

BAB II

TINJAUAN PUSTAKA

2.1 Kajian Teori

2.1.1 Internet of Things (IoT)

Internet of Things (IoT) adalah teknologi yang memungkinkan berbagai perangkat atau benda saling terhubung melalui jaringan untuk bertukar data dan bekerja secara otomatis tanpa harus selalu diawasi manusia. Teknologi ini memanfaatkan sensor, perangkat lunak, koneksi internet untuk mengontrol, menghubungkan, serta mengirimkan informasi antar perangkat [5].

Konsep *Internet of Things (IoT)* diterapkan dengan menghubungkan modul ESP32-CAM ke jaringan internet, perangkat juga untuk berkomunikasi data secara otomatis. Selama sistem berjalan, pengguna hanya perlu memantau dan mengontrol perangkat melalui website, tanpa harus berada di perangkat.

Internet of Things (IoT) memiliki keunggulan dalam mendukung pengembangan sistem berbasis teknologi modern. Kelebihannya adalah memudahkan integrasi dengan berbagai perangkat seperti sensor, Mikrokontroler, web, dan sistem basis data.



Gambar 2.1 Internet Of Things (IoT)

2.1.2 Website

Website adalah sekumpulan halaman web yang saling terhubung dan dapat diakses melalui jaringan internet menggunakan browser. Website berfungsi untuk media penyampaian informasi sebagai pengguna untuk mengelola, memperoleh, dan menampilkan data. Dalam sistem monitoring menggunakan modul ESP32-CAM, website berfungsi sebagai media penyampaian informasi oleh pengguna untuk mengelola, memperoleh dan menampilkan data gambar yang dikirimkan oleh perangkat melalui API. Website banyak digunakan sebagai sistem informasi berbasis jaringan yang fleksibel, mudah diakses, dan mampu proses pengelolaan data secara lebih efektif [6].

Pada penelitian ini, website berperan sebagai media utama bagi pengguna untuk melihat hasil monitoring secara *real-time*. ESP32-CAM akan mengambil gambar menggunakan kamera yang terpasang, kemudian gambar tersebut dikirimkan melalui koneksi internet menggunakan API menuju server website. Selanjutnya, server akan memproses dan menyimpan data sehingga dapat ditampilkan pada halaman website yang telah di rancang. Selain media penyimpanan data, website juga berfungsi sebagai antarmuka pengguna (*user interface*) untuk memudahkan proses monitoring secara *real-time*.

2.1.3 Website Monitoring

Website monitoring adalah suatu sistem berbasis web yang dirancang untuk mengelola, menampilkan, dan memantau data yang diperoleh dari perangkat melalui jaringan internet. Website, data yang dapat diperoleh dari perangkat monitoring yang dikirim dan disimpan ke pengguna dalam bentuk

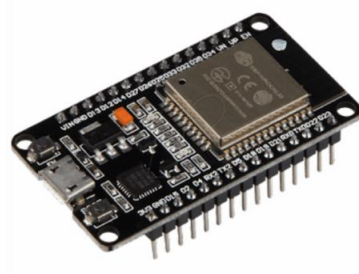
informasi. Penggunaan website dapat memonitoring memberikan kemudahan dalam mengakses data untuk melakukan melalui jaringan internet.

Pada penelitian ini, website monitoring dimanfaatkan untuk mengirimkan data gambar yang dikirim oleh modul ESP32-CAM melalui API. Gambar berhasil diterima dan disimpan pada server dan ditampilkan pada halaman website ke penggunaan yang melakukan pemantaun jarak jauh.

2.1.3 ESP32

ESP32 adalah *mikrokontrol* yang dikembangkan oleh espressif sistem dan banyak digunakan dalam pengembangan sistem *Internet of Things* (IoT). ESP32 memiliki kemampuan komunikasi nirkabel berupa Wi-Fi dan *Bluetooth* yang sudah tertanam langsung di dalam chip, sehingga sangat mendukung proses pengiriman dan pertukaran data melalui jaringan internet. Selain itu, ESP32 juga dilengkapi prosesor dual-core 32-bit yang mampu memproses data dengan cepat dan stabil memiliki fitur hemat daya seperti *dynamic power scaling* dan *multiple power modes* sehingga konsumsi dayanya lebih efisien [7].

Modul ini digunakan untuk mengambil gambar menggunakan kamera yang terintegrasi pada perangkat, sehingga memproses dan pengiriman data gambar tersebut ke website melalui jaringan internet menggunakan API. Dengan adanya koneksi Wi-Fi pada ESP32-CAM, proses pengiriman data dapat dilakukan secara *real-time* tanpa membutuhkan perangkat tambahan lainnya.



Gambar 2.2 ESP32

2.1.5 ESP32-CAM

ESP32-CAM merupakan salah satu modul *mikrokontroler* yang telah mengintegrasikan pada kemampuan proses data dan pengambilan dalam satu perangkat. Modul ini dilengkapi dengan fitur Wi-Fi dan *Bluetooth* sehingga dapat digunakan untuk sistem *monitoring* berbasis *Internet of Things* (IoT). ESP32-CAM berfungsi sebagai untuk mengambil gambar, memproses data, serta mengirimkan hasil gambar ke website melalui jaringan internet.

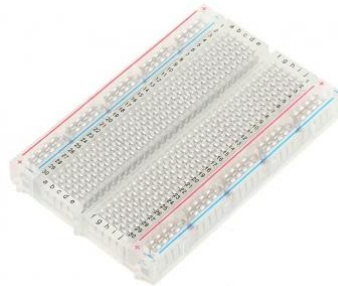
ESP32-CAM memiliki modul kamera ukuran kecil dan dilengkapi kamera OV2640 yang dapat digunakan untuk menangkap gambar maupun video streaming secara *real-time*. Karena kemampuannya tersebut, ESP32-CAM juga diterapkan pada sistem *smart home*, *wireless monitoring*, keamanan rumah, dan pengenalan wajah [8]. Lensa kamera Pada modul OV2640 yang di rancang untuk menangkap Cahaya dari objek yang berada di depan kamera ke sensor gambar. Lensa dapat berpengaruh terhadap ketajaman pada pencahayaan gambar yang secara detail.



Gambar 2.3 ESP32-CAM

2.1.6 *Beardboard*

Beardboard merupakan papan rangkai elektronik yang digunakan untuk merangkai dan menghubungkan komponen tanpa harus melakukan penyolderan. Penggunaan *beardboard* pada sistem ini bertujuan untuk mempermudah proses perancang, pengujian, dan *troubleshooting* rangkai sebelum alat dibuat secara permanen. Dengan *beardboard* ini setiap pin pada ESP32-CAM dapat dihubungkan menggunakan kabel jumper sehingga proses pengembangan istem monitoring menjadi lebih praktis dan fleksibel.



Gambar 2.4 Beardboard

2.1.7 *Buzzer*

Buzzer adalah komponen elektronik output yang mengambilkan suara sebagai indikator atau alarm. Dalam sistem monitoring berbasis ESP32-CAM, *buzzer* digunakan untuk memberikan peringatan ketika sensor mendeteksi kondisi tertentu, seperti adanya gerakan atau objek mencurigakan. Setelah sensor aktif, ESP32-CAM dapat mengambil gambar dan mengirimkan data gambar dan mengirimkan data gambar ke website melalui API. Pengguna *buzzer* membantu meningkatkan keamanan karena pengguna dapat mengetahui adanya aktivitas secara langsung melalui suara alarm maupun monitoring berbasis web [9].



Gambar 2.5 Buzzer

2.1.8 Sensor PIR

Sensor PIR merupakan sensor yang digunakan untuk mendeteksi gerakan berdasarkan perubahan radiasi inframerah dihasilkan oleh objek. Sensor ini bekerja secara pasif dengan menerima perubahan radiasi inframerah tanpa memancarkan energi sendiri. Pada sistem monitoring berbasis ESP32, sensor PIR berperan sebagai pendeteksi awal ketika terdapat pergerakan manusia atau objek. Sinyal yang dihasilkan sensor kemudian diteruskan ke ESP32 untuk diproses dan digunakan sebagai pemicu tindakan selanjutnya, seperti mengaktifkan kamera atau mengirimkan informasi hasil deteksi ke sistem monitoring [11].



Gambar 2.6 Sensor PIR

2.1.10 Kabel Data Micro

Pada sistem yang menggunakan modul ESP32, kabel data micro ini digunakan sebagai media komunikasi antar komputer dan modul ESP32 melalui

antarmuka USB. Kabel memungkinkan proses transfer data, debugging, monitoring serial, sehingga konektivitas jaringan yang dibutuhkan dalam implementasi sistem *Internet of Things* (IoT). Kabel data mikro ini juga berfungsi sebagai sumber catu daya sehingga perangkat dapat beroperasi secara stabil saat proses pengujian langsung [12].



Gambar 2.7 Kabel Data Micro

2.1.11 Kabel Jumper

Kabel jumper adalah kabel yang menghubungkan satu komponen elektronik dengan komponen lainnya pada rangkaian prototipe tanpa memerlukan penyolderan. Kabel ini berfungsi sebagai media penghubung jalur data, maupun distribusi tegangan antara mikrokontroler, sensor, aktuator, dan perangkat pendukung lainnya sehingga sistem dapat bekerja sesuai rancangan. Pada sistem berbasis *Internet of Things* (IoT), kabel jumper dapat memudahkan proses perakitan, pengujian, dan pengembangan rangkaian koneksi antar perangkat yang dapat diubah atau disesuaikan dengan mudah. Penggunaan kabel jumper banyak yang memanfaatkan modul ESP32-CAM sebagai pusat pengiriman data melalui jaringan internet [13].

Kabel jumper dapat memungkinkan proses transmisi data dan penyaluran tegangan yang baik sehingga komponen dapat saling berkomunikasi dalam sistem

monitoring. Penggunaan kabel jumper dapat mempermudah proses perancangan perangkat dan pengujian yang tidak memerlukan penyolderan secara permanen, sehingga lebih efektif dalam pengembangan sistem pengiriman data gambar ke website melalui API.



Gambar 2.8 Kabel Jumper

2.1.12 Database

Database adalah kumpulan data yang disimpan dan saling terhubung sehingga dapat dikelola, diakses dengan mudah sesuai kebutuhan sistem [14]. Database juga sebagai media penyimpanan data yang dikirim melalui API, seperti tangkapan gambar, informasi waktu pengambilan data, dan data monitoring lainnya. Dengan adanya database, data yang di peroleh dari perangkat yang tersimpan dengan baik dan dapat digunakan untuk pemantauan pengolahan informasi.

Adapun fungsi database dalam sistem monitoring ESP32-CAM sebagai berikut:

1. Menyimpan data gambar dan informasi monitoring ESP32-CAM
2. Mengirim data dari ESP32-CAM ke database

3. Menampilkan data yang tersimpan dalam database
4. Mengambil data dan mengirimkan data ke server
5. Mengelola proses penyimpanan dan pengambilan data dari database

2.1.13 PostgreSQL

PostgreSQL adalah sistem manajemen basis data *Relational Database Management System* (RDBMS), *open source* yang digunakan untuk menyimpan, mengelola, dan memproses data secara terstruktur. Ini mendukung penggunaan bahasa *Structured Query Language* (SQL) untuk melakukan operasi pengolahan data seperti menambah, mengubah, menghapus, dan menampilkan data yang tersimpan dalam database.

Dengan menggunakan API nya, PostgreSQL dapat digunakan sebagai media penyimpanan data. Data yang dikirim oleh ESP32-CAM, termasuk informasi gambar, waktu pengambilan gambar, dan data pendukung lainnya, disimpan dalam database PostgreSQL. Ini membuat lebih mudah untuk diakses, ditampilkan, dan dikelola melalui website dengan lebih efisien.

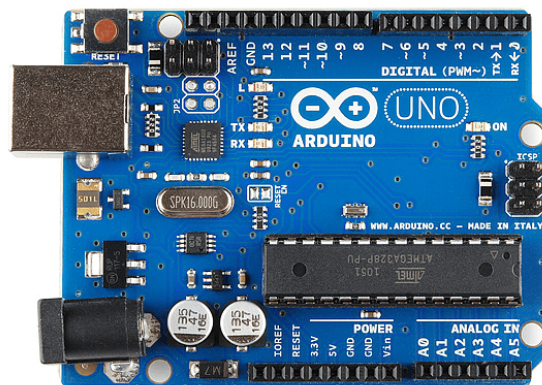


Gambar 2.9 PostgreSQL

2.1.14 Arduino Uno

Arduino uno adalah platfrom mikronkotroler yang dirancang untuk memudahkan pengembangan pada sistem elektronik. Arduino dapat dihubungkan dalam komputer melalui kabel USB untuk proses pemograman dan pengiriman data, sehingga banyak digunakan dalam pengembangan sitem otomatis, dan monitoring yang memerlukan intergrasi antara prangkat keras dan perangkat lunak [15].

Arduino uno digunakan sebagai dasar konsep mikrokontroler untuk mengolah data dari perangkat keras, komunikasi jaringan untuk mengirimkan data gambar ke server atau website melalui API untuk kebutuhan monitoring secara *real time*.



Gambar 2.10 Arduino Uno

2.1.15 Hypertext Transfer Protocol Secure (HTTPS)

HTTPS adalah protokol komunikasi yang digunakan untuk mengamankan pertukaran data antara client dan server melalui enkripsi *SSL/TLS*. Dengan *HTTPS*, data yang dikirimkan melalui jaringan internet yang akan dienkripsi sehingga tidak dapat mudah di sadap. *HTTPS* merupakan mekanisme utama untuk menjaga kerahasiaan, integritas, dan autentikasi selama data proses transmini.

Dalam penelitian ini, *HTTPS* dimanfaatkan untuk sebagai media komunikasi yang aman antara perangkat *API* dan *server*. Protokol ini dapat membantu menjaga data gambar dan informasi monitoring yang dikirim ke website agar tetap terlindungi selama proses pengirimannya berlangsung.

2.1.16 Box Arduino

Box Arduino adalah wadah atau kotak pelindung yang digunakan untuk menempatkan rangkaian arduino dan komponen elektronik pendukung lainnya agar terlindung dari gangguan fisik, debu, kelembapan, dan benturan yang dapat mengganggu kinerja sistem. Penggunaan kotak Arduino juga membuat instalasi perangkat lebih mudah, sehingga lebih mudah untuk menjalankan perangkat, dan mengembangkan sistem.

Dalam penelitian yang menggunakan modul Arduino atau ESP32-CAM, kotak berfungsi sebagai tempat penyimpanan komponen utama seperti mikrokontroler, modul sensor, kabel penghubung, dan catu daya. Selain memberikan perlindungan, kotak juga membantu menjaga kestabilan sistem dengan mengurangi risiko terjadinya *short circuit* (hubungan singkat) karena kontak langsung dengan benda luar.



Gambar 2.11 Box Arduino

2.1.17 JavaScript Object Notation (JSON)

JSON adalah format pertukaran data yang ringan, mudah dibaca oleh manusia, dan mudah diproses oleh mesin. *JSON* banyak digunakan dalam komunikasi client server, terutama dalam pengiriman data melalui API. *JSON* Ini data yang lebih sederhana dan ukuran file yang lebih kecil dari *XML*. Studi tersebut menunjukkan bahwa *JSON* dapat meningkatkan kecepatan parsing aplikasi berbasis web.

Dalam penelitian ini, *JSON* digunakan sebagai untuk mengemas data sebelum dikirimkan melalui API ke *server*. *JSON* ini juga digunakan untuk data monitoring yang disusun secara lebih terstruktur sehingga memudahkan proses pengolahan dan penyajian informasi pada website.

2.1.18 Application Programming Interface (API)

Application Programming Interface (API) adalah antarmuka yang dirancang oleh pengembangan sistem untuk memungkinkan berkomunikasi dan pertukaran data antara berbagai aplikasi yang berbeda secara terprogram. Sebagai penghubung, API memungkinkan pengguna untuk mengakses fungsi atau layanan tertentu tanpa mengetahui proses internal sistem. API digunakan untuk berkomunikasi antara modul ESP32-CAM dan server. Dalam penelitian ini, API digunakan sebagai media komunikasi antara modul ESP32-CAM dengan server, data gambar yang dikumpulkan dikirim, disimpan dalam database, dan ditampilkan melalui website dengan efisien [16].

2.1.19 Server

Server adalah sebuah sistem komputer yang di dalamnya bisa diakses oleh perangkat lain (*client*) yang terhubung dalam jaringan. Perangkat ini dapat

meminta layanan seperti penyimpanan file, komunikasi jaringan, mengakses web, dan basis data. Dengan *server* ini dapat memproses permintaan untuk mengirimkan hasilnya kembali. *Server* biasanya menggunakan sistem yang dirancang khusus untuk pengolahan jaringan [17].

Selain itu, server biasanya menggunakan sistem yang dirancang khusus untuk membantu pengelolaan jaringan agar dapat beroperasi secara optimal, stabil, dan memiliki tingkat keamanan yang tinggi. Karena kemampuan ini, server menjadi komponen penting dalam membantu pertukaran data dan pengelolaan informasi pada berbagai sistem berbasis jaringan.

2.1.20 Arduino IDE

Arduino IDE adalah media pemrograman pada papan yang harus diprogramkan. Sebagai perangkat lunak dapat digunakan untuk mengembangkan sketsa pemrograman [18]. Dalam penelitian ini, Arduino IDE dapat digunakan sebagai media pengembangan untuk mengunggah pemrogram modul ESP32-CAM dalam proses pengambilan gambar, pengolahan data dan pengiriman data gambar ke website melalui API.

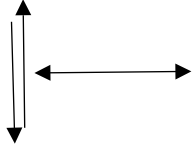
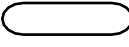
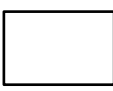
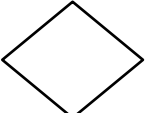


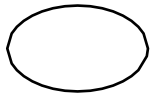
Gambar 2.12 Arduino IDE

2.1.21 Flowchart

Flowchart adalah diagram yang digunakan untuk menggambarkan proses atau langkah-langkah kerja sesuatu sistem dengan menggunakan simbol-simbol tertentu. Diagram ini membantu menunjukkan alur proses secara logis yang dihubungkan dengan garis panah untuk menunjukkan arah aliran data. *Flowchart* tidak hanya berfungsi sebagai media komunikasi, tetapi juga dapat memudahkan perancang dan pengembangan terhadap meningkatkan pemahaman mengenai bagian-bagian dari *flowchart*. *Flowchart* memiliki beberapa simbol, yang terbuat dalam table dibawah ini.

Tabel 2.1 Simbol-Simbol *Flowchart*

No	Simbol	Nama Simbol	Fungsi
1		<i>Flow</i>	Digunakan untuk menggabungkan antar simbol-simbol yang satu dengan simbol yang lain.
2		<i>Terminal</i>	Simbol yang menyatakan awal atau akhir suatu program.
3		<i>Output/Input</i>	Simbol yang menyatakan proses input atau output tanpa tergantung peralatan.
4		<i>Process</i>	Simbol yang menyatakan suatu proses yang dilakukan komputer
5		<i>Predefined Process (Sub Program)</i>	Simbol untuk pelaksanaan suatu bagian (sub-program) atau prosedur.
6		<i>Decision</i>	Simbol yang merupakan kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban yaitu ya dan tidak.

7		<i>Connector</i>	Digunakan untuk penghubung alur dalam halaman yan
8		<i>Document</i>	Simbol yang menyatakan bahwa input berasal dari dokumen dalam bentuk fisik, atau output yang perlu dicetak.

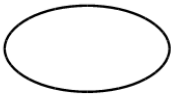
2.1.22 *Unified Modeling Language (UML)*

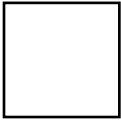
Unified modeling language (UML) adalah Bahasa pemodelan standar yang digunakan untuk memvisualisasikan, merancang, dan mendokumentasikan sistem perangkat lunak secara visual. UML membantu dalam proses analisis dan perancangan sistem yang dapat menghubungkan antar komponen sebelum sistem diimplementasi.

1. Use Case Diagram

Use Case Diagram digunakan untuk menggambarkan interaksi antara actor dengan system yang dikembangkan.

Tabel 2.2 *Use Case Diagram*

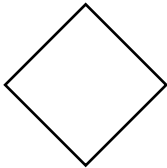
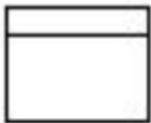
No	Simbol	Nama Simbol	Keterangan
1.		Actor	Merepresentasikan pengguna system (Admin dan Operator)
2.		Use Case	Menunjukkan abstraksi dan interaksi antara system dan aktor
3.		Association	Menghubungkan antara actor dan use case

4.		System Boundry	Menunjukkan batas system yang dikembangkan
----	---	----------------	--

2. Activity Diagram

Activity diagram digunakan untuk menggambarkan alur aktivitas atau proses bisnis yang terjadi di dalam system secara berurutan.

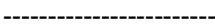
Tabel 2.3 Activity Diagram

No	Simbol	Nama Simbol	Keterangan
1.		Status Awal	Sebuah diagram aktivitas memiliki sebuah status awal.
2.		Aktifitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
3.		Decisio	Percabangan dimana ada pilihan aktivitas yang lebih satu
4.		Control Flow	Menunjukkan alur proses
5.		Final State	Menandakan akhir proses
6.			Memisahkan aktivitas pengguna dan sistem

3. Sequence Diagram

Sequence diagram digunakan untuk menggambarkan urutan intraksi antar objek atau komponen sistem berdasarkan waktu.

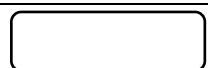
Tabel 2.4 Sequence Diagram

No.	Simbol	Nama Simbol	Keterangan
1.		Actor	Pengguna sistem
2.		Lifeline	Komponen yang digambarkan garis putus terhubung dengan objek
3.		Message	Pesan atau permintaan antar objek
4.		Activation	Proses durasi aktivasi sebuah operasi

4. Class Diagram

Class diagram digunakan untuk menggambarkan struktur data dan hubungan antar kelas dalam sistem.

Tabel 2.5 Class Diagram

No	Simbol	Nama Simbol	Keterangan
1.		Class	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama
2.		Association	Yang menghubungkan antara objek satu dengan objek lainnya.