BAB II

TINJAUAN PUSTAKA

2.1 Teori Terkait

Pada penelitian yang dilakukan oleh F. D. Silalahi dkk. (2021) dalam jurnalnya yang berjudul *"Implementasi Internet of Things (IoT) Dalam Monitoring Suhu dan Kelembaban Ruang Produksi Obat Non Steril Menggunakan Arduino Berbasis Web"*, membahas sistem monitoring suhu berbasis web menggunakan teknologi IoT. Penelitian ini bertujuan memberikan akses *real-time* bagi pengguna dalam memantau suhu dan kelembaban. Sistem ini menggunakan sensor DHT11 dan DHT22 yang dikoneksikan ke Arduino, lalu data dikirim ke *Database* MySQL melalui ESP8266. Data ditampilkan melalui antarmuka web berbasis PHP dan JavaScript. Dengan sistem ini, pengguna dapat mengakses informasi suhu dan kelembaban dari berbagai perangkat serta mendapatkan peringatan otomatis jika nilai di luar batas normal. Hasil penelitian menunjukkan sistem ini bekerja dengan baik dalam menampilkan data secara akurat serta memberikan notifikasi *real-time* [5].

Pada penelitian lain yang dilakukan oleh Diah Ajeng Setiawati dkk. (2020) dalam artikel yang berjudul "Rancang Bangun Sistem Pengendalian Suhu Ruang Pengering *Hybrid* Tipe Rak Berputar Berbasis Mikrokontroler Arduino Mega", dijelaskan mengenai pengembangan sistem pengendalian suhu pada mesin pengering *hybrid* tipe rak berputar. Penelitian ini bertujuan

untuk memastikan suhu ruang pengering tetap stabil guna mendukung efisiensi dan kualitas proses pengeringan. Hasil penelitian menunjukkan bahwa sistem yang dirancang dapat menjaga suhu rata-rata ruang pengering mendekati titik setel (*setpoint*), dengan nilai suhu rata-rata sebesar 46,24°C pada uji tanpa bahan dan 46,99°C pada uji dengan bahan. Selain itu, kecepatan kipas dan blower juga stabil mendukung pengendalian suhu yang optimal. Penelitian ini menyimpulkan bahwa sistem pengendalian suhu yang dirancang mampu bekerja dengan baik, serta dapat diterapkan untuk meningkatkan efisiensi dalam pengeringan bahan berbasis mesin *hybrid* tipe rak berputar. Hal ini menunjukkan potensi besar untuk diaplikasikan dalam proses pengeringan hasil pertanian dengan kebutuhan suhu yang terkontrol [6].

Penelitian oleh Syamsiana dkk. (2022) berjudul "Implementasi Sistem Monitoring Alat Pengering Biji Kopi Berbasis IoT (Internet of Things)" bertujuan mengembangkan alat pengering kopi dengan sistem monitoring berbasis IoT. Alat ini dilengkapi sensor suhu dan arus listrik untuk memantau kondisi pengering secara *real-time*. Sistem monitoring berbasis IoT memungkinkan pengguna melihat data suhu dan konsumsi daya melalui dashboard website. Hasil penelitian menunjukkan sistem mampu bekerja dengan baik, memberikan data akurat dan mudah diakses, serta membantu proses pengeringan lebih efisien. Penelitian ini menyimpulkan bahwa integrasi IoT pada alat pengering kopi dapat meningkatkan kontrol, efisiensi, dan kualitas hasil pengeringan [7].

2.2 Landasan Teori

2.2.1 Rempah Jamu

Rempah jamu adalah bahan alami yang digunakan dalam pembuatan jamu tradisional untuk menjaga kesehatan dan mengobati berbagai penyakit. Rempah-rempah ini umumnya berasal dari tanaman herbal yang memiliki kandungan zat aktif bermanfaat bagi tubuh. Jahe (*Zingiber officinale*) Jahe merupakan salah satu rempah yang banyak digunakan dalam pembuatan jamu karena khasiatnya yang tinggi. Jahe mengandung senyawa *gingerol* yang memiliki sifat antiinflamasi dan antioksidan, sehingga bermanfaat untuk meningkatkan daya tahan tubuh, meredakan mual, serta membantu pencernaan. Selain itu, jahe juga sering digunakan untuk kesehatan. Kandungan aktif dalam jahe memiliki efek antiinflamasi, antibakteri, dan antioksidan. Dalam pembuatan jamu, jahe sering dimanfaatkan untuk meredakan masalah pencernaan, meningkatkan daya tahan tubuh, dan menghangatkan badan. Jahe juga membantu mengurangi mual, nyeri otot, serta mempercepat pemulihan tubuh [8].

2.2.2 Sistem

Sistem adalah suatu kesatuan yang terdiri dari elemen-elemen yang saling berhubungan dan bekerja bersama untuk mencapai tujuan tertentu. Setiap elemen dalam sistem memiliki peran yang mendukung fungsi keseluruhan. Sistem dapat ditemukan dalam berbagai bidang,

baik teknologi maupun kehidupan sehari-hari, dan digunakan untuk menyederhanakan proses, meningkatkan efisiensi, serta membantu dalam mencapai hasil yang diinginkan[9].

2.2.3 *Monitoring*

Monitoring adalah proses mengamati dan mencatat suatu keadaan atau aktivitas secara berkala untuk mengetahui perubahan yang terjadi. Tujuannya agar informasi penting bisa diperoleh secara tepat waktu. *Monitoring* dapat dilakukan secara manual atau otomatis, dan digunakan dalam berbagai bidang, seperti lingkungan, kesehatan, dan teknologi, untuk membantu pengambilan keputusan berdasarkan data yang dikumpulkan[10].

2.2.4 *Internet of Things* (IoT)

*Internet of Things (*IoT*)* merupakan suatu sistem yang memungkinkan berbagai objek dan individu terhubung dalam suatu jaringan, sehingga dapat bertukar data secara otomatis tanpa memerlukan interaksi langsung antar manusia yaitu sumber ke tujuan atau interaksi manusia ke komputer. *IoT* merupakan perkembangan teknologi yang menjanjikan, *IoT* dapat mengoptimalkan kehidupan dengan sensor sensor cerdas dan benda yang memiliki jaringan dan bekerjasama dalam internet [11].

2.2.5 *Visual Studio Code*

Visual Studio Code adalah editor teks sumber terbuka yang disediakan secara gratis oleh *Microsoft*. Aplikasi ini bersifat lintas platform, sehingga dapat digunakan di berbagai sistem operasi seperti *Windows*, *Linux*, dan *macOS*. Visual Studio Code adalah aplikasi yang sangat kuat yang, meskipun ringan, menawarkan banyak kemampuan. Editor ini mendukung berbagai bahasa pemrograman, *termasuk Java*, *JavaScript*, *C++*, dan banyak lainnya. Visual Studio Code juga dilengkapi dengan berbagai fitur berguna seperti penyorotan sintaks, penyelesaian kode, potongan kode, refaktorisasi, *debugging*, dan integrasi *Git*. Seperti editor kode lainnya, misalnya *sublime* atau atom. Visual studio code dapat mengenali berbagai bahasa pemrograman dan secara otomatis menyesuaikan skema warna sesuai dengan fungsi dalam kode yang sedang ditulis. Ini menjadikannya aplikasi yang sangat efisien bagi para pengembang *website software*, maupun aplikasi lainnya yang kompleks [12].

2.2.6 *Arduino 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. Arduino memanfaatkan bahasa pemrograman yang mirip dengan C, tetapi dirancang khusus

untuk memudahkan pengembangan perangkat keras sehingga mudah dipelajari, terutama bagi pemula. Arduino IDE juga menyediakan banyak contoh kode dan *library* yang siap pakai, memudahkan pengguna dalam membuat proyek elektronik [13].

2.2.7 *Database*

Database adalah sekumpulan data dan informasi yang disusun Secara teratur dalam suatu tempat penyimpanan, memungkinkan kita mengaksesnya dengan mudah melalui program komputer. *Database* hadir untuk membantu kita mengelola data dengan lebih efektif, memeriksa, memproses, atau mengolahnya agar dapat menghasilkan informasi yang bermanfaat. Pada intinya, *Database* adalah kumpulan data yang saling terhubung, dirancang untuk memudahkan kehidupan kita dalam mengatur dan memahami informasi secara lebih terstruktur dan efisien [14].

2.2.8 *Bootstrap*

Bootstrap adalah *framework CSS* yang dirancang untuk membantu pengembang dalam membuat *website* yang responsif dan memiliki tampilan menarik dengan lebih mudah. Ketidakonsistenan terhadap aplikasi individual membuat pengembangan dan pemeliharaannya menjadi sulit. *Bootstrap* adalah css tetapi dibentuk

dengan *LESS*, sebuah *pre-prosessor* yang memberi fleksibilitas dari *css* biasa.

2.2.9 *Laragon*

Laragon merupakan perangkat lunak yang dibuat untuk mempermudah proses pembuatan serta pengelolaan server lokal. *Laragon* menawarkan solusi bagi *web developer* yang memerlukan lingkungan pengembangan yang efisien, stabil, dan fleksibel.

2.2.10 *PhpMyAdmin*

PhpMyAdmin adalah perangkat lunak bebas yang ditulis dalam bahasa pemrograman PHP yang digunakanuntuk menangani administrasi MySQL melalui Jejaring Jagat Jembar *(World Wide Web*). phpMyAdmin mendukungberbagai operasi MySQL, diantaranya (mengelola basis data, tabel-tabel, bidang (*fields*), relasi (*relations*), indeks, pengguna (users), perizinan (permissions), dan lain-lain. Antarmukanya yang sederhana memudahkan pengguna dalam mengelola *Database* dengan cepat dan efisien [15].

2.2.11 *MySQL*

MySQL merupakan sistem manajemen basis data relasional (DBMS) yang bersifat open *source*. Sistem ini awalnya dikembangkan oleh sebuah perusahaan asal Swedia pada sekitar tahun 1994-1995. Meskipun demikian, dasar pengembangan kode untuk *MySQL* sudah ada sejak tahun 1979. dan saat ini *MySQL* sudah diambil alih oleh *Oracle Corp. MySQL* didistribusikan dengan *Licensi open source GPL (General Publik License). MySQL* banyak digunakan karena cepat, andal, dan mudah dintegrasikan dengan bahasa pemrograman seperti *php, python*, dan *java*, sehingga mendukung pengembangan aplikasi *web* [16].

2.2.12 *HTML*

HyperText Markup Language (HTML) adalah suatu bahasa yang digunakan untuk menulis halaman web dan menampilkan berbagai informasi didalam sebuah browser internet. HTML ditulis dalam format dokumen ASCII atau teks biasa. Artinya, file yang dibuat menggunakan perangkat lunak pengolah kata dapat disimpan dalam format ASCII standar, yang dirancang agar tidak bergantung pada sistem informasi tertentu.

2.2.13 *CSS*

Cascading Style Sheets atau *CSS* adalah suatu bahasa lembar gaya yang digunakan untuk mengaturtampilan suatu dokumen yang ditulis dalam bahasa markah. *CSS* paling sering digunakan untuk memformat halaman *web* yang dibuat dengan *HTML* dan *XHTML*. Namun, *CSS* juga dapat diterapkan pada berbagai jenis dokumen *XML*, seperti *SVG* dan *XUL*. Standar *CSS* dikembangkan dan dikelola oleh *World Wide Web Consortium* (W3C).

2.2.14 *UML (*Unifield Modeling Language*)*

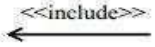
UML, atau Unified Modeling Language, adalah bahasa yang digunakan untuk menggambarkan desain perangkat lunak. Dengan *uml* kita bisa memvisualisasikan, menjelaskan, dan mendokumentasikan berbagai bagian dari sistem perangkat lunak. Seperti halnya arsitek yang membuat rencana untuk sebuah bangunan, arsitek perangkat lunak menggunakan diagram *uml* untuk membantu pengembang dalam membuat perangkat lunak. Ada berbagai jenis diagram uml yang umum digunakan dalam pengembangan sistem, antara lain: (1) *Use Case*, menggambarkan fungsi sistem dan interaksi antara pengguna atau sistem lain dengan sistem. (2) *Activity Diagram*, menunjukkan alur aktivitas atau proses dalam sistem. (3) *Sequence Diagram*, memperlihatkan urutan interaksi antar objek dalam sistem berdasarkan urutan waktu (4) *Class Diagram*, menggambarkan struktur statis sistem

dengan menunjukkan kelas-kelas beserta atribut, metode, dan hubungan antar kelas seperti asosiasi, generalisasi, serta dependensi[17].

Terdapat beberapa diagram *UML* yang sering digunakan dalam pengembangan sebuah sistem yaitu:

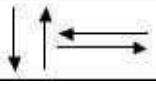
1. *Use Case Diagram*: merupakan gambaran dari fungsionalitas yang diharapkan dari sebuah sistem dan merepresentasikan sebuah interaksi antara aktor dan sistem. Didalam *use case* terdapat aktor yang merupakan sebuah gambaran entitas dari manuasia atau sebuah sistem yang melakukan pekerjaan di sistem.

Tabel 2. 1 Use Case Diagram

Simbol	**Keterangan**
	Aktor : Mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan *use case*
	Use case : Abstraksi dan interaksi antara sistem dan aktor
	Association : Abstraksi dari penghubung antara aktor dengan use case
	Generalisasi : Menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan use case
	Menunjukkan bahwa suatu use case seluruhnya merupakan fungsionalitas dari use case lainnya
	Menunjukkan bahwa suatu use case merupakan tambahan fungsional dari use case lainnya jika suatu kondisi terpenuhi

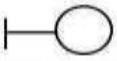
2. *Activity Diagram*: Sebuah diagram aktivitas UML menggambarkan perilaku dinamis dari suatu sistem atau bagian dari sistem melalui aliran kontrol antara aksi yang dilakukan sistem.

Tabel 2. 2 Activity Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		*Activity*	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		*Action*	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		*Initial Node*	Bagaimana objek dibentuk atau diawali.
4		*Activity Final Node*	Bagaimana objek dibentuk dan diakhiri
5		*Decision*	Diguanakan untuk menggambarkan suatu keputusan / tindakan yang harus diambil pada kondisi tertentu
6		*Line Connector*	Digunakan untuk menghubungkan satu simbol dengan simbol lainnya

3. *Sequence Diagram*: menggambarkan interaksi antar objek didalam dan di sekitar sistem (termasukpengguna, *display* dan sebagainya) berupa message yang digambarkan terhadap waktu. *Sequence diagram* biasa digunakan untuk menggambarkan skenario atau langkah- langkah yang dilakukansebagai respons dari sebuah event untuk menghasilkan output tertentu.

Tabel 2. 3 Sequence Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		*Actor*	Menggambar orang yang sedang berinteraksi dengan sisitem.
2		*Entity Class*	Menggambarkan hubungan yang akan dilakukan
3		*Boundary Class*	Menggambarkan sebuah gambbaran dari foem
4		*Control Class*	Menggambarkan penghubung antara boundary dengan tabel
5		*A focus of Control & A Life Line*	Menggambarkan tempat mulai dan berakhirnya massage
6		*A massage*	Menggambarkan Pengiriman Pesan

4. *Class Diagram*: merupakan gambaran struktur dan deskripsi dari *class, package* dan objek yang saling berhubungan seperti diantaranya pewarisan, asosiasidan lainnya.

Tabel 2. 4 Class Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		*Generalization*	Hubungan dimana objek anak *(descendent)* berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (*ancestor*).
2		*Nary Association*	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		*Class*	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		*Collaboration*	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
5		*Realization*	Operasi yang benar-benar dilakukan oleh suatu objek.
6		*Dependency*	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri *(independent)* akan mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri
7		*Association*	Apa yang menghubungkan antara objek satu dengan objek lainnya

2.2.15 *Black box testing*

Black box testing, atau yang sering disebut sebagai *Behavioral Testing*, adalah jenis pengujian yang menilai perangkat lunak berdasarkan hasil dari input dan output, tanpa memperhatikan bagaimana kode di dalamnya bekerja. Metode ini biasanya diterapkan pada tahap akhir pengembangan untuk memastikan bahwa perangkat lunak beroperasi dengan baik dan memenuhi kebutuhan yang telah ditentukan.