasangan filter air untuk meningkatkan hasil pembenihan ikan koi dengan meningkatkan kualitas air di kolam pemijahan (Susanti dkk., 2022).

II.3 Sistem *Smart Dashboard Monitoring*

Smart dashboard monitoring adalah metode penyajian informasi yang menggunakan kombinasi gambar dan tulisan untuk menampilkan data dan informasi secara lebih efisien. Dengan teknologi *Internet of Things* (IoT), *smart dashboard* memungkinkan pemantauan dan pengendalian aktivitas atau proses secara *real-time* melalui tampilan visual yang mudah dipahami. Metode ini memanfaatkan berbagai sensor dan perangkat yang terhubung ke internet untuk mengumpulkan dan mengirimkan data secara kontinu, sehingga memberikan informasi yang akurat dan terkini kepada pengguna.

Smart dashboard monitoring dilengkapi dengan indikator kinerja utama, atau *Key Performance Indicators* (KPI), yang disesuaikan dengan kebutuhan manajemen terkait informasi yang ditampilkan dalam *dashboard*. KPI ini membantu manajemen dalam mengevaluasi dan memonitor performa sistem atau proses tertentu, serta memudahkan pengambilan keputusan berdasarkan data yang

tersedia. Dengan demikian, *smart dashboard monitoring* tidak hanya meningkatkan efisiensi dalam pengolahan data tetapi juga mendukung manajemen dalam mencapai tujuan strategis organisasi (Apriyani dkk., 2022).

II.4 User Interface dan User Experience

User Interface (UI) dan *User Experience* (UX) adalah dua aspek penting dalam pengembangan aplikasi dan sistem informasi yang mempengaruhi kenyamanan pengguna dan efektivitas kerja. UI merujuk pada segala yang berkaitan dengan tampilan dan antarmuka aplikasi, sedangkan UX mencakup cara pengguna berinteraksi dengan aplikasi (Fachridwan & Sondari, 2023).

UI dan UX sangat berperan dalam meningkatkan efisiensi, produktivitas, dan kepuasan pengguna. Hal ini dapat dilihat dari berbagai penelitian yang menggunakan metode *design thinking* untuk mengembangkan UI dan UX aplikasi. Dari beberapa penelitian menunjukkan bahwa penerapan metode *design thinking* dalam perancangan UI dan UX dapat menghasilkan solusi yang sesuai dengan kebutuhan pengguna dan memberikan kenyamanan yang baik (Kurniawan dkk., 2023).

II.5 PHP: Hypertext Preprocessor

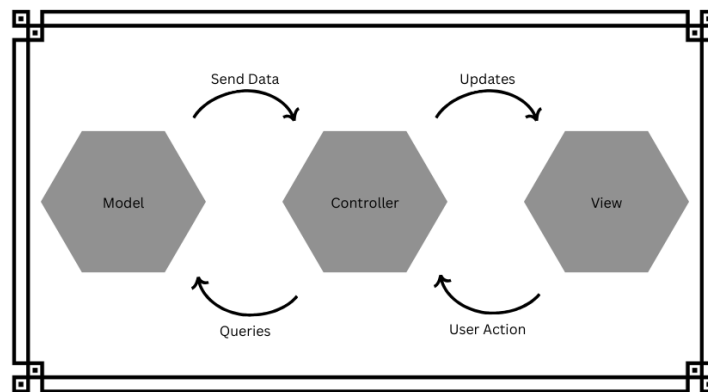
PHP adalah bahasa pemrograman yang digunakan untuk membangun aplikasi berbasis web. PHP dapat digunakan untuk membuat halaman web dinamis, mengakses *database*, dan melakukan pemrosesan data pada sisi server. Dengan menulis kode PHP pada *file* dengan ekstensi *.php*, kemudian *file* tersebut di unggah ke server web, kode PHP akan diinterpretasikan oleh server web dan menghasilkan *output* yang dapat ditampilkan pada halaman web. Selain itu, PHP juga dapat digunakan bersama dengan *database* seperti MySQL untuk mengakses dan memanipulasi data (Jiwandono dkk., 2022).

PHP memiliki keunggulan dalam kemudahan integrasi dengan berbagai layanan web dan alat-alat pengembangan. Misalnya, PHP dapat berinteraksi dengan berbagai protokol seperti HTTP, POP3, SNMP, LDAP, dan lainnya, yang memungkinkan pengembang untuk membuat aplikasi web yang kompleks dan terintegrasi. Selain itu, PHP mendukung banyak *framework* dan pustaka yang

mempermudah pengembangan aplikasi, seperti Laravel, Symfony, dan CodeIgniter. Dengan keunggulan ini, PHP tidak hanya berguna untuk pengembangan situs web sederhana tetapi juga untuk aplikasi web yang lebih besar dan lebih kompleks, menjadikannya pilihan populer di kalangan pengembang web di seluruh dunia (Syafitri dkk., 2022).

II.6 Laravel

Laravel adalah kerangka kerja aplikasi web berbasis PHP yang *open source* dan digunakan untuk pengembangan aplikasi web. Dengan sintaks yang ekspresif dan elegan, Laravel menyediakan berbagai fitur yang memudahkan dalam membangun aplikasi, seperti sistem *routing* yang kuat, manajemen sesi, dan sistem yang ekstensif. Selain itu, Laravel juga mendukung berbagai macam basis data dan memiliki mekanisme migrasi basis data yang kuat untuk memudahkan pengembangan dan pemeliharaan aplikasi. Dengan komunitas yang besar dan aktif, serta dokumentasi yang lengkap, Laravel menjadi pilihan yang populer bagi para pengembang web di seluruh dunia (Lisnawati dkk., 2023).



Gambar II.2 Konsep MVC Laravel

Pada Gambar II.2 dijelaskan bahwa Laravel juga dikenal dengan konsep *Model-View-Controller* (MVC) yang memisahkan logika aplikasi, tampilan, dan interaksi pengguna. Hal ini memudahkan dalam pengelolaan kode dan pengembangan fitur baru. Selain itu, Laravel memiliki *blade templating engine* yang memungkinkan pengembang untuk membuat tampilan aplikasi dengan cara yang lebih ekspresif dan efisien. Dengan adanya fitur-fitur tersebut, Laravel

menjadi salah satu kerangka kerja yang sangat *powerful* dan dapat mempercepat proses pengembangan aplikasi web (Gianistika, 2022).

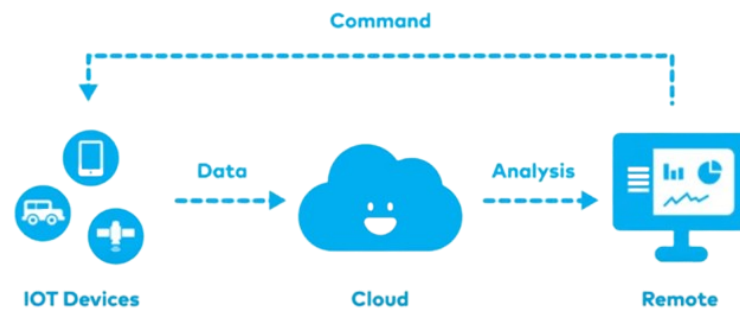
II.7 MySQL

MySQL adalah sebuah sistem manajemen basis data relasional yang bersifat *open source*. MySQL digunakan untuk mengelola dan menyimpan data dalam sebuah *database*. Sistem ini dapat digunakan pada berbagai sistem operasi seperti Windows, Linux, dan Mac OS, serta mendukung berbagai bahasa pemrograman seperti PHP, Java, dan Python. Dengan fitur-fitur seperti keamanan yang baik, dukungan transaksi, dan kemampuan untuk mengelola database yang besar, MySQL menawarkan solusi yang andal dan efisien untuk pengelolaan data.

MySQL juga memiliki komunitas pengguna yang besar dan aktif, sehingga pengguna dapat dengan mudah menemukan solusi untuk masalah yang dihadapi. Dengan dukungan komunitas ini, pengguna dapat mengakses berbagai sumber daya, seperti dokumentasi, forum diskusi, dan tutorial. MySQL digunakan oleh banyak perusahaan dan organisasi besar di seluruh dunia untuk mengelola data mereka, memungkinkan pembuatan, pengelolaan, dan akses *database* dengan mudah dan efisien (Susilowati, 2021).

II.8 Internet of Things (IoT)

Internet of Things (IoT) adalah sebuah konsep yang memungkinkan perangkat elektronik untuk saling terhubung dan berkomunikasi melalui jaringan internet seperti pada Gambar II.3. Hal ini memungkinkan akses dan kontrol terhadap perangkat secara terpusat, baik secara lokal maupun dari jarak jauh. IoT telah diterapkan dalam berbagai bidang, seperti *smart aquarium* yang memanfaatkan sensor suhu, sensor kekeruhan air, dan motor servo untuk memberi makan ikan secara otomatis. Selain itu, IoT juga digunakan dalam *smart security* berbasis *module cam* untuk menanggulangi angka kejahatan dengan fitur pencegahan aktif dan koneksi ke aplikasi *smartphone*. Di sektor pertanian, IoT digunakan dalam konsep *smart farming* untuk memantau dan mengontrol kondisi pertumbuhan tanaman secara otomatis (Bella, 2023).

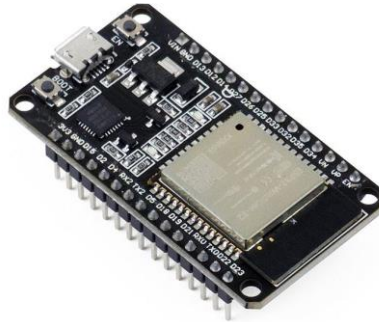


Gambar II.3 Konsep *Internet of Things*

Sejarah IoT dimulai pada tahun 2009, ketika seorang pria bernama Kevin Ashton mengemukakan sebuah ide menghubungkan beberapa perangkat dan bertukar data dengan perangkat lainnya dalam satu jaringan. Konsep IoT telah berkembang sejak saat itu dan telah diterapkan dalam berbagai bidang, seperti medis, *cloud computing*, dan beberapa bidang keilmuan lain. Dengan adanya IoT, diharapkan dapat tercipta ekosistem yang lebih efisien dan terkoneksi, serta memberikan kemudahan dalam berbagai aspek kehidupan (Wilianto & Kurniawan, 2018).

II.9 Node MCU ESP32

Pada Gambar II.4 terdapat Node MCU ESP32 yaitu sebuah mikrokontroler yang dikembangkan oleh Espressif Systems. Mikrokontroler ini memiliki berbagai fitur yang membuatnya ideal untuk digunakan dalam proyek *Internet of Things* (IoT) dan proyek lainnya. Beberapa fitur yang dimiliki Node MCU ESP32 antara lain modul WiFi yang terpasang *onboard*, modul kamera untuk menangkap gambar dan video, serta kemampuan untuk bekerja dengan berbagai jenis sensor seperti sensor PIR Motion, sensor ultrasonik, dan sensor GPS. Selain itu, Node MCU ESP32 juga menggunakan perangkat lunak seperti FreeRTOS dan *framework* Arduino, serta memiliki beragam modul perpustakaan seperti UART, I2C, SPI, ADC, DAC, dan PWM (Bagye dkk., 2023).



Gambar II.4 Mikrokontroler ESP32 Modul WiFi

II.10 Sensor Suhu DS18B20

Sensor suhu DS18B20 adalah sensor digital yang umum digunakan untuk mengukur suhu dalam berbagai aplikasi. Sensor ini dapat mengukur suhu mulai dari -55°C hingga $+125^{\circ}\text{C}$ dengan akurasi $\pm 0,5^{\circ}\text{C}$. Sensor ini menggunakan antarmuka *1-wire* untuk berkomunikasi dengan mikrokontroler dan dapat ditenagai oleh jalur yang sama yang digunakan untuk komunikasi. Sensor ini sering digunakan dalam sistem pemantauan suhu untuk akuarium, ruang dingin, pemanas air, dan aplikasi industri serta rumah tangga lainnya. Sensor ini telah diimplementasikan dalam berbagai proyek penelitian dan pengembangan, seperti sistem pakan ikan otomatis berdasarkan suhu air, perangkat deteksi tingkat stres, dan sistem kontrol suhu air. Sensor ini dikenal karena akurasi dan keandalannya dalam pengukuran suhu (Fitri & Rasyid, 2023).

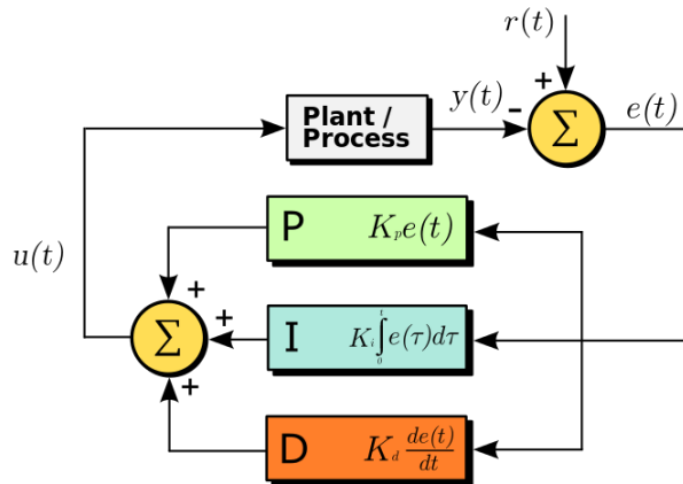


Gambar II.5 Sensor Suhu DS18B20

Pada Gambar II.5 terdapat Sensor Suhu DS18B20 yang mempunyai kelebihan dari sensor suhu DS18B20 meliputi akurasi tinggi dengan $\pm 0,5^{\circ}\text{C}$, antarmuka 1-wire untuk komunikasi yang efisien, serta kemampuan untuk digunakan dalam berbagai aplikasi seperti pemantauan suhu akuarium, ruang dingin, dan pemanas air. Selain itu, sensor ini telah terbukti cocok untuk digunakan dalam jaringan pengukuran suhu udara yang akurat. Namun, sensor ini memiliki beberapa keterbatasan, termasuk jarak maksimum komunikasi yang relatif kecil, yaitu 10 meter, serta batas suhu maksimal yang juga relatif rendah, yaitu $+125^{\circ}\text{C}$. Meskipun demikian, sensor ini tetap menjadi pilihan yang populer dalam berbagai proyek, termasuk dalam pemantauan suhu tubuh pasien dan pengembangan sistem pemantauan kesehatan (Elyounsi & Kalashnikov, 2021).

II.11 Proportional Integral Derivative (PID)

Proportional Integral Derivative (PID) adalah sebuah metode pengendalian yang digunakan dalam berbagai aplikasi untuk menjaga variabel proses pada nilai yang diinginkan. Kontrol PID bekerja dengan menghitung kesalahan antara nilai yang diinginkan (*setpoint*) dan nilai aktual dari variabel proses. Kontrol Proportional Integral Derivative (PID) digunakan untuk mengatur suhu dalam berbagai aplikasi, termasuk dalam pengeringan biji kopi, mesin pengepres kopi, mesin molding mikroplastik, dan mesin pengering pelet. Metode ini bekerja dengan menghitung *error* antara *set point* (suhu yang diinginkan) dan suhu aktual, lalu menyesuaikan *output* kontrol (misalnya pemanas) berdasarkan proporsional, integral, dan komponen turunan dari *error* tersebut. Kelebihan metode PID termasuk kemampuannya untuk menghasilkan respons yang cepat dan stabil terhadap perubahan suhu, tetapi pengaturan parameter PID yang tepat diperlukan untuk mencegah *overshoot* atau osilasi. Selain itu, metode PID juga memungkinkan untuk *monitoring* dan kontrol jarak jauh melalui internet (*IoT monitoring*) dalam beberapa implementasi. Dalam beberapa kasus, pengaturan parameter PID yang tepat dapat menjadi tantangan, dan penggunaan metode ini memerlukan pemahaman yang baik tentang sistem yang dikendalikan serta karakteristiknya (Pramudyo dkk., 2021). Adapun konsep metode PID pada Gambar II.6 dan persamaan (II-1) merepresentasikan model matematis PID.



Gambar II.6 Konsep Metode PID

$$mv(t) = K_p e(t) + K_i \int_0^t e(\tau) dt + K_d \frac{de(t)}{dt} \quad (\text{II-1})$$

Komponen kontrol PID ini terdiri dari tiga jenis yaitu proporsional, integratif dan derivatif. Ketiganya dapat dipakai bersamaan maupun sendiri-sendiri tergantung dari respon yang kita inginkan terhadap suatu plant.

1. Kontrol Proporsional

Kontrol Proporsional (P) adalah metode kontrol yang menggunakan sinyal kesalahan (selisih antara besaran yang diinginkan dengan besaran aktual) untuk menghasilkan keluaran yang proporsional dengan besarnya sinyal kesalahan. Metode ini memiliki berbagai keterbatasan karena sifat kontrol yang tidak dinamis, namun cukup mampu untuk memperbaiki respon transien khususnya *rise time* dan *settling time* dalam aplikasi-aplikasi dasar yang sederhana (Purwanto dkk., 2022).

2. Kontrol Integratif

Pengontrol Integral (I) berfungsi untuk menghasilkan respons sistem yang memiliki kesalahan keadaan mantap nol. Dalam pengontrol integral, keluaran pengontrol merupakan hasil penjumlahan yang terus menerus dari perubahan *input*, dan jika sinyal kesalahan tidak mengalami perubahan, maka *output* akan menjaga keadaan seperti sebelum terjadinya perubahan *input*. Pengontrol integral dapat memperbaiki sekaligus menghilangkan respon *steady state*, namun pemilihan parameter K_i (konstanta integral) yang tidak tepat dapat menyebabkan respon transien yang tinggi sehingga dapat menyebabkan

ketidakstabilan sistem. Pemilihan K_i yang sangat tinggi juga dapat menyebabkan *output* berosilasi karena menambah orde sistem (Agustin dkk., 2020).

3. Kontrol Derivatif

Pengontrol Derivatif (D) memiliki sifat yang mirip dengan operasi derivatif. Perubahan mendadak pada masukan pengontrol akan mengakibatkan perubahan yang sangat besar dan cepat. Ketika masukan tidak berubah, keluaran pengontrol juga tidak berubah. Namun jika masukan berubah mendadak, keluaran akan menghasilkan sinyal berbentuk impuls. Jika masukan berubah naik secara perlahan, keluaran akan menjadi besar dan dipengaruhi oleh kecepatan naik dari masukan serta konstanta K_d . Pengontrol derivatif hanya bereaksi saat ada perubahan *error*, sehingga tidak dapat digunakan sendirian (Qurani, 2021).

Pengontrol PID (Proporsional-Integral-Derivatif) memiliki kontribusi besar dari ketiga parameter P, I, dan D. Penyetelan konstanta K_p , K_i , dan K_d akan memengaruhi sifat dari masing-masing elemen, dan konstanta yang menonjol akan memberikan kontribusi pengaruh pada respon sistem secara keseluruhan.


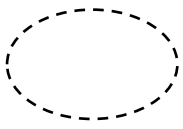
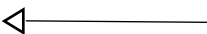




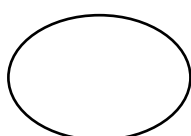
II.12 Unified Modeling Language (UML)

UML atau *Unified Modeling Language* adalah bahasa pemodelan standar yang digunakan dalam rekayasa perangkat lunak untuk memvisualisasikan, merancang, dan mendokumentasikan sistem perangkat lunak. UML menyediakan seperangkat notasi grafis dan kerangka pemodelan yang memungkinkan pengembang membuat diagram untuk berbagai aspek sistem, seperti diagram *use case diagram*, *use case scenario*, *activity diagram*, dan *deployment diagram*. Fungsinya dalam perancangan perangkat lunak adalah membantu pengembang untuk berkomunikasi dan memahami struktur, perilaku, dan fungsionalitas sistem secara lebih terstruktur dan terukur (Ramadhan dkk., 2023). UML juga mendukung banyak alat pengembangan perangkat lunak, digunakan dalam pengembangan aplikasi mobile berbasis Android, *website*, dan lain-lain.

II.12.1 Use Case Diagram

Use case diagram adalah salah satu jenis diagram dalam *Unified Modeling Language* (UML) yang digunakan untuk menggambarkan interaksi antara pengguna dan sebuah sistem. *Use case diagram* berguna untuk memvisualisasikan kebutuhan fungsional sistem dan interaksi antara pengguna dan sistem (Adhitama, 2018). Diagram ini terdiri dari aktor, *use case*, dan hubungan antara keduanya. Pada Tabel II-1 dijelaskan secara singkat mengenai simbol-simbol yang umum digunakan dalam *use case diagram*:

Tabel II-1 Simbol-Simbol *Use Case Diagram*

Simbol	Nama	Keterangan
	<i>Actor</i>	Mewakili pengguna atau sistem lain yang berinteraksi dengan sistem.
	<i>Collaboration</i>	Interaksi antara objek satu dengan lainnya untuk saling bekerja sama
	<i>Extend</i>	Menunjukkan bahwa satu <i>use case</i> dapat memperluas <i>use case</i> lain.
	<i>Include</i>	Menunjukkan bahwa satu <i>use case</i> mencakup fungsionalitas dari <i>use case</i> lain.
	<i>Generalization</i>	Hubungan objek keturunan memiliki perilaku yang berasal dari objek induk
	<i>Dependency</i>	Hubungan antara suatu perubahan objek mandiri akan berpengaruh dengan objek yang bergantung pada objek lainnya.
	<i>Association</i>	Garis yang menghubungkan aktor dengan <i>use case</i> , mewakili komunikasi atau interaksi antara keduanya.
	<i>Use Case</i>	Mewakili fungsionalitas spesifik atau unit kerja diskrit yang dilakukan oleh sistem.

Cara membuat *use case diagram* meliputi beberapa langkah:

- a. Identifikasi *actor*, *use case*, dan interaksi antara *actor* dan *use case* yang akan dijelaskan dalam diagram.
- b. Jelaskan setiap *use case* dengan menggunakan bahasa yang sederhana dan jelas.
- c. Jelaskan interaksi antara *actor* dan *use case* dengan menggunakan *arrows* (panah) yang menunjukkan aliran kontrol dan hubungan antara elemen dalam diagram.

II.12.2 Use Case Scenario

Use case scenario adalah deskripsi interaksi antara sistem dan aktor yang menggambarkan bagaimana sistem akan digunakan dalam situasi tertentu. Sementara itu, simbol-simbol dalam konteks ini dapat merujuk pada simbol-simbol yang digunakan dalam pemodelan *use case*, seperti *actor*, *use case*, dan hubungan di antara mereka. Dalam konteks pemodelan *use case*, simbol-simbol yang umum digunakan antara lain:



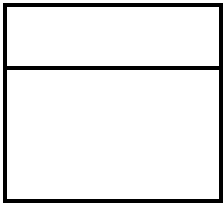
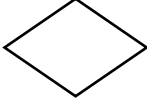


- a. *Actor*:
Digambarkan dengan simbol manusia atau sesuatu di luar sistem yang berinteraksi dengan sistem.
- b. *Use case*:
Digambarkan dengan *elips*, mewakili fungsionalitas sistem yang dapat digunakan oleh aktor.
- c. Hubungan antara *actor* dan *use case*:
Digambarkan dengan garis, menunjukkan hubungan antara *actor* dan fungsionalitas sistem yang terkait.

Dengan demikian, *use case scenario* memberikan pemahaman yang jelas tentang bagaimana sistem akan digunakan dalam situasi nyata, sementara simbol-simbol dalam pemodelan *use case* membantu dalam memvisualisasikan interaksi antara aktor dan fungsionalitas sistem (Khatri, 2018).

II.12.3 Activity Diagram

Activity diagram adalah salah satu jenis diagram yang digunakan dalam pemodelan sistem informasi. Diagram ini digunakan untuk menyajikan aliran proses, interaksi antara komponen sistem, dan penanganan logika bisnis dengan gambar dan lebih mudah dipahami. Berikut beberapa simbol yang digunakan dalam *activity diagram*.

Tabel II-2 Simbol-Simbol *Activity Diagram*

Simbol	Nama	Keterangan
	Status awal	Menggambarkan status awal pada aktivitas.
	Aktivitas	Menandai langkah-langkah yang terjadi dalam suatu proses.
	<i>Swimlane</i>	Digunakan untuk membagi diagram menjadi bagian-bagian yang merepresentasikan entitas atau peran yang berbeda, seperti departemen atau orang yang terlibat dalam suatu proses.
	<i>Decision</i>	Menandai titik dalam diagram di mana keputusan harus dibuat, biasanya direpresentasikan dengan simbol berbentuk berlian.
	Status akhir	Menggambarkan status akhir pada suatu proses.
	<i>Join</i>	Menghubungkan suatu objek dengan objek lainnya.

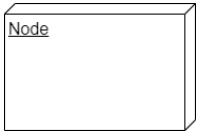
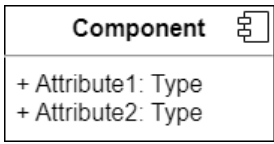

Dalam penjelasan Tabel II-2 simbol-simbol ini digunakan untuk memvisualisasikan aliran proses dan interaksi antara komponen sistem dengan lebih jelas dan mudah dipahami. *Activity diagram* sering digunakan dalam

pengembangan perangkat lunak untuk mengidentifikasi langkah-langkah yang perlu diperbaiki atau menambahkan, serta untuk menggambarkan aliran proses yang tidak jelas atau kompleks (Prediansya & Cahyana, 2023).

II.12.4 Deployment Diagram

Deployment diagram adalah salah satu jenis diagram yang digunakan dalam pemodelan sistem informasi untuk menggambarkan bagaimana komponen sistem ditempatkan pada perangkat keras dan bagaimana komponen tersebut berinteraksi satu sama lain. Berikut beberapa simbol yang digunakan dalam *deployment diagram* (Zahra & Priyadi, 2023).

Tabel II-3 Simbol-Simbol *Deployment Diagram*

Simbol	Deskripsi
	Menunjukkan perangkat keras atau <i>platform</i> yang digunakan untuk menjalankan komponen sistem.
	Menunjukkan bagian dari sistem yang diimplementasikan pada <i>node</i> tertentu.
	Menunjukkan hubungan antara <i>node</i> dan komponen sistem, di mana <i>node</i> digunakan untuk menjalankan komponen sistem.

Dalam penjelasan Tabel II-3 simbol-simbol ini digunakan untuk memvisualisasikan bagaimana komponen sistem ditempatkan pada perangkat keras dan bagaimana komponen tersebut berinteraksi satu sama lain. *Deployment diagram* sering digunakan dalam pengembangan perangkat lunak untuk memastikan bahwa komponen sistem dapat diimplementasikan pada perangkat keras yang tersedia dan berinteraksi dengan benar.

II.12.5 Sequence Diagram

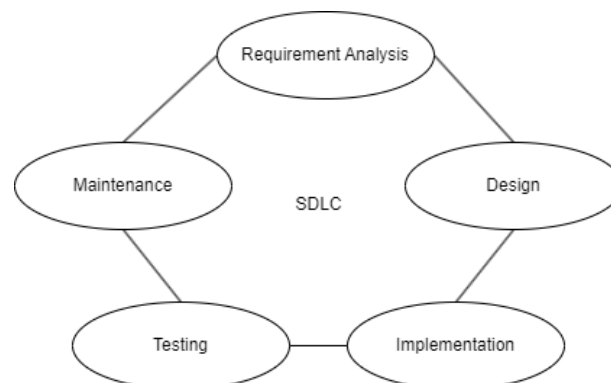
Sequence diagram adalah suatu diagram yang digunakan dalam analisis sistem dan desain sistem untuk menggambarkan interaksi antara objek-objek dalam suatu sistem. Diagram ini menampilkan urutan acara yang terjadi antara objek-objek tersebut dan bagaimana mereka berinteraksi.

Dalam *sequence diagram*, objek-objek diwakili oleh garis-garis vertikal yang disebut *lifelines*. *Lifelines* ini menunjukkan keberadaan objek-objek dalam sistem dan bagaimana mereka berinteraksi. Acara yang terjadi antara objek-objek tersebut digambarkan sebagai kotak-kotak yang berisi teks yang menjelaskan apa yang terjadi.

Dengan menggunakan *sequence diagram*, dapat dengan jelas menggambarkan interaksi antara objek-objek dalam sistem dan bagaimana mereka berinteraksi, sehingga memudahkan dalam analisis dan desain sistem (Ismanto & Pungkasanti, 2021).

II.13 Software Development Life Cycle (SDLC)

Software Development Life Cycle merupakan model pendekatan sistematis dalam proses pengembangan sistem informasi (yas dkk., 2023). Dalam mengembangkan suatu kerangka sistem informasi diperlukan adanya metode yang diharapkan dapat membantu dalam menyelesaikan rancangan sistem informasi yang diharapkan. SDLC sendiri memiliki beberapa tahapan seperti pada Gambar II.7 berikut.



Gambar II.7 Tahapan SDLC

SDLC memiliki 5 tahapan yaitu, fase menganalisis kebutuhan (*requirement analysis*), desain (*design*), implementasi (*implementation*), uji coba (*testing*), dan

pemeliharaan (*maintenance*). Pada SDLC terdapat beberapa model yang umum digunakan diantaranya model *prototype*, *iterative incremental*, *design thinking* dan SCRUM. Pemilihan model SDLC bergantung pada kebutuhan karakteristik pada proyek yang dikembangkan.

II.14 Metode Prototipe

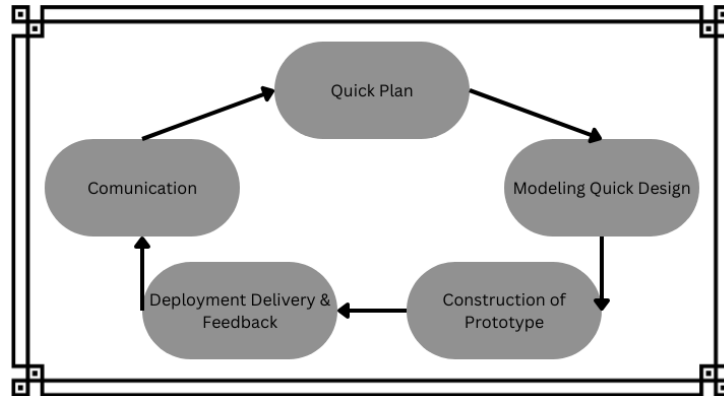
Metode prototipe adalah salah satu teknik dalam pengembangan produk yang digunakan sebagai bagian dari *Software Development Life Cycle* (SDLC). SDLC merupakan kerangka kerja yang mendefinisikan tahapan-tahapan yang dilalui dalam proses pengembangan perangkat lunak, mulai dari perencanaan, analisis, desain, implementasi, hingga pemeliharaan.

Dalam konteks SDLC, metode prototipe menekankan pada pembuatan model awal atau prototipe dari sistem yang akan dikembangkan. Prototipe ini berfungsi sebagai alat untuk menguji dan mengembangkan ide, konsep, dan desain secara iteratif. Pengguna dan pengembang dapat berinteraksi dengan prototipe untuk mengevaluasi fungsionalitas, mengidentifikasi kebutuhan yang belum terpenuhi, dan menemukan masalah atau kekurangan dalam desain awal. Melalui umpan balik yang diperoleh dari pengguna, prototipe dapat disempurnakan secara bertahap hingga memenuhi kebutuhan dan harapan yang diinginkan. Prototipe dapat didefinisikan sebagai model fisik atau virtual dari suatu produk yang belum selesai, yang digunakan untuk menguji fungsionalitas, ergonomi, dan estetika. Metode prototipe memiliki beberapa kelebihan, seperti memungkinkan pengembang untuk menguji konsep dan desain secara langsung, mengurangi risiko, dan memperbaiki desain secara efektif.

Pada Gambar II.8 merupakan tahap-tahap dalam metode prototipe meliputi:

1. Definisikan masalah dan tujuan: Identifikasi masalah yang akan diselesaikan dan tujuan dari prototipe.
2. Desain prototipe: Membuat desain prototipe yang mencakup fungsionalitas, estetika, dan ergonomi.
3. Pembuatan prototipe: Membuat prototipe secara fisik atau virtual.

4. Pengujian prototipe: Menguji fungsionalitas, ergonomi, dan estetika dari prototipe.
5. Perbaikan dan iterasi: Memperbaiki desain berdasarkan hasil pengujian dan mengulangi proses pembuatan prototipe.
6. Validasi: Memeriksa apakah prototipe memenuhi tujuan dan memenuhi kebutuhan pengguna.



Gambar II.8 Tahap-Tahap Prototipe

Metode prototipe dapat digunakan dalam berbagai bidang, seperti desain produk, perangkat elektronik, perangkat lunak, dan arsitektur. Metode ini memungkinkan pengembang untuk mengembangkan desain yang lebih efektif dan efisien, serta memperbaiki desain secara berkala (Hadikusuma & Rahmadewi, 2023).

II.15 Perbandingan Metode

Selain metode prototipe, terdapat metode SDLC lainnya seperti *iterative incremental* dan SCRUM. Tabel II.4 berikut menyajikan perbandingan dari ketiga metode tersebut berdasarkan karakteristiknya.

Tabel II-4 Perbandingan Karakteristik Metode Pengembangan Sistem

Karakteristik	Prototipe	Iterative Incremental	SCRUM
Tujuan Utama	Fokus pada pengembangan prototipe yang dapat memberikan gambaran awal kepada pengguna.	Mencapai hasil yang lebih baik melalui pengulangan dan peningkatan secara bertahap.	Meningkatkan kecepatan dan fleksibilitas pengembangan melalui iterasi pendek (<i>sprint</i>).
Proses Pengembangan	Dapat digunakan untuk mengidentifikasi kebutuhan dan perubahan desain.	Proses pengembangan melibatkan iterasi yang berulang dan penambahan fungsionalitas secara bertahap.	Proses pengembangan melibatkan <i>sprint</i> dengan fokus pada pengiriman nilai bisnis yang tinggi.
Pengujian	Pengujian umumnya terbatas pada validasi konsep dan fitur yang mendasar.	Pengujian dilakukan secara berulang pada setiap iterasi.	Pengujian terintegrasi dalam setiap <i>sprint</i> .
Pengelolaan Risiko	Risiko dikurangi melalui validasi konsep sejak awal.	Pengelolaan risiko terintegrasi dalam proses pengembangan.	Risiko dikelola melalui transparansi dan inspeksi terus-menerus selama <i>sprint review</i> .

Karakteristik	Prototipe	Iterative Incremental	SCRUM
Cocok Diimplementasikan Pada Pengembangan	Cocok untuk proyek dengan kebutuhan yang mungkin belum jelas dan perlu validasi cepat seperti pengembangan perangkat IoT.	Cocok untuk proyek besar dengan kemampuan untuk melihat kemajuan secara berkala seperti pengembangan 2 <i>scope</i> atau lebih.	Cocok untuk proyek yang membutuhkan fleksibilitas tinggi dan tanggapan cepat terhadap perubahan kebutuhan seperti pembuatan aplikasi, dan <i>website</i> .

Berdasarkan Tabel II-4 metode prototipe cocok digunakan dalam pengembangan alat berbasis *Internet of Things* (IoT) dan aplikasi *smart dashboard* karena lebih fleksibel dan manajemen risikonya sedang.

II.16 Pengujian Kalibrasi

Pengujian kalibrasi adalah prosedur penting dalam laboratorium pengujian dan kalibrasi untuk memastikan bahwa instrumen pengukuran atau alat ukur yang digunakan memenuhi standar kualitas yang ditetapkan. Proses ini melibatkan pengecekan dan penyesuaian instrumen untuk memastikan bahwa pembacaannya akurat dan konsisten dengan standar yang diketahui. Tanpa kalibrasi yang tepat, hasil pengukuran dapat menjadi tidak dapat diandalkan, yang dapat mempengaruhi kualitas produk dan keputusan yang dibuat berdasarkan data tersebut.

Dengan melakukan pengujian kalibrasi secara rutin, laboratorium dapat memastikan bahwa instrumen pengukurannya tetap akurat dan dapat diandalkan. Hal ini penting untuk memenuhi persyaratan kualitas yang diperlukan dalam berbagai industri dan aplikasi. Hasil pengujian yang akurat sangat penting untuk

memastikan bahwa produk memenuhi spesifikasi dan standar yang telah ditetapkan, serta untuk menjaga kepercayaan pelanggan dan mematuhi peraturan industri (Sardjono dkk., 2021).

II.17 Penelitian Terdahulu

Penelitian terdahulu yang relevan dengan pengembangan sistem *monitoring* dan kontrol suhu kolam atau akuarium pada budidaya ikan koi berbasis *Internet of Things* (IoT) dengan penerapan kontrol PID disajikan dalam Tabel II-5. Hasil temuan tersebut membentuk dasar acuan bagi penulis sebagai sumber referensi dalam merinci konsep dan implementasi pengembangan sistem yang sedang dikembangkan. Penulis dapat menggali wawasan yang mendalam terkait dengan aspek-aspek tertentu dari proyek ini, serta memperkaya landasan teoritis dan metodologis yang mendasari pengembangan sistem pada penelitian ini.

Tabel II-5 Perbandingan Penelitian Terdahulu

No.	Judul	Penulis	Hasil
1.	Sistem <i>Monitoring</i> Ph Dan Suhu Air Pada Tambak Udang Menggunakan Protokol Websocket	Adhi Kurniawan dan Heru Nurwasito (2019)	<ol style="list-style-type: none"> 1. Artikel ini memberikan deskripsi yang jelas dan terperinci tentang desain sistem, termasuk sensor <i>node</i>, <i>server</i>, <i>database server</i>, dan antarmuka web. 2. Akurasi sensor dalam mendapatkan informasi pH dan suhu, dengan tingkat akurasi yang mengesankan sebesar 99% dan tingkat kesalahan di bawah 1%. 3. Pengembangan dan implementasi sistem pemantauan waktu nyata untuk kualitas air di tambak udang.
2.	Rancang Bangun Sistem <i>Monitoring</i> PH dan Suhu Air pada Akuaponik Berbasis <i>Internet of Thing</i> (IoT)	Dini Megawati, Kholidiyah Masykuroh, dan Danny Kurnianto (2020)	<ol style="list-style-type: none"> 1. Alat yang dikembangkan memiliki akurasi pengukuran yang tinggi dan sampel air memiliki kualitas suhu dan TDS yang baik. 2. Sistem yang dikembangkan memiliki akurasi pengukuran yang tinggi, berkisar antara 98,28% hingga 100%. 3. Sistem pemantauan berbasis IoT dapat menjadi alat yang dapat diandalkan untuk menilai kualitas air.
3.	Implementasi <i>Internet Of Aquaculture</i> untuk deteksi Kualitas Lingkungan Secara Cepat dalam Upaya Pemberdayaan	Yuni Kilawati, Yunita Maimunah, dan Adharul Muttaqin (2020)	<ol style="list-style-type: none"> 1. Sistem ini menggunakan 4 jenis sensor, yaitu sensor pH, suhu, <i>turbidity</i>, dan <i>dissolved oxygen</i>. 2. Peminat atau penghobi ikan koi meningkat setiap tahunnya, sehingga permintaan akan ikan koi juga meningkat pesat. 3. Aplikasi IoT dalam implementasi ini memberikan visualisasi data dan informasi kepada pengguna dengan cara yang intuitif dan mudah dipahami.

No.	Judul	Penulis	Hasil
	Kelompok Pembudidayaan Ikan Koi di Blitar		4. Standar kualitas air untuk pemeliharaan ikan koi telah dirumuskan dalam Standar Nasional Indonesia (SNI).
4.	Perancangan Dan Implementasi Alat <i>Monitoring</i> Dan <i>Controlling</i> Kualitas Air Pada Kolam Ikan Koi	Riezky Fakhriza, Basuski Rahmat, dan Sri Astuti (2021)	<ol style="list-style-type: none"> 1. Sistem ini menggunakan mikrokontroler untuk membaca dan menyimpan data input dari sensor suhu, pH, dan kadar amonia air. 2. Sensor suhu DS18B20 memiliki tingkat akurasi sebesar 98,01% bila dibandingkan dengan termometer. 3. Sistem ini memiliki dua mode, yaitu otomatis dan manual, dan dapat dikendalikan melalui aplikasi pada <i>smartphone</i> dengan menggunakan protokol MQTT. 4. Sistem ini menggunakan fitur WiFi pada NodeMCU untuk terhubung dengan aplikasi pada <i>smartphone</i> yang menggunakan sistem operasi android. 5. Telah dilakukan pengujian terhadap sensor suhu, pompa, <i>water heater</i>, dan <i>cooler</i>.
5.	Sistem Pmantau Kualitas Air Kolam Ikan Koi	Dian Ariyanto, dan Medilla Kusriyanto (2023)	<ol style="list-style-type: none"> 1. Menggunakan mikrokontroler ESP32 sebagai pusat sistem. 2. Terdapat tiga sensor yang digunakan, yaitu sensor pH, sensor suhu DS18B20, dan sensor TDS. 3. Data dikirim ke Blynk 4. Sistem ini juga terhubung dengan <i>output relay</i> yang mengontrol keran pengisian kolam dan keran pengurasan bak filter air. 5. Tempat sistem yang menggunakan 3D <i>printing</i>.

No.	Judul	Penulis	Hasil
6.	Sistem Pengatur Suhu Dan PH Air Aquarium Otomatis Dengan Metode <i>Fuzzy Logic</i> Berbasis NodeMCU	Aldi Triavin Dwi Putra, dan Mardi Hardjianto (2023)	<ol style="list-style-type: none"> 1. Mikrokontroler yang digunakan yaitu NodeMCU ESP8266. 2. Menggunakan 2 sensor yaitu sensor pH 4502C dan sensor suhu DS18B20. 3. Data dikirim ke <i>database</i> lokal yang tersedia. 4. Nilai pH yang terbaca sensor akan ditentukan nilai keanggotaan <i>fuzzy</i> oleh NodeMCU, yang akan menentukan durasi pompa untuk membuang dan menambahkan air pada akuarium untuk menyeimbangkan pH. 5. Menggunakan metode <i>Fuzzy Set</i> untuk menentukan <i>output</i> berdasarkan <i>input</i> dari sensornya.
7.	<i>Prototype</i> Pengatur Kadar PH dan Pemeberian Pakan Ikan Koi Secara Otomatis Menggunakan Mikrokontroller	Dhana, dan Hilda Nur Rama (2021)	<ol style="list-style-type: none"> 1. Mengembangkan sebuah purwarupa pengatur kadar pH dan pemberian pakan ikan koi secara otomatis. 2. Menggunakan sensor tingkat keasaman dan sensor ketinggian air. 3. Larutan asam memiliki pH lebih rendah, sedangkan larutan basa memiliki pH lebih tinggi. Nilai pH bisa kurang dari 0 untuk asam yang sangat kuat, atau lebih besar dari 14 untuk basa yang sangat kuat 5 4. 4. Hasil pemantauan sistem telah berhasil ditampilkan pada <i>website</i> IoT Cloud. 5. Komponen yang dipakai antara lain Mikrokontroller ATmega328P Arduino Uno R3, Sensor <i>Ultrasonic Range Finder</i> HC-SR04 (ketinggian Air), Sensor <i>Probe</i> PH-4502C <i>Signal Amplifier</i> (keasaman air), <i>1st Channel of 4 Channel</i> SRD-05VDC-SL-C <i>Relay Module</i> (Pompa penguras), <i>2nd Channel of 4 Channel</i> SRD-05VDC-SL-C <i>Relay Module</i> (Pompa pengganti ait), RTC DS1307 (modul RTC).

No.	Judul	Penulis	Hasil
8.	Rancang Bangun Sistem Otomatisasi Suhu Dan Monitoring pH Air Aquascape Berbasis IoT (<i>Internet of Thing</i>) Menggunakan NodemcuEsp8266 Pada Aplikasi Telegram	Dendy Ramdani, Fahrudin Mukti Wibowo, dan Yoso Adi Setyoko (2020)	<ol style="list-style-type: none"> 1. Menggunakan sensor DS18B20 untuk mendeteksi suhu dan pH <i>Detection Sensor Module</i> untuk mendeteksi tingkat keasaman air dan mikrokontroler seri Node MCU ESP8266. 2. Ketentuan parameter pada suhu air <i>aquascape</i> yaitu 25°C-28°C dan nilai kadar pH antara 6,0-8,0. 3. Menggunakan aplikasi bot Telegram untuk mengirimkan hasil data dari pengujian <i>monitoring</i> pH air <i>aquascape</i> secara <i>real-time</i>. 4. Menggunakan NTP <i>client server</i> untuk memberikan penjadwalan yang <i>real-time</i>. 5. Hasil dari penelitian berupa alat yang mampu melakukan proses otomatisasi suhu dan <i>memonitoring</i> nilai pH menggunakan sensor pH pada Bot telegram 6. Kesimpulannya jika sensor suhu menyatakan nilai kurang dari 25°C maka lampu akan otomatis menyala dan sebaliknya apabila sensor mendeteksi suhu lebih dari 28°C maka <i>cooling fan</i> akan menyala.
9.	Sistem <i>Monitoring</i> Kualitas Air Tambak Ikan Koi (<i>Cyprinus</i>)	Sofiana Yuli Damayanti, Teguh Andriyanto, dan	<ol style="list-style-type: none"> 1. Hasil penelitian berupa aplikasi <i>monitoring</i> Ph dan suhu air, mengirimkan notifikasi, dan pengendalian <i>relay</i> untuk menguras dan mengisi air secara <i>real-time</i>.

No.	Judul	Penulis	Hasil
	Carpio) Berbasis Teknologi <i>Internet of Things</i> (IoT)	Aidina Ristywan (2021)	<ol style="list-style-type: none"> 2. Menggunakan Mikrokontroler ESP32, sensor Suhu DSB1820, sensor PH air, LCD, dan aplikasi Blynk serta Thingspeak untuk <i>output</i> data. 3. Aplikasi Thingspeak yang diperuntukan bagi pengguna Android/IOS bernama Thingview. Aplikasi ini hanya dapat membaca sensor Ph dan suhu air tanpa dapat melihat penyimpanan datanya. 4. Aplikasi Blynk menjadi <i>interface</i> untuk <i>memonitoring</i>, menerima notifikasi dan mengontrol <i>relay</i>. 5. Alat ini dikemas menggunakan box akrilik, dengan tambahan stop kontak untuk memberi daya pompa air.
10.	Sistem <i>Monitoring</i> Suhu Air pada Kolam Benih Ikan Koi Berbasis <i>Internet of Things</i>	Slamet Indriyanto, Fikra Titan Syifa, dan Hanif Aditya Permana (2020)	<ol style="list-style-type: none"> 1. Suhu sangat berpengaruh terhadap proses metabolisme, dan proses metabolisme akan berpengaruh terhadap pertumbuhan ikan. 2. Sistem yang dirancang terdiri dari NodeMCU ESP8266, ESP 8266 sensor suhu DS18B20, <i>relay</i> 1 <i>channel</i>, <i>heater</i>, dan satu daya. 3. <i>Board</i> untuk menerima data sensor suhu DS18B20 4. Sebagai <i>output</i>, <i>board</i> mengendalikan <i>relay</i> untuk menyalakan dan mematikan <i>heater</i>. 5. Ketika sensor suhu mendeteksi suhu di bawah normal yaitu kurang dari 25°C maka <i>heater</i> akan aktif, tapi ketika sensor suhu mendeteksi suhu atas normal yaitu lebih dari 27°C maka otomatis <i>heater</i> akan mati. Data suhu yang didapatkan akan dikirim ke <i>platform</i> Thingspeak melalui jalur internet agar kondisi suhu pada kolam budidaya dapat di-<i>monitor</i> secara langsung oleh pengguna.