

BAB II

TINJAUAN PUSTAKA

2.1 Teori Terkait

Pada penelitian yang dilakukan oleh Very Kurnia Bakti, et.all mengenai sistem pemantauan ruang data center dengan menggunakan *Application Programming Interface* (API) Tuya. Mereka membangun sistem *IoT* untuk memantau ruang data center dengan memanfaatkan berbagai sensor secara bersamaan seperti *DHT11*, *LM35*, *MQ2*, *MQ*, dan bagian-bagian lainnya dalam kendali jarak jauh melalui komputer atau gawai dengan monitoring berupa *Application Programming Interface* (API) Tuya[1].

Taufik Rahman dengan timnya melakukan penelitian terkait tentang Sistem Monitoring Suhu, Asap dan Api Ruang Server *Information And Communication Technology (ICT)* DI Universitas Bina Insan. Riset ini bertujuan untuk membuat alat pemantauan kondisi server dengan sensor suhu, sensor asap dan sensor api *DFR0011*. Hasil pengukuran dari sensor-sensor ini akan disimpan otomatis dalam *database* dan dapat diakses melalui *website*. Dengan demikian, alat ini akan membantu administrator dalam melakukan pemantauan secara *realtime* selama 24 jam[3].

Muhamad Supriyanto dan Fajar Agung Nugroho sebagai rekannya telah melakukan penelitian mengenai sistem pendeteksi asap dan pemantauan suhu dan kelembaban di ruang server berbasis *iot* dengan *Metode Fuzzy Logic*. Tujuan riset ini adalah untuk membuat sebuah perangkat detektor asap dan pemantau suhu serta kelembaban menggunakan sensor *DHT22* dan

sensor *MQ-2* yang terhubung ke *Internet of Things (IoT)*. Perangkat ini akan mampu memantau suhu, kelembaban, dan mendeteksi keberadaan asap. Data dan informasi pengukuran suhu dan kelembaban akan disimpan ke dalam *database*. Melalui sebuah *website* monitoring, administrator atau pengguna akan dapat mengakses informasi hasil pemantauan dari perangkat secara jarak jauh[4].

Menurut penelitian Doni Halim dan rekannya mengenai sebuah sistem peringatan dini dengan menggunakan beragam sensor pada ruang server yang dioperasikan dengan basis *Arduino*. Riset ini bertujuan untuk membuat suatu sistem yang dapat peringatan secara cepat saat terjadi masalah. Sistem ini akan dirancang sebagai prototipe menggunakan *Arduino Uno* dipadukan dengan sensor *DHT22*, *MQ-2*, detector Api, *beep*, *relay*, serta unit *Ethernet Shield* untuk mengirimkan notifikasi melalui *surel*[5].

2.2 Landasan Teori

2.2.1 Sistem Monitoring

Dikenal sebagai langkah terstruktur untuk memeriksa apakah kinerja sistem sesuai dengan standar yang ditetapkan, dengan cara memantau secara teratur, membandingkan dengan standar, mengidentifikasi perbedaan, dan melakukan perbaikan jika diperlukan. Hal ini membantu organisasi atau perusahaan untuk memastikan penggunaan sumber daya yang efisien dan agar dapat mencapai tujuan yang telah ditetapkan[6].

2.2.2 Xampp

Dikenal sebagai aplikasi server web yang menggabungkan *Apache*, *MySQL*, dan *phpMyAdmin* dalam satu paket yang terpadu. Huruf "X" menunjukkan bahwa *XAMPP* dapat diinstal dan berjalan pada berbagai sistem operasi seperti *Windows*, *Linux*, *MacOS*, dan *Solaris*. Ini memungkinkan pengguna untuk mengelola server web tanpa memperhatikan sistem operasi yang digunakan[7].

Dan juga merupakan *software* gratis yang mendukung berbagai sistem operasi serta kumpulan program-program yang memungkinkan pengguna membuat lingkungan pengembangan web lokal tanpa harus menginstal setiap komponen secara terpisah pada komputer mereka[8].

2.2.3 Mysql

Dikenal sebagai merupakan server basisdata *open source* dan umum digunakan dalam berbagai aplikasi, terutama untuk pembuatan situs web atau server. *MySQL* berperan sebagai sistem manajemen basis data yang menggunakan bahasa *query SQL* sebagai bahasa utamanya. Umumnya, *MySQL* digunakan bersama dengan *PHP* untuk mengembangkan aplikasi server yang dinamis dan andal[9].

Dan juga merupakan basis data terkenal karena kecepatannya dalam pengiriman dan penerimaan data. Selain itu,

MySQL mendukung penggunaan multiuser dan dapat diakses oleh beberapa pengguna secara bersamaan[10].

2.2.4 Php

Merupakan kependekan dari *Hypertext Preprocessor*. Ini adalah scripting language yang dijalankan di server dan hasilnya dikirimkan ke klien, seperti browser web, untuk digunakan oleh pengguna sehingga dapat melihat dan berinteraksi dengan konten yang dihasilkan[11].

Dikenal sebagai bahasa pemrograman server-side karena eksekusi kode *PHP* terjadi di sisi server web, bukan pada perangkat pengguna akhir. Dengan kata lain, saat pengguna mengakses situs web yang menggunakan *PHP*, permintaan mereka akan diproses oleh server, yang menjalankan kode *PHP* untuk menghasilkan halaman web yang kemudian dikirim kembali ke perangkat pengguna melalui browser mereka[12].

2.2.5 Database

Merupakan sekumpulan informasi yang tersusun secara terstruktur dan disimpan di dalam komputer. Informasi ini dapat diakses dan diperiksa menggunakan program komputer untuk mendapatkan data yang diinginkan[13].

Dan juga disusun sedemikian rupa sehingga dapat dengan cepat dan mudah diakses kembali dan dimanfaatkan[14].

2.2.6 Visual Studio Code

Dikenal sebagai adalah editor teks yang dibuat oleh *Microsoft*, memiliki bobot yang ringan dan diandalkan, serta dapat digunakan pada berbagai sistem operasi, termasuk *Linux*, *Mac*, dan *Windows*. Editor ini secara default mendukung bahasa pemrograman seperti *JavaScript*, *TypeScript*, dan *Node.js*. Tambahan lagi bahasa pemrograman lainnya dapat diakses melalui *plugin* yang tersedia di tool aplikasi *Visual Studio Code*, termasuk *C++*, *C#*, *Python*, *Go*, *Java*, *PHP*, dan masih banyak lagi[15].

2.2.7 UML (Unified Modelling Language)

Merupakan bahasa yang telah distandarisasi untuk membuat representasi visual dari sistem perangkat lunak. Dengan *UML*, pengembang dapat memvisualisasikan aspek-aspek penting dari sistem dan memudahkan pemahaman tentang bagaimana sistem beroperasi.

Berikut beberapa model *UML* yaitu seperti di bawah ini :

1. *Use Case* Diagram

UseCase Diagram menggambarkan bagaimana actor-aktor eksternal berinteraksi dengan sistem untuk mencapai tujuan tertentu dan juga memiliki sifat yang statis. Simbol-simbol *UseCase* Diagram dapat di lihat dalam Tabel 2.1.

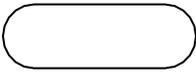
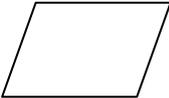
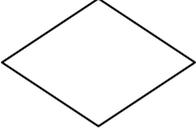
Tabel 2. 1 Simbol *UseCase* Diagram

No	Simbol	Deskripsi
1		<i>Actor</i> Adalah sekumpulan peran yang dimainkan oleh pengguna ketika berinteraksi dengan suatu kasus penggunaan.
3		<i>Include</i> Adalah gambaran ketika usecase dipanggil usecase lain.
4		<i>Extend</i> Adalah gambaran untuk memperluas usecase target.
5		<i>Association</i> Adalah garis yang menghubungkan antara satu objek dengan objek lain.
6		<i>UseCase</i> Merupakan urutan tindakan yang dilakukan oleh suatu sistem yang menghasilkan suatu hasil yang dapat diukur oleh actor.

2.2.8 Flowchart

Dikenal sebagai gambaran visual dari tindakan serta keputusan yang diambil dalam suatu proses menggunakan simbol-simbol grafis. Ini membantu untuk memahami proses secara sistematis, baik dalam pemrograman maupun rekayasa sistem. Simbol-simbol flowchart dapat dilihat pada Tabel 2.2.

Tabel 2. 2. Simbol *Flowchart*.

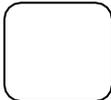
Simbol	Keterangan
	<i>Terminator / Terminal</i> adalah ikon atau simbol yang menandakan status awal dan akhir dari diagram alur.
Simbol	Keterangan
	<i>Preparation / Penyusunan</i> Adalah simbol yang menunjukkan pengenalan variabel yang digunakan.
	<i>Input Output</i> adalah simbol untuk mengatur nilai variabel dan menampilkan nilai dari variabel tersebut.
	<i>Process / Proses</i> adalah simbol yang menandakan pelaksanaan operasi tertentu seperti penggunaan rumus, perhitungan, atau pemberian nilai pada suatu variabel.
	<i>Decision / Simbol Keputusan</i> adalah simbol yang digunakan untuk membuat pilihan berdasarkan kondisi Ya/Tidak. Ini memberikan opsi untuk menentukan apakah suatu kondisi benar (true) atau salah (false).
	<i>Arrow / Arus</i> simbol ini menggambarkan aliran dalam diagram alur program. Oleh karena itu, dalam menggambarkan aliran data perlu menggunakan simbol panah.

2.2.9 Activity Diagram

Merupakan penggambaran visual dari urutan langkah-langkah dalam suatu proses, menggunakan simbol-simbol seperti

kotak dan panah untuk menggambarkan aliran kerja. Simbol-simbol dari *Activity Diagram* bisa dilihat dalam Tabel 2.3

Tabel 2. 3. *Activity Diagram*

No	Simbol	Deskripsi
1		<i>Activity</i> Adalah sebuah simbol yang mewakili tindakan atau langkah konkret dalam suatu proses.
No	Simbol	Deskripsi
2		<i>Action</i> Adalah sebuah state yang mencerminkan eksekusi dari suatu aksi tertentu dalam proses.
3		<i>Initial Node</i> Adalah titik awal dari aliran kerja atau proses.
4		<i>Final Node</i> Adalah simbol yang menunjukkan titik dimana aliran kerja atau proses berakhir.
5		<i>Decision</i> Adalah simbol yang menunjukkan titik dalam aliran kerja dimana suatu keputusan harus dibuat.