

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Penelitian terdahulu menjadi dasar dalam pengembangan sistem pada penelitian ini, terutama dalam penggunaan ESP32-CAM sebagai pemantauan visual dan sensor laser sebagai pendeteksi gangguan. Namun, masih terdapat beberapa aspek yang dapat dikembangkan, seperti integrasi fitur monitoring, pengelolaan data, tampilan website, serta penyediaan informasi keamanan secara lebih lengkap. Oleh karena itu, penelitian ini mengembangkan sistem pemantauan keamanan gudang elektronik menggunakan sensor laser dan ESP32-CAM berbasis website dengan fitur monitoring, live camera, penyimpanan hasil capture, pemantauan suhu dan kelembapan, serta pengelolaan data secara terintegrasi.

Penelitian ini dilakukan oleh Adi Prasetyo (2023) dengan judul "Sistem Monitoring Security Menggunakan Laser dan ESP32-CAM Berbasis website". Penelitian tersebut berhasil membangun sistem monitoring keamanan menggunakan sensor laser dan ESP32-CAM berbasis Internet of Things (IoT). Sistem mampu mendeteksi adanya gangguan pada area yang dipantau, mengambil gambar menggunakan ESP32-CAM, serta menampilkan informasi melalui website sehingga pengguna dapat memantau kondisi gudang[7]. Hasil penelitian terdahulu, penelitian yang dilakukan oleh Adi Prasetyo (2023), diketahui bahwa sistem monitoring keamanan berbasis

IoT menggunakan sensor laser dan ESP32-CAM telah mampu melakukan pemantauan kondisi gudang melalui website

Penelitian ini dilakukan Agus Bambang Aldiansyah, Maftahatul Hakimah, dan Tukadi Tukadi (2022). Dalam penelitiannya yang berjudul “Sistem Pemantauan dan Kontrol Rumah Berbasis *Internet Of Things* (IoT)”. Penelitian ini menghasilkan sebuah perangkat monitoring mikrokontroler yang dapat diakses menggunakan sinyal RFID dan mampu mengambil gambar foto dan video. Hasil tersebut dapat ditinjau melalui web aplikasi yang terhubung dengan REST API sistem. Dan didapatkan Quality of Service: kualitas delay 24 ms sangat baik, kualitas throughput 391 Kb/s yaitu sangat bagus, kualitas jitter 0,179 ms yaitu sangat bagus, dan paket loss 0,4% sangat bagus[8].

Penelitian ini dilakukan Kristianus Catur Prasetya dan Azahari (2026) dengan judul “Sistem Pemantauan Gudang Berbasis ESP32-CAM untuk Deteksi Gerakan dan Keberadaan Objek”. Penelitian ini menghasilkan sistem keamanan gudang yang memanfaatkan ESP32-CAM untuk mendeteksi gerakan dan keberadaan objek secara otomatis. Sistem mampu mengambil gambar ketika terdeteksi adanya aktivitas mencurigakan di area gudang. Hasil penelitian menunjukkan bahwa penggunaan ESP32-CAM dapat meningkatkan proses pengawasan gudang secara real-time dan membantu mengurangi risiko kehilangan barang[9].

Penelitian ini dilakukan Kashif Saleem dan Ahmed Zohier Elhendi (2024). Dalam penelitiannya yang berjudul “Sistem Monitoring Rumah

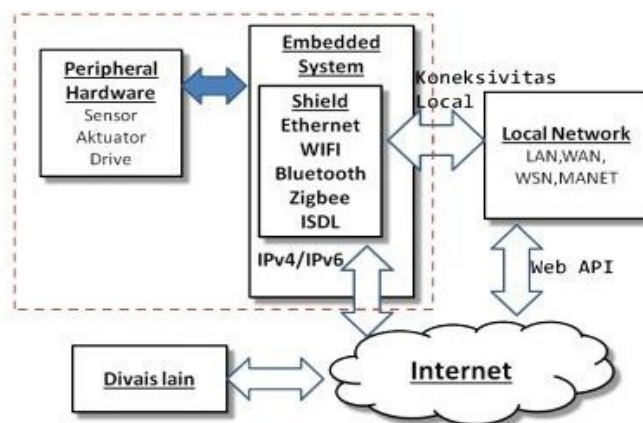
Berbasis Teknologi Cloud Computing”. Pada Penelitian ini menghasilkan sebuah sistem monitoring memanfaatkan teknologi cloud computing yang dapat diakses diberbagai gadget dengan bantuan kamera webcam untuk mengambil gambar. Webcam dikenal memiliki banyak tipe atau spesifikasi serta dapat dioperasikan secara mudah dengan biaya yang murah [10]. Hasil dari *monitoring* ini berupa gambar dengan ukuran 420 x 370 pixel. Hasil Penelitian menunjukkan bahwa sistem *monitoring* rumah ini dapat bermanfaat bagi pemilik rumah yang sering meninggalkan rumah dalam keadaan kosong. Selain itu juga dapat memberi rasa aman bagi pemilik rumah karena dapat memantau keadaan rumah dari jarak jauh hanya dengan menggunakan fasilitas internet untuk mengakses sistem tersebut.

Penelitian yang dilakukan oleh Hollanda Arief Kusuma, Setia Budi Wijaya, dan Deny Nusyirwan (2023). Dalam penelitiannya yang berjudul "Sistem Keamanan Rumah Berbasis ESP32-CAM dan Telegram Sebagai Notifikasi" menghasilkan sistem keamanan yang memanfaatkan ESP32-CAM, sensor gerak. Sistem dapat mengirimkan informasi secara otomatis ketika terdeteksi adanya aktivitas mencurigakan. Hasil penelitian menunjukkan bahwa integrasi ESP32-CAM dengan media notifikasi dapat meningkatkan efektivitas pemantauan keamanan secara real-time. Perbedaan penelitian ini dengan penelitian yang dilakukan penulis adalah media monitoring yang digunakan berupa website serta menggunakan sensor laser sebagai sistem deteksi utama pada area gudang elektronik[11].

2.2 Landasan Teori

2.2.1 IoT (Internet Of Things)

Internet of Things (IoT) adalah arsitektur terdiri dari *hardware* khusus, sistem *software*, *Web API*, protocol yang bersama membuat lingkungan yang mulus dimana *device embedded* pintar dapat terkoneksi ke *internet* semisal data sensor dapat diakses dan sistem control dapat digerakkan melalui internet. *Device* dapat terhubung ke internet menggunakan berbagai cara seperti *Ethernet*, *WIFI*, *Bluetooth*, dan sebagainya[12]. *Device* mungkin juga tidak teknokesi dengan internet secara langsung, namun dikelompokkan dalam kluster (sebagai contoh jaringan sensor) dan terhubung ke base station (terhubung ke *internet*).



Gambar 2. 1 Arsitektur IoT

Sumber: <https://journal.formosapublisher.org/>

2.2.2 Website

Website atau situs dapat diartikan sebagai kumpulan halaman - halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman. Hubungan antarsatu halaman web dengan halaman *web* yang lainnya disebut *hyperlink*, sedangkan teks yang dijadikan media penghubung disebut *hypertext*[13]. Ada beberapa hal yang dipersiapkan untuk membangun *website* gratis, maka harus tersedia unsur-unsur pendukungnya sebagai berikut:

1. Nama *domain* (*Domain name/URL-Uniform Resource Locator*)
2. Gudang *website*
3. CMS (*Content Management System*)



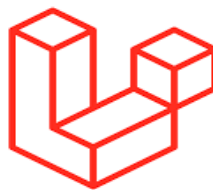
Gambar 2. 2 Website

Sumber: <https://png.pngtree.com/png>

2.2.3 Laravel

Laravel merupakan *framework* PHP *open-source* yang digunakan untuk membangun aplikasi berbasis web secara cepat, terstruktur, dan efisien. *Framework* ini menyediakan berbagai fitur seperti sistem

routing, template engine Blade, autentikasi, migrasi database, serta manajemen *session* yang memudahkan pengembang dalam membuat aplikasi modern. Laravel menerapkan konsep MVC (*Model View Controller*) sehingga kode program menjadi lebih rapi, mudah dikembangkan, dan mudah dipelihara. Selain itu, Laravel juga mendukung berbagai sistem database seperti MySQL, PostgreSQL, dan SQLite. *Framework* ini banyak digunakan karena memiliki dokumentasi yang lengkap, keamanan yang baik, serta mampu mempercepat proses pengembangan aplikasi *web* dinamis[14].



Gambar 2. 3 Laravel

Sumber: <https://download.logo.wine/logo/Laravel-Logo.wine.png>

2.2.4 PHP

PHP atau kependekan dari *Hypertext Preprocessor* adalah salah satu bahasa pemrograman *open source* yang sangat cocok atau dikhususkan untuk pengembangan *web* dan dapat ditanamkan pada sebuah skripsi HTML. Bahasa PHP dapat dikatakan menggambarkan beberapa bahasa pemrograman seperti C, Java, dan Perl serta mudah untuk dipelajari. PHP merupakan bahasa *scripting server – side*, dimana

pemrosesan datanya dilakukan pada sisi server. Sederhananya, serverlah yang akan menerjemahkan skrip program, baru kemudian hasilnya akan dikirim kepada *client* yang melakukan permintaan[15]. Adapun pengertian lain PHP adalah akronim dari *Hypertext Preprocessor*, yaitu suatu bahasa pemrograman berbasis kode-kode (*script*) yang digunakan untuk mengolah suatu data dan mengirimkannya kembali ke *web browser* menjadi kode HTML”.



Gambar 2. 4 Logo PHP

Sumber: <https://www.svgrepo.com/show/303656/php-logo.svg>

2.2.5 CSS

CSS adalah singkatan dari *Cascading Style Sheet* yaitu dokumen web yang berfungsi mengatur elemen HTML dengan berbagai property yang tersedia sehingga dapat tampil dengan berbagai gaya yang diinginkan. Sebagian orang menganggap CSS bukan termasuk salah satu bahasa pemrograman karena memang strukturnya yang sederhana, hanya berupa kumpulan-kumpulan aturan yang mengatur style elemen HTML[16]. Cara kerja CSS dalam memodifikasi HTML dengan memilih elemen yang akan diatur kemudian memberikan property yang sesuai dengan tampilan yang

diinginkan. pada elemen HTML, skrip CSS terdiri atas 3 bagian yaitu Selector untuk memilih elemen yang akan diberi aturan, property yang merupakan aturan yang diberikan dan value sebagai nilai dari aturan yang diberikan.



Gambar 2. 5 Logo CSS

Sumber: <https://zonalogo.com/assets/css-logo-png-svg.webp>

2.2.6 JavaScript

JavaScript adalah bahasa yang digunakan untuk membuat program yang digunakan agar dokument HTML yang ditampilkan pada sebuah *Browser* menjadi lebih interaktif. *JavaScript* memberikan beberapa fungsionalitas ke dalam halaman web, sehingga dapat menjadi sebuah program yang disajikan dengan menggunakan antar muka *web*. *JavaScript* merupakan bahasa *script*, yaitu bahasa yang tidak memerlukan kompiler untuk dapat menjalankannya, tetapi cukup dengan Interpreter. Tidak perlu ada proses kompilasi terlebih dahulu agar program dapat dijalankan[17]. *Browser web Netscape Navigator* dan *Internet Explorer* adalah salah satu contoh dari salah banyak interpreter, karena kedua browser ini telah dilengkapi dengan Interpreter *JavaScript*. Tetapi tidak semua browser *web* dapat menjadi

interpreter *javascript* karena belum tentu browser tersebut dilengkapi dengan interpreter *JavaScript*.



Gambar 2. 6 Logo JavaScript

Sumber: [https://1000logos.net/ JavaScript-Logo.png](https://1000logos.net/JavaScript-Logo.png)

2.2.7 MySQL

MySQL adalah salah satu jenis *database* yang banyak digunakan untuk membuat aplikasi berbasis web yang dinamis. MySQL termasuk jenis RDBMS (*Relational Database Management Sistem*). MySQL ini mendukung Bahasa pemrograman PHP. MySQL juga mempunyai query atau bahasa SQL (*Structured Query Language*) yang simple dan menggunakan *escape character* yang sama dengan PHP. MySQL adalah sebuah implementasi dari sistem manajemen basis data relasional (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (*General Public License*). Pengguna dapat secara bebas menggunakan MySQL, namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan yang bersifat komersial[18].



Gambar 2. 7 MySQL

Sumber: <https://download.logo.wine/logo/MySQL.png>

2.2.8 Visual Code

Visual code *software* yang sangat ringan, kuat sumbernya berjalan dari desktop. Muncul dengan Builtpin dukungan untuk JavaScript, naskah dan node.js dan memiliki array beragam yang tersedia untuk Bahasa lain seperti C++, C#, Python, dan PHP[19].



Gambar 2. 8 Logo Visual Code

Sumber: <https://e7.pngegg.com/pngimages.png>

2.2.9 API Telegram

API Telegram merupakan layanan antarmuka pemrograman aplikasi (*Application Programming Interface*) yang memungkinkan sistem untuk mengirim dan menerima pesan secara otomatis melalui *platform* Telegram menggunakan bot. Dalam sistem pemantauan keamanan gudang, API Telegram dapat dimanfaatkan sebagai media notifikasi *real-time* yang mengirimkan informasi berupa pesan teks,

gambar, maupun peringatan kepada pengguna ketika terjadi aktivitas tertentu pada sistem[20]. Dengan adanya fitur ini, proses pemantauan menjadi lebih cepat, efisien, dan dapat diakses kapan saja melalui *smartphone*, sehingga meningkatkan efektivitas sistem keamanan gudang berbasis IoT dan *website*.



Gambar 2. 9 Logo Telegram

<https://download.logo.wine/logo/Telegram.png>

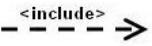
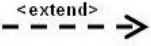
2.2.10 UML

UML (Unified Modeling Language) adalah “bahasa” pemodelan untuk sistem atau perangkat lunak yang berparadigma “berorientasi objek”. Pemodelan (modeling) sesungguhnya digunakan untuk menyederhanakan permasalahan-permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipelajari dan dipahami[21]. Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut:

1. Use Case Diagram

Use case diagram adalah teknik untuk menunjukkan hubungan antara pengguna sistem dengan sistem itu sendiri. Hasilnya berupa skema sederhana untuk memudahkan *user* membaca dan memahami informasi yang diberikan.

Tabel 2. 1 Use Case Diagram

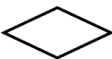
Gambar	Keterangan
	Abstraksi dan interaksi antara sistem dan aktor.
	Mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan <i>use case</i> .
	<i>Generalization</i> artinya hubungan descendent atau objek anak membagiakan struktur data dan perilakunya dari objek induk, sedangkan objek induk ini berarti objek yang berada diatasnya atau biasa disebut ancestor.
	<i>Dependency</i> artinya suatu proses ketika hubungan pada suatu elemen mandiri atau biasa disebut sebagai independent, yang kemudian mempengaruhi lelemen lain yang tidak bergantung pada elemen mandiri tersebut (<i>independent</i>).
	<i>Include</i> berfungsi untuk mengkategorikan <i>use case</i> sumber dengan cara eksplisit.
	<i>Extend</i> berguna untuk mengkategorikan atau menspesifikasikan apabila <i>use case</i> targer melakukan perluasan perilakudari sumber ke suatu titik yang telah diberikan.

2. Activity Diagram

Activity diagram memodelkan alur kerja (*work flow*) sebuah proses bisnis dan urutan aktivitas dalam suatu proses yang

dapat dimodelkan sebuah alur kerja dari satu aktivitas ke aktivitas lainnya atau dari satu aktivitas kedalam keadaan sesaat (*state*).

Tabel 2. 2 Activity Diagram

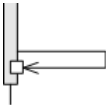
Gambar	Keterangan
Status Awal 	<i>Start Point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktivitas.
Aktivitas 	<i>Activities</i> , menggambar kan suatu proses/kegiatan bisnis.
Percabangan / <i>decision</i> 	<i>Decision Points</i> , menggambarkan pilihan untuk pengambilan keputusan, true atau false.
Penggabungan / <i>join</i> 	<i>Join</i> (penggabungan) atau rake, digunakan untuk menunjukkan adanya dekomposisi.
Status akhir 	<i>End Point</i> , akhir aktivitas.

3. Sequence Diagram

Sequence Diagram adalah suatu diagram yang menjelaskan interaksi objek dan menunjukkan atau memberi tanda komunikasi diantara objek tersebut.

Tabel 2. 3 Sequence Diagram

Gambar	Keterangan
	<i>Entity Class</i> , Bagian dari sistem yang berisi kumpulan entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.

Gambar	Keterangan
	<i>Boundary Class</i> , berisi kumpulan kelas yang menjadi interfaces atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan form entry dan form cetak.
	<i>Control class</i> , suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.
	<i>Message</i> , simbol mengirim pesan antar class.
	<i>Recursive</i> , menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.
	<i>Activation</i> , mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivasi sebuah operasi.
	<i>Lifeline</i> , garis titik-titik yang terhubung dengan objek, sepanjang lifeline terdapat activation

4. Class Diagram

Class diagram ini menggambarkan struktur dan deskripsi serta hubungan antar class diagram tersebut.

Tabel 2. 4 Class Diagram

<i>Multiplicity</i>	Penjelasan
1	Satu dan hanya satu
0..*	Boleh tidak ada atau 1 atau lebih
1..*	1 atau lebih
0..1	Boleh tidak ada, maksimal 1
n..n	Batasan antara. Contoh 2..4 mempunyai arti minimal 2 maksimal 4