

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Penelitian ini dilakukan oleh Nur Azizah tahun 2021 dengan judul “PENYIRAMAN DAN PEMUPUKAN TANAMAN BAWANG MERAH SECARA OTOMATIS PADA GREENHOUSE MENGGUNAKAN INTERNET OF THINGS (IOT)”. Penelitian ini mengembangkan sistem penyiraman dan pemupukan otomatis untuk tanaman bawang merah dalam greenhouse berbasis Internet of Things (IoT) menggunakan Arduino Uno dan NodeMCU ESP8266. Sistem ini memanfaatkan sensor kelembaban tanah (Soil Moisture Sensor) untuk mendeteksi kondisi tanah, serta Real Time Clock (RTC) untuk mengatur jadwal pemupukan otomatis setiap Senin pukul 10.00 WIB selama 60 hari. Notifikasi dikirim ke pengguna melalui aplikasi Blynk untuk pemantauan jarak jauh, sementara LCD digunakan untuk monitoring langsung. Hasil pengujian menunjukkan sistem bekerja dengan baik dalam mengoptimalkan penyiraman dan pemupukan, sehingga dapat meningkatkan efisiensi perawatan tanaman bawang merah dalam greenhouse[6].

Penelitian ini dilakukan oleh Muhammad Brian Fatkhul Yaqin¹, Bagus Setya Rintyarna, Herry Setyawan tahun 2024 dengan judul “RANCANG BANGUN PROTOTIPE SMART GREENHOUSE BERBASIS IOT UNTUK MENGONTROL PERTUMBUHAN TANAMAN BAWANG MERAH (ALLIUM ASCALONICUM L)”. Penelitian ini mengembangkan

prototipe smart greenhouse berbasis Internet of Things (IoT) untuk mengontrol pertumbuhan tanaman bawang merah dengan memanfaatkan sensor DHT22 untuk suhu, soil moisture YL-69 untuk kelembaban tanah, sensor pH untuk mengontrol nutrisi tanah, RTC DS3231 untuk penjadwalan penyemprotan, dan servo SG-90 untuk pemberian obat. Sistem ini menggunakan mikrokontroler Arduino dan ESP32, serta dapat diakses melalui aplikasi Blynk untuk pemantauan dan kontrol jarak jauh. Hasil pengujian menunjukkan bahwa sistem bekerja dengan tingkat error sensor yang rendah (DHT22: 0,5%, soil moisture: 0,8%, pH: 0,39%) dan tingkat keberhasilan IoT sebesar 100%, sehingga dapat membantu petani meningkatkan efisiensi perawatan bawang merah dalam berbagai kondisi cuaca[7].

Penelitian ini dilakukan oleh Vicro Zulif Nufusu mahasiswa politeknik harapan bersama tahun 2021 dengan judul ANALISIS DAN PERANCANGAN SISTEM PENYIRAMAN TANAMAN OTOMATIS BERBASIS INTERNET OF THINGS. Sebagai media penyiraman yang mempermudah pekerjaan dengan cara kerja yang otomatis dengan menggunakan Arduino Uno R3 dan memanfaatkan sensor ultrasonic untuk mendeteksi keberadaan suatu benda dan memperkirakan jaraknya dan dipantau menggunakan aplikasi berbasis website. Dengan digunakannya alat ini dapat mempermudah seorang pekerja kebun maupun pembudidaya tanaman dalam merawat tanamannya agar tetap terjaga dengan baik[8].

Penelitian ini dilakukan oleh I Wayan Wikananda Adikara, Agus Muliantara, mahasiswa Universitas Udayana Bali tahun 2023 dengan judul RANCANG ALAT PENYIRAMAN DAN PEMUPUKAN TANAMAN BERBASIS IOT. Sebagai media penyiraman dan pemupukan untuk mempermudah pekerjaan dengan cara kerja yang otomatis dengan menggunakan Arduino Uno R3 dan memanfaatkan sensor Soil Moisture untuk mendeteksi kelembapan tanah dan menggunakan RTC untuk pengaturan waktu. Dengan digunakannya alat ini dapat mempermudah pekerjaan dalam merawat tanamannya agar tetap terjaga dengan baik[9]

Penelitian yang dilakukan oleh Renata Eka Budiani, Joseph Dedy Irawan , Deddy Rudhistiar tahun 2024 dalam penelitian nya yang berjudul “SISTEM MONITORING DAN PENYIRAMAN OTOMATIS PADA TANAMAN CABAI BERBASIS INTERNET OF THINGS (IOT)”. Penelitian ini membahas tentang bagaimana memonitoring penyiraman otomatis pada tanaman cabai dengan menggunakan android di Telegram[12]

Penelitian yang dilakukan oleh Gerald Rhamadhany, Noni Juliasari tahun 2024 dalam penelitian nya yang berjudul “Rancang Bangun Prototipe Sistem Monitoring Pemupukan Dan Penyiraman Tanaman Otomatis Berbasis Internet of Things”. Penelitian ini membahas tentang penggunaan teknik prototyping. Metode prototyping ini memiliki beberapa tahapan untuk menggunakan metode ini, yang dapat dilakukan sesuai dengan keinginan pengguna. Setelah melalui tahapan-tahapan tersebut, pengguna harus menjalankan perintah yang diinginkan menggunakan aplikasi yang

disediakan atau alamat situs web server lokal. Alamat situs web memungkinkan pengguna untuk memantau irigasi dan pemupukan otomatis[13]

Penelitian yang dilakukan oleh Doni, dkk (2020) tentang Sistem Monitoring Tanaman Hidroponik Berbasis Iot (Internet of Thing) Menggunakan Nodemcu ESP8266. Metode yang digunakan adalah Fuzzy Sugano dan hasil pengujian yang dilakukan menunjukkan bahwa sistem dapat bekerja dengan baik. Hal ini ditunjukkan dengan kemampuan sistem menyiram secara otomatis berdasarkan output yang diperoleh dari sensor DHT11, akan tetapi sistem ini belum bisa melakukan monitoring[14]

2.2 Landasan Teori

2.2.1 Arduino Software IDE

Arduino IDE (*Integrated Development Environment*) merupakan suatu aplikasi *open source* sebagai *text editor* untuk membuat, membuka, mengedit, dan mevalidasi kode serta melakukan upload ke *board* arduino atau mikrokontroler lainnya. Program yang digunakan pada Arduino IDE disebut dengan istilah “*sketch*” yaitu *file source code* Arduino dengan ekstensi.ino. Arduino menggunakan bahasa pemrograman sendiri yang menyerupai bahasa C. Bahasa di dalam Arduino (*Sketch*) sudah dilakukan perubahan untuk memudahkan bagi pemula. Di dalam IC mikrokontroler Arduino telah terdapat program bernama *Bootloader* yang berfungsi sebagai penengah antar *compiler* Arduino dengan mikrokontroler.



Gambar 2. 1 Arduino IDE

2.2.2 MYSQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL atau DBMS yang *multithread*, *multi-user*. Sebagai *database server*, MySQL dapat dikatakan lebih unggul dibandingkan *database server* lainnya dalam *query* data. Hal ini terbukti untuk *query* yang dilakukan oleh *single user*, kecepatan *query* MySQL bisa sepuluh kali lebih cepat dari PostgreSQL dan lima kali lebih cepat dibandingkan Interbase[3].



Gambar 2. 2 My SQL

2.2.3 HTML

HyperText Markup Language (HTML) adalah suatu bahasa yang digunakan untuk menulis halaman web dan menampilkan berbagai

informasi didalam sebuah browser internet. HTML ditulis dengan bentuk dokumen ASCII atau text biasa. Dengan kata lain, berkas yang dibuat dalam perangkat lunak pengolah kata dan disimpan dalam format ASCII normal, yang dirancang untuk tidak tergantung pada suatu sistem informasi tertentu. HTML merupakan pengembangan dari standar pemformatan dokumen text, yaitu *Standard Generalized Markup Language* (SGML). HTML saat ini merupakan standar internet yang didefinisikan dan dikendalikan oleh *World Wide Web Consortium* [2].



Gambar 2. 3 HTML

2.2.4 PHP

Menurut ID CloudHost, Laravel adalah sebuah framework PHP yang dirilis dibawah lisensi MIT, dibangun dengan konsep MVC (model view controller). Laravel adalah pengembangan website berbasis MVP yang ditulis dalam PHP yang dirancang untuk meningkatkan kualitas perangkat lunak dengan mengurangi biaya pengembangan awal dan biaya pemeliharaan, dan untuk meningkatkan pengalaman bekerja dengan aplikasi dengan menyediakan sintaks yang ekspresif, jelas dan menghemat waktu[11]. MVC adalah sebuah pendekatan perangkat lunak yang memisahkan aplikasi logika dari presentasi. MVC

memisahkan aplikasi berdasarkan komponen- komponen aplikasi, seperti : manipulasi data, controller, dan user interface.



Gambar 2. 4 LARAVEL

2.2.5 UML

UML (*Unified Modeling Language*) adalah metode pemodelan secara visual sebagai sarana untuk merancang dan atau membuat software berorientasi objek. Karena UML ini merupakan bahasa visual untuk pemodelan bahasa berorientasi objek, maka semua elemen dan diagram berbasiskan pada paradigma object oriented. UML sendiri juga memberikan standar penulisan sebuah sistem blue print, yang meliputi konsep bisnis proses, penulisan kelas-kelas dalam bahasa program yang spesifik[1].

Beberapa diagram yang digunakan di UML (*Unifed Modeling Language*) :

1. Use Case Diagram

Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah use case

mempresentasikan sebuah interaksi antara aktor dengan sistem. Use case merupakan sebuah pekerjaan tertentu, misalnya login ke sistem, meng-*create* sebuah daftar belanja, dan sebagainya. Seorang atau sebuah aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan - pekerjaan tertentu.

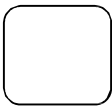
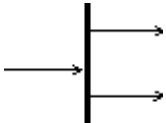
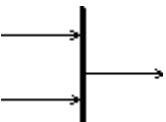
Tabel 2. 1 Use Case Diagram

Gambar	Nama	Keterangan
	Actor	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
	Association	Apa yang menghubungkan antara objek satu dengan objek lainnya.
	System	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
	Use Case	Deskripsi dari urutan aksi – aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.

2. Activity Diagram

Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing - masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity* diagram juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. *Activity* diagram merupakan state diagram khusus, dimana sebagian besar *state* adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya *state* sebelumnya (*internal processing*). Oleh karena itu *activity* diagram tidak menggambarkan behaviour internal sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses - proses dan jalur - jalur aktivitas dari level atas secara umum. Sebuah aktivitas dapat direalisasikan oleh satu *use case* atau lebih. Aktivitas menggambarkan proses yang berjalan, sementara *use case* menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktifitas. *Decision* digunakan untuk menggambarkan *behaviour* pada kondisi tertentu. Untuk mengilustrasikan proses - proses paralel (*fork* dan *join*) digunakan titik sinkronisasi yang dapat berupa titik, garis horizontal atau vertikal.

Tabel 2. 2 Activity Diagram

Gambar	Nama	Keterangan
	<i>Activity</i>	Memperlihatkan bagaimana masing – masing kelas antarmuka saling berinteraksi satu sama lain.
	<i>Action</i>	<i>State</i> dari sistem yang mencerminkan eksekusi suatu aksi.
	<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
	<i>Final Node</i>	Bagaimana objek dibentuk dan dihancurkan.
 	<i>Fork/Join</i>	Digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu

3. Sequence Diagram

Diagram sequence dimaksudkan untuk mengembangkan komunikasi antara objek, bukan memanipulasi data saat berkomunikasi.

Tabel 2. 3 Sequence Diagram

No.	Simbol	Nama	Keterangan
1.		<i>Life Line</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
2.		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi- informasi tentang aktifitas yang terjadi
3.		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi tentang aktifitas yang terjadi

4. Class Diagram

Diagram kelas adalah inti dari proses pemodelan objek, baik *forward engineering* maupun *reverse engineering* memanfaatkan diagram ini. *Forward engineering* adalah proses perubahan model menjadi kode program sedangkan *reverse engineering* sebaliknya, merubah kode program menjadi model.

Diagram kelas memiliki fitur-fitur yang menyertainya yaitu atribut dan operasi. Atribut (attribute) dan operasi (operation) menggambarkan perilaku suatu kelas serta perluasannya seperti stereotypes, tagged values, dan batasan (constraints) merupakan fitur- fitur sebuah kelas.

Tabel 2. 4 Class Diagram

No.	Simbol	Nama	Keterangan
1.		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
2.		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3.		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4.		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
5.		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.

6.		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan memengaruhi elemen yang tidak mandiri.
7.		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.