

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Penelitian yang dilakukan oleh Faisal Arief Deswar, dkk pada tahun (2021). Penelitian berfokus pada pembuatan sistem monitoring suhu dan kelembaban ruang server secara *real-time*. Penelitian ini memiliki kelebihan berupa pemanfaatan teknologi yang memungkinkan kontrol dan monitoring suhu ruang server secara jarak jauh melalui aplikasi Android, sistem ini juga memiliki beberapa kekurangan, di antaranya jangkauan IR Transmitter yang terbatas hingga 420 cm, akurasi sensor DHT11 yang kurang presisi dibandingkan sensor lain, ketergantungan penuh pada koneksi internet dan server Heroku, belum adanya fitur notifikasi atau alarm saat suhu di luar batas aman, [4].

Dari penelitian yang dilakukan Rafika Andari, dkk pada tahun (2021). Penelitian yang dilakukan yaitu merancang sebuah sistem kendali suhu yang dapat mengontrol suhu yang dihubungkan dengan mikrokontroler dengan kontrol logika *fuzzy* sebagai pengendalinya, karena logika *fuzzy* adalah suatu cara yang tepat untuk mengatakan nilai kontinyu, penelitian ini merancang protipe pengendali suhu ruangan menggunakan metode penalaran secara manual. Kekurangan dalam penelitian tidak menggunakan pengendalian ruangan dengan *website* atau mobile app[5].

Penelitian yang dilakukan oleh Fajar Ciputra Daeng, dkk pada tahun (2024). Penelitian yang dilakukan yaitu sistem monitoring kelembaban suhu pada penyimpanan Gudang loofah dengan penggunaan humidifier, hasil penelitian menunjukkan bahwa sistem monitoring yang dikembangkan mampu menjaga suhu dan kelembaban di dalam gudang loofah pada level yang diinginkan. Implementasi sistem ini dapat membantu mengoptimalkan kondisi penyimpanan loofah, mengurangi kerusakan produk, dan meningkatkan efisiensi penyimpanan. Tetapi kekurangan pada penelitian ini hanya mengandalkan alat mikrokontroler belum menggunakan *website* untuk sistem monitoringnya[6].

Penelitian yang dilakukan oleh Risam Santosa, dkk pada tahun (2023). Penelitian yang dilakukan yaitu sistem monitoring suhu dan kelembaban pada gudang penyimpanan PT Sakadarma Laboratories yang menggunakan metode prototype. Sistem monitoring suhu dan kelembaban berhasil memenuhi tujuan penelitian dengan efektif, implementasi sistem ini memberikan kemudahan bagi pengguna dengan mengeliminasi beberapa proses pekerjaan sebelumnya. Kelebihan penelitian ini pengguna tidak perlu lagi melakukan monitoring secara manual, karena sistem secara otomatis memantau dan mencatat nilai suhu dan kelembaban[7].

Penelitian yang dilakukan oleh Rosyid Mufti Abdurrohman, dkk pada tahun (2023). Penelitian ini membahas tentang prototipe monitoring suhu dan kelembaban secara real-time, sistem ini di rancang untuk memberikan pemantau akurat dan *real-time* terhadap suhu dan kelembapan lingkungan sekitar. Penelitian ini memiliki kelebihan karena mampu memberikan pengukuran suhu dan kelembapan yang cukup akurat dengan respons yang cepat secara real-time. Dalam merancang sebuah sistem yang dapat memantau keadaan suhu dan kelembaban pada ruang server dengan memanfaatkan jaringan sensor nirkabel sehingga memungkinkan pengiriman data atau pesan dilakukan secara nirkabel. Antarmuka yang digunakan pada penelitian ini yaitu mobile view yang memungkinkan administrator memantau keadaan suhu dan kelembaban ruang server di android sehingga administrator jaringan tidak perlu lagi masuk ke ruang server. penelitian ini memiliki beberapa kekurangan, di antaranya rentang pengukuran suhu dan kelembapan yang terbatas, belum adanya integrasi dengan platform website untuk pemantauan jarak jauh,[8].

Berdasarkan beberapa penelitian diatas mengenai pengembangan suhu dan kelembaban masih belum otomatis sering kali mengandalkan pengaturan manual dan masih sedikit yang membahas tentang penelitian humidifier untuk kelembaban ruangan. Maka penelitian ini mengembangkan sistem monitoring humidifier otomatis berbasis website yang mampu mengisi kekurangan pada sistem yang ada dengan

memberikan kontrol otomatis terhadap kelembaban ruangan yang dapat disesuaikan dengan lingkungan. Hal ini meningkatkan efisiensi dan kenyamanan dalam pengendalian kelembaban ruangan.

2.2 Landasan Teori

2.2.1 Sistem Monitoring

Sistem monitoring atau sistem pengawasan adalah suatu upaya yang sistematis untuk menetapkan kinerja standar pada perencanaan untuk merancang sistem umpan balik informasi, untuk membandingkan kinerja aktual dengan standar yang telah ditentukan, untuk menetapkan apakah telah terjadi suatu penyimpangan tersebut, serta untuk mengambil tindakan perbaikan yang diperlukan untuk menjamin bahwa semua sumber daya perusahaan atau organisasi telah digunakan secara efektif dan efisien mungkin guna mencapai tujuan. Monitoring menyediakan data dasar untuk menjawab permasalahan serta akan memberikan informasi tentang status, proses monitoring umumnya dilakukan untuk tujuan memeriksa terhadap proses suatu objek atau mengevaluasi kondisi[9].

2.2.2 PHP

PHP merupakan salah satu bahasa pemrograman berbasis web yang ditulis oleh dan untuk pengembang web. PHP pertama kali dikembangkan oleh Rasmus Lerdorf, seorang pengembang *software* dan anggota tim Apache, dan dirilis pada akhir tahun 1994. PHP dikembangkan dengan tujuan awal hanya untuk mencatat pengunjung pada *website* pribadi Rasmus Lerdorf. PHP merupakan bahasa pemrograman berbasis web yang dibuat secara khusus untuk membangun aplikasi berbasis web. Selain tersedia secara gratis, PHP juga mudah dipelajari oleh siapapun [10].

2.2.3 MySQL

MySQL merupakan sebuah program database server yang mampu menerima dan mengirimkan datanya dengan sangat cepat, multiuser serta menggunakan perintah standar SQL (*Structured Query Language*) [11].

2.2.4 Visual Studio Code

Visual Studio Code merupakan sebuah aplikasi editor code open source yang dikembangkan oleh Microsoft untuk sistem operasi Windows, Linux, dan MacOS. Visual Code memudahkan dalam penulisan code yang mendukung beberapa jenis pemrograman, seperti C++, C#, Java, Python, PHP. Visual Code memiliki kemampuan untuk mengidentifikasi jenis bahasa pemrograman yang digunakan dan memberi variasi warna sesuai

dengan fungsi dalam rangkaian code tersebut. Visual Studio Code juga telah terintegrasi ke Github. Selain itu fitur lainnya adalah kemampuan untuk menambah ekstensi dimana para pengembang dapat menambah ekstensi untuk menambah fitur yang tidak ada di Visual Studio Code [12].

2.2.5 Database

Database merupakan suatu koleksi terstruktur dari data yang saling terkait, disimpan dalam media penyimpanan komputer, dan dapat diakses serta dikelola menggunakan perangkat lunak khusus. Database digunakan untuk menyimpan, mengelola, dan mengorganisir data dengan tujuan memberikan akses yang efisien, aman, dan terstruktur terhadap informasi [13].

2.2.6 Codeigniter

CodeIgniter adalah Sebuah framework php yang bersifat open source dan menggunakan metode MVC (Model,View,Controller) untuk memudahkan developer atau programmer dalam membangun sebuah aplikasi berbasis web tanpa harus membuatnya dari awal [14].

2.2.7 Java Script

Sekumpulan script yang fungsinya dijalankan pada dokumen *HTML*. *JavaScript* adalah bahasa skrip yang awalnya dikembangkan khusus untuk penggunaan di *website*. Bahasa ini digunakan sebagai bahasa pemrograman untuk memberikan fungsi tambahan pada halaman *website* menggunakan eksekusi perintah di sisi pengguna, yaitu di dalam browser, bukan di server *website*.

2.2.8 UML (*Unified modeling language*)

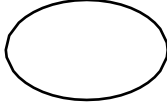
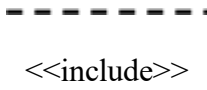
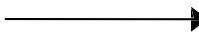
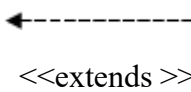
Unified Modeling Language merupakan salah satu metode pemodelan visual yang digunakan dalam perancangan dan pembuatan sebuah *software* yang berorientasikan pada objek. UML merupakan sebuah standar penulisan atau semacam blue print dimana didalamnya termasuk sebuah bisnis proses, penulisan kelas-kelas dalam sebuah bahasa yang spesifik.

Dalam perancangan sistem terdapat UML yang sering digunakan sebagai berikut:

1. *Use Case Diagram*

Dalam pembuatan sistem informasi, *use case diagram* merupakan pemodelan untuk *behavior* atau kelakuan sistem informasi. Diagram ini juga bersifat statis. Untuk simbol *Use Case Diagram* disajikan pada Tabel 2.1.

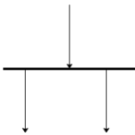
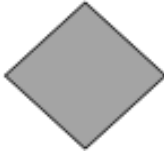
Tabel 2. 1 Use Case Diagram

No	Simbol	Keterangan
1.		<i>Use case</i> merupakan deskripsi fungsional yang telah disediakan oleh sistem sebagai entitas yang menghasilkan hasil yang terukur untuk suatu <i>actor</i> .
2.		<i>Actor</i> merupakan himpunan peran untuk berinteraksi dengan <i>usecase</i>
3.		<i>Association</i> merupakan garis yang menghubungkan objek satu dengan objek yang lain
4.		<i>Include</i> merupakan gambaran jika <i>usecase</i> dipanggil oleh <i>usecase</i> lain
5.		<i>Dependecy</i> merupakan garis panah yang menunjukan jika <i>actor</i> berinteraksi secara pasif
6.		<i>Extend</i> merupakan gambar jika memperluas <i>usecase</i> target.

2. Activity Diagram

Activity Diagram atau Diagram Aktivitas merupakan diagram yang bersifat statis, yang menggambarkan aktivitas dari suatu sistem bisnis. Untuk simbol dari diagram aktivitas disajikan pada Tabel 2.2.

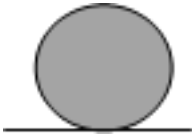
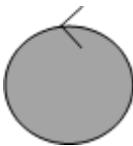
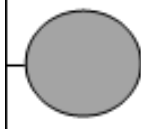
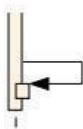
Tabel 2. 2 Activity Diagram

No	Simbol	Keterangan
1.		<i>End point</i> atau <i>final Node</i> merupakan gambaran akhir dari suatu aktivitas
2.		<i>Start Point</i> merupakan awal dari suatu aktivitas yang peletakannya pada pojok kiri atas
3.		<i>Fork</i> atau join digunakan untuk memarallelkan suatu kegiatan atau penggabungan 2 kegiatan parallel menjadi satu
4.		<i>Activity</i> merupakan gambaran dari suatu proses
5.		<i>Decision</i> merupakan pilihan pengambilan suatu keputusan <i>false or true</i>

3. *Sequence Diagram*

Sequence Diagram atau Diagram Urutan mendeskripsikan diagram interaksi yang mengirimkan pesan dan diterima antar objek. Untuk simbol diagram urutan disajikan pada Tabel 2.3.

Tabel 2. 3 Squence Diagram

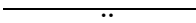
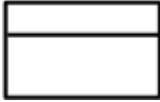
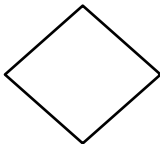
No	Simbol	Keterangan
1.		<i>Entity Class</i> merupakan bagian sistem yang membentuk deskripsi awal sistem dan berisi kumpulan kelas dalam bentuk entitas yang mendasari untuk membuat database
2.		<i>ControlClass</i> merupakan gambaran penghubung antara <i>Boundaryclass</i> dengan suatu table
3.		<i>Boundary Class</i> merupakan gambaran dari penggambaran table
4.		Pesen atau <i>message</i> menunjukkan pengiriman pesen antar <i>class</i>
5.		<i>Self message</i> menunjukkan pengiriman suatu pesan yang akan dikirim ke objek itu sendiri.

6.		<i>Activation</i> menggambarkan suatu objek yang melakukan sebuah aksi/eksekusi operasi
7.		<i>Lifeline</i> garis titik yang terhubung ke objek disepanjang garis lifeline memiliki aktivitas

4. Class Diagram

Class Diagram merupakan diagram yang bersifat statis, dalam diagram ini memperlihatkan himpunan kelas, antarmuka, serta relasi. Simbol dari Class Diagram disajikan pada Tabel 2.4.

Tabel 2. 4 Class Diagram

No	Simbol	Keterangan
1.		<i>Generallization</i> , merupakan dimana objek <i>descendent</i> membagikan perilaku dan struktur data objek Induknya
2.		<i>Class</i> , adalah kumpulan objek yang saling berbagi.
3.		<i>Nary Assocation</i> , digunakan untuk asosiasi terhindar dengan objek lainnya.

4.		<i>Collaboration</i> merupakan deskripsi urutan aksi yang ditampilkan suatu sistem yang memiliki konsekuensi terukur bagi <i>actor</i>
5.		Merupakan operasi yang valid dilakukan oleh suatu objek
6.		Merupakan garis penah yang menunjukkan jika actor berinteraksi secara pasif
7.		<i>Association</i> merupakan garis yang menghubungkan objek satu dengan objek yang lain