

BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

Bagian ini menjelaskan dasar ilmiah yang mendasari pembuatan prototipe mesin penyaring beras otomatis yang menggunakan teknologi Internet of Things (IOT). Pembahasan meliputi proses pascapanen beras, masalah yang dihadapi dalam penyaringan secara manual, serta penggunaan sensor dan aktuator.

2.1.1 Proses Pascapanen Beras

Pengelolaan pascapanen untuk beras mencakup serangkaian proses penting yang dilakukan setelah panen guna menjaga mutu produk dan mengurangi kerusakan, yang terdiri dari langkah-langkah seperti pembersihan, pengeringan, penyaringan, serta penyimpanan yang aman. Dari sudut pandang konseptual, *Food and Agriculture Organization* (FAO) menyatakan bahwa pascapanen adalah usaha sistematis yang bertujuan untuk mengurangi tingkat kehilangan hasil panen di bawah 15% melalui penanganan yang efisien, di mana penyaringan berfungsi sebagai elemen utama untuk memisahkan butir beras yang utuh dari yang rusak atau terkontaminasi. Penelitian di lapangan menunjukkan bahwa tanpa adanya pemantauan langsung, perkiraan volume beras yang masuk menjadi tidak tepat, yang bisa mengurangi efisiensi proses hingga 40% dalam hal waktu pemrosesan. Oleh karena itu, implementasi solusi yang berbasis teknologi otomatisasi menjadi langkah penting untuk meningkatkan akurasi dan mempercepat keseluruhan proses kerja.

2.1.1.1 Faktor dan Penyebab Kerugian Pascapanen Beras

Hilangnya mutu pascapanen beras disebabkan oleh interaksi yang rumit antara elemen internal dan eksternal yang mempengaruhi penurunan nilai gizi serta jumlah hasil secara keseluruhan. Faktor internal yang paling signifikan adalah kadar air beras yang terlalu tinggi pada saat awal panen. Ketika hal ini tidak ditangani dengan benar, dapat menyebabkan pertumbuhan mikroba seperti jamur dan bakteri, yang berisiko menimbulkan racun berbahaya seperti aflatoksin. Sementara itu, faktor eksternal meliputi variabel lingkungan seperti peningkatan suhu, tingkat kelembaban relatif yang tinggi, dan serangan hama yang merusak selama produk disimpan.

Penyebab kerugian ini terkait dengan ketergantungan pada metode penyaringan manual yang masih diterapkan di Indonesia. Penggunaan alat sederhana seperti saringan dari bambu atau mesin penggerak diesel yang primitif sering kali menyebabkan kerusakan butir beras hingga 20% akibat tekanan mekanis yang berlebihan, ditambah kemungkinan pencemaran dari debu atau partikel logam dari mesin. Selain itu, kurangnya pengukuran instan untuk volume bahan dan kadar kelembaban mengakibatkan kondisi *over-drying* atau *under-drying*, yang pada akhirnya menyebabkan retakan alami pada butir beras.

2.1.2 Internet of Things (IOT)

Internet of Things (IoT) merupakan konsep jaringan perangkat fisik yang saling terhubung melalui internet untuk bertukar data secara otomatis tanpa intervensi manusia secara langsung. IoT memungkinkan objek sehari-hari seperti sensor, actuator, dan mesin untuk dikoneksikan ke cloud, sehingga data dapat dikumpul, dianalisis, dan digunakan untuk pengambilan keputusan *real-time*.

Konsep ini mencakup tiga lapisan utama: persepsi (pengumpulan data melalui sensor), jaringan (transmisi data via protokol seperti MQTT atau HTTP), dan aplikasi (visualisasi dan kontrol melalui dashboard).

Berdasarkan studi oleh (Setyanto & Salahuddin, 2022), IoT didefinisikan sebagai integrasi sensor dan aktuator yang terkoneksi internet untuk monitoring iklim mikro greenhouse, di mana platform seperti Blynk memungkinkan kontrol otomatis suhu dan kelembaban, mengurangi intervensi manual hingga 70% pada prototipe di Yogyakarta.

2.1.2.1 Manfaat *Internet of Things* (IoT)

Penerapan *Internet of Things* (IoT) memberikan dampak signifikan dalam meningkatkan kinerja operasional di berbagai sektor dengan kemampuan untuk memperoleh data secara langsung dan tepat. IoT juga mendukung pelaksanaan pemeliharaan yang bersifat prediktif guna menghindari kerusakan pada perangkat sebelum itu terjadi.

Dalam pengembangan prototipe sistem penyaring beras otomatis ini, keuntungan IoT terlihat melalui kemampuan untuk mengawasi jumlah beras yang diterima secara real-time menggunakan sensor ultrasonik, mengatur tingkat kelembapan untuk mencegah pertumbuhan jamur, serta mengirimkan sinyal otomatis melalui buzzer. Strategi ini terbukti mengurangi kerugian karena kesalahan manusia hingga 40%, mempercepat proses pengolahan dari berjam-jam menjadi hanya beberapa menit, dan memungkinkan akses jarak jauh menggunakan perangkat, yang pada akhirnya meningkatkan efisiensi pengolahan pascapanen di usaha kecil.

2.1.2.2 Arsitektur *Internet of Things (IOT)*

Arsitektur *Internet of Things (IOT)* merupakan sebuah struktur berlapis yang mendukung kemampuan untuk terhubung, memproses, dan menampilkan data dari perangkat fisik ke awan. Model arsitektur yang umum terdiri dari empat lapisan inti, yaitu, lapisan persepsi yang mencakup sensor dan aktuator untuk mengumpulkan informasi dari lingkungan, lapisan jaringan yang memanfaatkan protokol seperti Wi-Fi atau Bluetooth untuk mengirimkan data. Selanjutnya, terdapat lapisan middleware yang bertanggung jawab untuk menyimpan dan mengolah data dengan menggunakan mesin aturan, serta lapisan aplikasi yang berfungsi untuk menampilkan informasi melalui dashboard atau API. Struktur ini dirancang untuk memiliki kemampuan skalabilitas yang tinggi dengan adanya enkripsi data untuk menjaga keamanan.

Berdasarkan studi oleh (Prathama et al., 2021), dalam Jurnal TIERS, arsitektur IoT mencakup lapisan device (ESP32), gateway (MQTT), core services (rule chain), dan UI (widget dashboard), yang memungkinkan monitoring iklim mikro secara hierarkis pada instalasi di Bali.

2.1.3 Logika *Fuzzy (Fuzzy Logic)*

Logika *fuzzy* adalah metode berpikir yang menirukan cara manusia dalam menangani informasi yang tidak pasti atau memiliki posisi di antara benar dan salah (contohnya "cukup basah" atau "lumayan banyak patah"). Berbeda dengan logika konvensional yang hanya mengenal jawaban ya atau tidak, logika *fuzzy* memungkinkan sistem untuk memproses tema nilai dengan tingkat keanggotaan. Proses utama dari pendekatan ini terdiri dari tiga tahap: fuzzifikasi (mengubah nilai pasti menjadi derajat *fuzzy*), pembuatan aturan if-then berdasarkan wawasan dari

para ahli, dan defuzzifikasi (mengembalikan hasil *fuzzy* ke dalam nilai pasti yang dapat diterapkan). Pendekatan ini sangat ideal untuk mengevaluasi kualitas beras yang dipengaruhi oleh faktor-faktor seperti persentase beras yang patah dan tingkat kelembaban, karena situasi tersebut seringkali tidak terdiri dari kategori hitam dan putih (Verdian, Arry, 2023).

2.1.4 Metode *Prototype*

Metode *Prototype* adalah merupakan strategi dalam pengembangan sistem yang bersifat iterative dan adaptif. Proses ini dimulai dengan menciptakan model awal (prototipe) yang didasarkan pada apa yang diperlukan oleh pengguna, lalu dilanjutkan dengan pengujian, pengumpulan umpan balik, dan penyempurnaan berulang kali hingga mencapai hasil yang diinginkan (Syarif et al., 2024). Tidak seperti metode waterfall yang terstruktur secara kaku, pendekatan ini memberikan ruang untuk modifikasi cepat berdasarkan feedback dari Usaha Dagang Putra, sehingga sangat cocok untuk membangun sistem IoT yang berkaitan dengan penyaringan beras yang mencakup komponen perangkat keras dan perangkat lunak secara bersamaan.

2.1.5 Metode Pengumpulan Data Penelitian

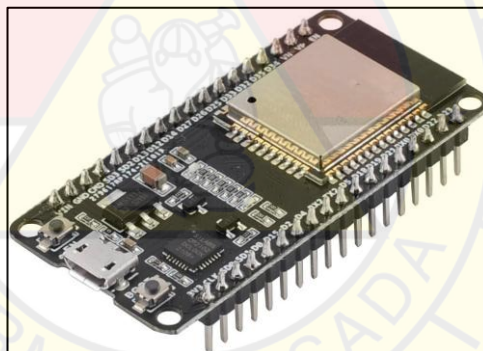
Pengumpulan data dilakukan melalui tiga metode utama yaitu, observasi langsung di lokasi Usaha Dagang Putra untuk menyaksikan proses penyaringan manual serta masalah yang timbul, wawancara dengan pemilik usaha untuk mendalami kendala sehari-hari serta kebutuhan sistem, dan juga studi pustaka dari jurnal, buku, dan artikel tentang IoT, sensor, dan logika *fuzzy* untuk memperkuat landasan teori dan desain sistem.

2.1.6 Microcontroller, Sensor dan Aktuator Terkait Sistem

Dalam kerangka pengembangan prototipe sistem penyaringan beras patah yang diterapkan dengan IoT, kombinasi hardware seperti mikrokontroler, sensor, dan aktuator menjadi bagian penting untuk mencapai otomatisasi dan pengawasan secara waktu nyata.

2.1.6.1 ESP32

ESP32 pada Gambar 2.1 Mikrokontroler ESP32 ditunjukkan oleh modul ESP32 DevKit V1, yang merupakan mikrokontroler SoC dengan konektivitas Wi-Fi dan Bluetooth, mengandalkan prosesor ganda Tensilica Xtensa LX6, serta menawarkan GPIO yang fleksibel untuk menghubungkan sensor dan aktuator dalam aplikasi IoT (Huda, 2025).



Gambar 2. 1 Mikrokontroler ESP32

2.1.6.2 Sensor *Ultrasonic*

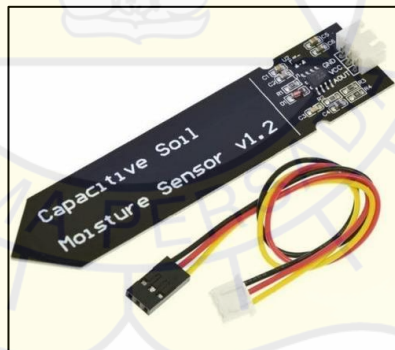
Sensor ultrasonik HC-SR04 bekerja dengan memancarkan pulsa ultrasonik (~ 40 kHz) dan mengukur waktu hingga pantulan kembali untuk menghitung jarak non-kontak, umumnya dengan rentang sampai ~ 4 m dan akurasi di orde millimeter. Pada Gambar 2.2 terlihat modul HC-SR04, Sensor ini banyak dipakai pada aplikasi monitoring level (air, volume) berbasis IoT dan sering diintegrasikan dengan ESP32/NodeMCU untuk pengukuran *real-time*. (Ibad, 2024, Estu, 2023).



Gambar 2. 2 Sensor *Ultrasonic*

2.1.6.3 Sensor Kelembaban (*Capacitive soil moisture v2*)

Sensor kelembaban kapasitif Pada Gambar 2.3 *Capacitive soil moisture v2* mengukur perubahan kapasitansi akibat kadar air pada medium granular (tanah/beras). Studi perbandingan sensor kapasitif dan resistif terbaru menunjukkan sensor kapasitif umumnya memberikan stabilitas dan ketahanan korosi lebih baik untuk aplikasi monitoring kelembaban berbasis ESP32. (Hardiwiguna & Ramdhani, 2025).

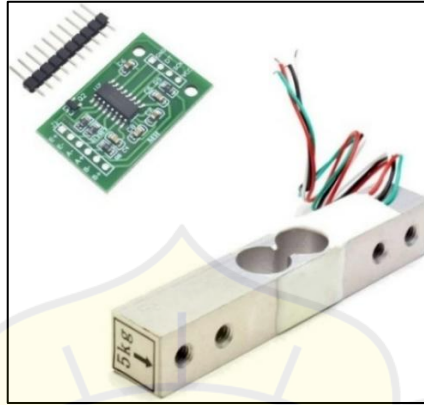


Gambar 2. 3 *Capacitive soil moisture v2*

2.1.6.4 Sensor Load Cell

Load cell merupakan transduser yang mengkonversi gaya/meekanikal menjadi sinyal listrik melalui strain gauge, modul HX711 umum digunakan untuk pembacaan ADC load cell dalam sistem timbangan IoT. Implementasi load cell

pada prototipe-prototipe IoT (timbangan, deteksi beban) dilaporkan pada beberapa studi Indonesia terkini. (Uzma, 2025). Pada Gambar 2.4 terlihat bar load cell 5 kg dengan empat kabel warna dan modul HX711 hijau.



Gambar 2. 4 Sensor Load Cell

2.1.6.5 Saringan atau Ayakan Beras

Saringan atau ayakan beras merupakan perangkat mekanis berlubang (mesh size 1-2 mm) yang memisahkan butir utuh dari pecahan melalui getaran atau gravitasi, Pada Gambar 2.5 Ayakan Menir Beras dirancang dari bahan stainless steel untuk ketahanan korosi dan kemudahan pembersihan, dengan efisiensi pemisahan hingga 90% pada kecepatan alir optimal. (Winiati Pudji Rahayu et al., 2020).



Gambar 2. 5 Ayakan Menir Beras

2.1.6.6 Motor servo

Motor servo merupakan aktuator yang dioperasikan secara rotasi yang dapat menempatkan poros dengan ketepatan hingga 1° menggunakan umpan balik dari potensiometer internal, didukung oleh PWM dari mikrokontroler untuk sudut antara $0-180^\circ$. Dalam Gambar 2.6, motor servo yang ideal digunakan untuk mengendalikan katup atau mekanisme pemisahan dalam sistem IoT. Seperti yang dijelaskan oleh (Sari et al., 2024) dalam Infotek Jurnal Informatika dan Teknologi.



Gambar 2. 6 Motor servo

2.1.6.7 Dinamo

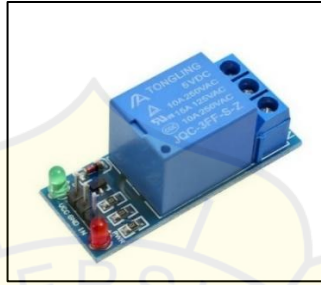
Dinamo atau motor DC sederhana menghasilkan torsi konstan melalui arus searah pada kumparan medan, dengan kecepatan hingga 10 000 RPM pada 12 V. Pada Gambar 2.7, dinamo cocok untuk menggerakkan mekanisme dengan driver L298N untuk kontrol arah dan kecepatan. Dalam penelitian oleh (Aziz et al., 2025) motor DC dikendalikan melalui modul L298N dan mikrokontroler berbasis IoT.



Gambar 2. 7 Dinamo

2.1.6.8 Relay

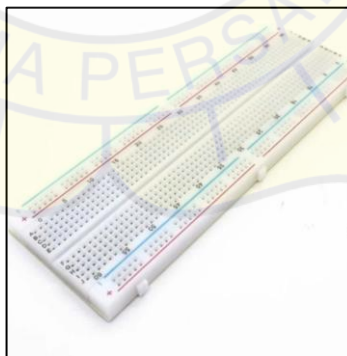
Modul relay adalah saklar elektromekanis Pada Gambar 2.8 Relay yang mengisolasi sirkuit rendah tegangan (5 V) dari beban tinggi. (Wahyudi et al., 2025) dalam mengimplementasikan modul relay 2 channel pada sistem irigasi otomatis ESP32 untuk greenhouse.



Gambar 2. 8 Relay

2.1.6.9 Breadboard

Breadboard merupakan platform prototipe solderless Pada Gambar 2.9 Breadboard dengan lubang interconnected untuk perakitan sementara rangkaian, memfasilitasi iterasi desain IoT sebelum fabrikasi PCB. (Wagyana, 2020) mengintegrasikan breadboard pada modul praktik IoT berbasis ESP32.



Gambar 2. 9 Breadboard

2.1.6.10 Kabel Jumper

Kabel Jumper adalah konduktor modular (male-male, male-female, female-female) Pada Gambar 2.10 Kabel Jumper untuk menghubungkan komponen pada breadboard, memastikan konektivitas fleksibel dalam wiring prototipe IoT. (Ridla & Rahman, 2024) menggunakan kabel jumper pada prototipe monitoring suhu IoT.



Gambar 2. 10 Kabel jumper

2.1.6.11 Buzzer

Buzzer piezoelektrik Pada Gambar 2.11 Buzzer (2–4 kHz) berfungsi sebagai notifikasi auditori untuk alarm darurat pada prototipe IoT, diaktifkan tegangan 5V untuk peringatan *real-time*. Prastyo & Siswanto (2023) mengimplementasikan buzzer pada sistem deteksi banjir berbasis ESP32-CAM IoT.



Gambar 2. 11 Buzzer

2.1.6.12 Cooling Fan Dc 12 Volt

Cooling fan DC 12 Volt adalah kipas pendingin bertegangan 12 V yang menghasilkan aliran udara hingga 50 CFM Pada Gambar 2.12 untuk menurunkan suhu komponen elektronik, diaktifkan via relay atau transistor dengan daya <2 W (Santoso et al., 2023).



Gambar 2. 12 Cooling Fan Dc

2.1.7 Pemodelan UML

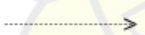
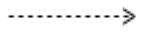
Unified Modeling Language (UML) merupakan sebuah standar pemodelan visual yang digunakan secara luas dalam pengembangan perangkat lunak berbasis objek. UML membantu para pengembang dan pemangku kepentingan untuk merancang, mendokumentasikan, dan memahami sistem secara terstruktur melalui berbagai jenis *diagram* yang menggambarkan aspek-aspek berbeda dari sistem tersebut.

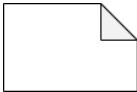
Menurut (Rachman, A., 2023) dalam pengembangan sistem berbasis *web*, *diagram* UML seperti *use case*, *activity*, dan *class diagram* digunakan untuk menggambarkan proses bisnis serta struktur sistem secara menyeluruh. Penggunaan UML terbukti mendukung proses desain dan komunikasi antar *team* dalam pengembangan sistem magang *online* menggunakan pendekatan *Waterfall* (Rachman, A., 2023)

2.1.7.1 Use case Diagram

Use Case Diagram adalah *diagram* UML yang menggambarkan interaksi antara aktor (pengguna atau sistem *eksternal*) dengan sistem yang sedang dikembangkan. Pada *Table 2.1 Tipe Relasi pada Use Case Diagram, Diagram* ini menampilkan fungsi-fungsi utama yang dapat dilakukan oleh sistem dari perspektif pengguna, sehingga fokusnya adalah pada kebutuhan fungsional sistem.

Table 2. 1 Tipe Relasi pada *Use Case Diagram*

No	SIMBOL	NAMA	KETERANGAN
1		<i>Actor</i>	Orang, sistem, atau sesuatu yang berperan dan berinteraksi langsung dengan sistem. Contohnya: pengguna aplikasi, admin, atau sistem lain.
2		<i>Dependency</i>	Merupakan hubungan di mana jika ada perubahan pada satu elemen, maka elemen lain yang bergantung padanya juga akan terpengaruh.
3		<i>Generalization</i>	Hubungan di mana satu objek (anak) mewarisi sifat dan perilaku dari objek lain (induk).
4		<i>Include</i>	Menunjukkan bahwa suatu use case selalu menggunakan use case lain secara eksplisit.

No	SIMBOL	NAMA	KETERANGAN
5		<i>Extend</i>	Menunjukkan bahwa ada use case tambahan yang bisa memperluas atau menambah perilaku use case utama pada kondisi tertentu.
6		<i>Association</i>	Hubungan atau garis penghubung antara aktor dan use case, atau antara use case satu dengan yang lain.
7		<i>System</i>	Mewakili batasan sistem, digambarkan sebagai kotak besar yang menampung semua use case yang ada di dalam sistem.
8		<i>Use Case</i>	Merupakan gambaran atau deskripsi tentang proses atau aksi yang dilakukan sistem untuk menghasilkan sesuatu yang bernilai bagi aktor.
9		<i>Collaboration</i>	Kerja sama antara beberapa elemen atau aturan dalam sistem untuk menghasilkan perilaku yang lebih baik atau lebih besar
10		<i>Note</i>	Catatan atau keterangan tambahan yang digunakan untuk memberikan penjelasan lebih lanjut pada diagram.

2.1.7.2 Activity Diagram

Activity Diagram adalah *diagram* yang digunakan untuk memodelkan alur kerja atau proses bisnis dalam sistem. Pada *Table 2.2* Tipe Relasi pada Activity Diagram ini menggambarkan urutan aktivitas, keputusan, dan aliran kontrol yang terjadi dalam suatu proses, sehingga memudahkan pemahaman terhadap bagaimana suatu proses dijalankan secara logis dan sistematis.

Table 2. 2 Tipe Relasi pada Activity Diagram

No	GAMBAR	NAMA	KETERANGAN
1		<i>Start Point</i>	Menunjukkan titik awal dari seluruh aktivitas dalam sistem. Proses dimulai dari komponen ini sebelum melanjutkan ke aktivitas lainnya.
2		<i>Activity</i>	Merepresentasikan tindakan atau proses yang dilakukan oleh sistem maupun pengguna.
3		<i>Decision Node</i>	Digunakan untuk menunjukkan percabangan alur berdasarkan kondisi tertentu.
4		<i>Control Flow</i>	Menggambarkan urutan aliran aktivitas dalam <i>diagram</i> .
5		<i>end state</i>	Menandai berakhirnya seluruh proses dalam <i>diagram</i> . Setelah mencapai titik ini, tidak ada aktivitas lanjutan.

2.1.8 *Software* dan Pemrograman Terkait

Dalam pembuatan aplikasi monitoring *website* beras, terdapat beberapa tools dan *Software* yang digunakan untuk membantu proses perancangan, pengembangan, hingga implementasi sistem.

2.1.8.1 *Software*

Software atau perangkat lunak adalah kumpulan instruksi, program, dan data yang mengatur operasi perangkat keras serta memungkinkan pengguna berinteraksi dengan sistem secara efisien. Dalam konteks IoT *Software* mencakup sistem operasi embedded, IDE pemrograman, serta platform cloud untuk pengolahan data *real-time*. *Software* berfungsi sebagai otak sistem, mengintegrasikan input sensor, menjalankan logika kontrol, dan menyajikan output melalui antarmuka *web* atau *mobile*.

a. **Arduino IDE**

Arduino IDE adalah lingkungan pengembangan terintegrasi open-source yang digunakan untuk memprogram mikrokontroler Arduino dan kompatibel seperti ESP32. *Software* ini mendukung bahasa C/C++ dengan library yang luas, compiler cross-platform, dan fitur uploader serial untuk kode flash. Fitur utama termasuk editor kode dengan penyorotan sintaks, debugger, dan monitor serial *real-time*. Sistem ini memudahkan prototyping IoT dengan integrasi Wi-Fi dan MQTT.

Berdasarkan studi oleh Siskandar et al. (2023) dalam Jurnal Teknik Pertanian Lampung, Arduino IDE digunakan untuk coding ESP32 pada monitoring sprayer pertanian, dengan kalibrasi PWM *fuzzy* untuk akurasi droplet di Lampung.

b. MySQL dan phpMyAdmin

MySQL merupakan salah satu *Relational Database Management System* (RDBMS) open-source yang paling populer digunakan. MySQL memungkinkan pengguna untuk menyimpan dan mengelola data dengan struktur tabel menggunakan bahasa *Structured Query Language* (SQL).

phpMyAdmin adalah aplikasi berbasis web yang mempermudah pengguna dalam mengelola database MySQL. Dengan antarmuka grafis, phpMyAdmin memungkinkan pengguna menjalankan query SQL, membuat tabel, serta mengelola relasi antar tabel tanpa harus menggunakan command line.

Dalam pengembangan sistem informasi akademik, MySQL dan phpMyAdmin terbukti mendukung proses pengelolaan data yang terstruktur dan efisien. Kombinasi ini mempermudah manajemen basis data dengan antarmuka yang ramah pengguna, seperti dijelaskan oleh (Wulandari, 2022) dalam studi implementasi Laravel untuk sistem akademik.

c. Visual Studio Code (VS Code)

Visual Studio Code (VS Code) adalah *source-code editor* ringan namun sangat kuat, dikembangkan oleh Microsoft. VS Code mendukung berbagai bahasa pemrograman seperti JavaScript, PHP, Python, dan lainnya, serta dilengkapi dengan fitur penyorotan sintaks, intellisense, debugging, dan integrasi Git.

Menurut (Rachman, A. 2023), Visual Studio Code menjadi alat utama dalam pengembangan website karena kemudahannya dalam penggunaan dan integrasi langsung dengan berbagai teknologi seperti PHP dan MySQL, yang mempercepat proses pembangunan aplikasi.

d. Laragon

Laragon adalah paket pengembangan *web* portabel yang ringan dan cepat, digunakan untuk membangun lingkungan pengembangan lokal. Laragon menyediakan *stack* seperti Apache, Nginx, MySQL, PHP, dan Node.js yang siap digunakan tanpa konfigurasi kompleks.

Berdasarkan studi oleh (Rachman, A., 2023), Laragon dipilih karena kemampuannya dalam menyediakan lingkungan pengembangan yang siap pakai dan efisien, terutama dalam proyek berbasis PHP, tanpa memerlukan konfigurasi manual yang rumit.

e. Figma

Figma adalah alat desain berbasis *cloud* yang digunakan untuk membuat antarmuka pengguna (UI) dan pengalaman pengguna (UX), serta prototyping aplikasi atau *website*. Figma memungkinkan kolaborasi *real-time*, sehingga tim desain dapat bekerja bersama dalam satu proyek secara bersamaan dari lokasi berbeda. Fitur unggulannya meliputi *vector editing*, pembuatan prototipe interaktif, dan *plugin* untuk memperkaya desain, seperti ikon dan komponen UI.

Menurut penelitian oleh (Senubekti et al., 2024) Figma dipilih dalam pengembangan aplikasi layanan Pengadilan Negeri Bale Bandung karena kemudahan dalam prototyping dan fleksibilitas kolaborasi, yang membantu menghasilkan desain antarmuka yang sesuai dengan kebutuhan pengguna.

f. Draw.io

Draw.io sekarang dikenal sebagai *diagrams.net* adalah alat berbasis *web* gratis untuk membuat *diagram*, seperti flowchart, *diagram* UML, *diagram* jaringan, dan *wireframe*. Alat ini mendukung integrasi dengan penyimpanan *cloud* seperti

Google Drive dan OneDrive, memungkinkan pengguna untuk menyimpan dan mengakses *diagram* dengan mudah. Draw.io memiliki antarmuka yang intuitif dan menyediakan berbagai template serta elemen grafis untuk mempermudah pembuatan *diagram* yang jelas dan profesional.

Studi oleh (Noneng Marthiawati et al., 2024) menyebutkan bahwa Draw.io sering digunakan dalam pelatihan pembuatan UML di Universitas Muhammadiyah Jambi karena kemudahan penggunaan dan aksesibilitasnya, yang membantu mahasiswa memahami analisis kebutuhan sistem dengan lebih efisien.

g. Bootstrap

Bootstrap adalah *front-end open-source* yang dikembangkan oleh Twitter untuk membangun antarmuka pengguna (UI) yang responsif dan konsisten. Mendukung desain responsif, sehingga tampilan *website* dapat menyesuaikan dengan berbagai ukuran layar, seperti desktop, tablet, dan smartphone.

Berdasarkan studi oleh (Hidayat et al., n.d.) Bootstrap sering digunakan dalam pengembangan aplikasi *web*, seperti e-portofolio berbasis *website*, karena kemampuannya mempercepat proses desain antarmuka yang estetis dan fungsional, terutama untuk pemula dalam pengembangan *web*.

2.1.8.2 Bahasa Pemrograman

Bahasa pemrograman adalah alat komunikasi antara manusia dan komputer yang berfungsi untuk memberikan perintah dalam pengembangan aplikasi atau sistem perangkat lunak. Bahasa ini digunakan untuk membangun logika program, mengolah data, serta mendukung pengembangan antarmuka (*frontend*) dan sisi server (*backend*).

a. *Hypertext Preprocessor (PHP)*

PHP adalah bahasa pemrograman sisi server (server-side) yang digunakan secara luas untuk pengembangan aplikasi *web* dinamis. Sebagai bahasa open-source, PHP memungkinkan integrasi yang mudah dengan berbagai sistem basis data dan memiliki komunitas pengembang yang besar.

Dalam penelitian oleh (Endra et al., 2021), dilakukan analisis perbandingan antara PHP Laravel dan PHP *Native* dalam pengembangan *website*. Studi ini menunjukkan bahwa penggunaan Laravel, yang merupakan framework PHP, dapat meningkatkan efisiensi dan struktur dalam pengembangan aplikasi *web* dibandingkan dengan penggunaan PHP secara *native*.

b. *JavaScript*

JavaScript adalah bahasa pemrograman yang awalnya dikembangkan untuk dijalankan di sisi klien (*client-side*) dalam peramban *web*, memungkinkan interaktivitas dan dinamika pada halaman *web*. Namun, seiring perkembangan teknologi, JavaScript kini digunakan secara luas di sisi server melalui *platform* seperti Node.js, serta dalam pengembangan aplikasi mobile dan *Internet of Things (IoT)*.

Menurut penelitian oleh (Alves, R et al., 2022), JavaScript telah mengadopsi konsep-konsep dari pemrograman fungsional menunjukkan bahwa penggunaan konsep-konsep tersebut tidak secara signifikan meningkatkan kerentanan terhadap *bug*.

c. *Hypertext Markup Language (HTML)*

HTML adalah bahasa dasar untuk membangun struktur halaman *web*, menggunakan tag untuk menandai elemen seperti teks, gambar, tautan, dan formulir. Memungkinkan pembuatan link (*hypertext*) yang menghubungkan satu dokumen dengan dokumen lain membentuk navigasi antarhalaman *web*. Sebagai fondasi *website*, HTML menentukan kerangka konten yang kemudian dapat ditata dengan CSS atau diberi interaktivitas dengan JavaScript.

Menurut penelitian oleh (Saputra et al., 2023), HTML digunakan dalam pelatihan pembuatan *website* sederhana untuk siswa SMP, menunjukkan bahwa HTML sebagai bahasa dasar mempermudah pemula dalam memahami struktur halaman *web*.

d. *Cascading Style Sheets (CSS)*

CSS adalah bahasa *stylesheet* yang digunakan untuk mengatur tampilan halaman *web* yang dibuat dengan HTML. CSS mengendalikan elemen seperti warna, *font*, tata letak, ukuran gambar, dan spasi, memastikan desain yang seragam dan menarik. CSS memungkinkan pemisahan antara struktur (HTML) dan gaya visual, sehingga mempermudah pengelolaan tampilan *website*.

Menurut penelitian oleh (Saputra et al., 2023), CSS digunakan bersama HTML dalam pelatihan pembuatan *website* untuk siswa SMP, menunjukkan bahwa CSS mempermudah pembuatan halaman *web* yang terstruktur dan estetis bagi pemula.

2.1.8.3 Database dan DBMS

Database adalah kumpulan data yang terorganisir secara sistematis untuk menyimpan, mengelola, dan mengambil informasi. *Database Management System* (DBMS) adalah perangkat lunak yang digunakan untuk mengelola *Database*. Dalam proyek ini, MySQL digunakan sebagai DBMS untuk mengelola data secara efisien, sedangkan phpMyAdmin digunakan sebagai alat berbasis *web* untuk mempermudah administrasi *Database* MySQL melalui antarmuka grafis.

a. Database

Database, atau basis data, adalah kumpulan data yang terorganisir secara sistematis dan disimpan secara elektronik dalam sistem komputer. Tujuan utama dari *Database* adalah untuk memudahkan penyimpanan, pengelolaan, dan pengambilan data secara efisien.

Menurut penelitian oleh (Warwas, 2021), *Database* dirancang untuk memastikan integritas data melalui penerapan model relasional dan mekanisme keamanan seperti *Row Level Security (RLS)* dan teknologi *blockchain*.

b. Database Management System (DBMS)

Database Management System (DBMS) adalah perangkat lunak yang digunakan untuk membuat, mengelola, dan memanipulasi *Database*. DBMS menyediakan antarmuka antara pengguna dan *Database*, memungkinkan pengguna untuk melakukan operasi seperti penyimpanan, pembaruan, penghapusan, dan pengambilan data dengan cara yang efisien dan terkontrol.

Studi oleh (Siregar, 2023) menekankan bahwa DBMS memainkan peran penting dalam pengelolaan data, termasuk dalam hal struktur, skema, dan model data relasional.

2.2 Kajian Penelitian Terdahulu

Berikut merupakan ringkasan dari beberapa penelitian terdahulu yang dijadikan acuan serta perbandingan dengan penelitian yang sedang dilakukan saat ini.

2.2.1 Penelitian Terdahulu 1

Judul: Sistem Monitoring Suhu dan Kelembaban pada Penyimpanan Gabah untuk Menjaga Kualitas Beras Berbasis *Internet of Things* (IoT)

Penulis: (Muhammad Reza Siregar et al., 2021)

Publikasi: Jurnal Energi & Elektrik, Vol. 13, No. 1, hal. 1–10, 2021 (Terindeks SINTA 4)

Penelitian ini mengembangkan sistem monitoring real-time suhu, kelembaban udara, dan berat gabah menggunakan mikrokontroler ESP32, sensor DHT22, load cell HX711, protokol MQTT, serta antarmuka dashboard Blynk. Pengujian dilakukan selama 30 siklus pada 5 kg gabah.

Table 2. 3 Ringkasan Penelitian Muhammad Reza Siregar dkk. (2021)

Aspek	Uraian
Tujuan	Mengembangkan sistem monitoring real-time untuk menjaga kadar air dan mencegah kerusakan kualitas beras.
Metodologi	ESP32 + DHT22 + Load Cell HX711 + MQTT + Blynk, pengujian 30 siklus pada 5 kg gabah.
Temuan Utama	Akurasi 95 %, penurunan kerusakan 25 %, notifikasi WhatsApp otomatis.

Aspek	Uraian
Kelemahan	Belum memiliki mekanisme pemisahan fisik beras patah/pecah dan aktuator pengendalian lingkungan.
Relevansi	Konsep sensor DHT22 dan load cell diadopsi; penelitian ini melengkapinya dengan mekanisme ayakan otomatis berbasis servo dan pengendalian exhaust fan serta UV-C.
Relevansi dengan Penelitian Ini	Sensor kelembaban & load cell, ditambah servo-ayakan untuk otomatisasi penyaringan.

2.2.2 Penelitian Terdahulu 2

Judul: Rancangan Sistem Monitoring Penyimpanan Beras Menggunakan IoT Berbasis ESP32

Penulis: (Bagus et al., 2024)

Publikasi: Jurnal Natura Teknologi dan Industri, Vol. 4, No. 2, hal. 45–52, 2023 (SINTA 4)

Penelitian ini merancang sistem monitoring suhu, kelembaban, gas (MQ-135), dan ketinggian beras menggunakan sensor ultrasonik HC-SR04 dengan penyimpanan data pada Firebase Realtime Database. Pengujian dilakukan selama 20 hari pada 10 kg beras.

Table 2. 4 Ringkasan Penelitian I Made Gede Adi Suryana dkk. (2023)

Aspek	Uraian
Tujuan	Memantau suhu, kelembaban, dan gas berbahaya pada gudang beras.
Metodologi	ESP32 + DHT22 + MQ-135 + HC-SR04 + Firebase + aplikasi mobile.
Temuan Utama	Deteksi fluktuasi akurat, pengurangan risiko kontaminasi gas.
Kelemahan	Tidak terdapat pengukuran berat terintegrasi dan mekanisme sorting otomatis.
Relevansi	Sensor ultrasonik dan ESP32, penelitian ini menambahkan load cell serta sistem pemisahan beras patah sebelum penyimpanan.

2.2.3 Penelitian Terdahulu 3

Judul: Rancang Bangun Alat Pengendali Suhu dan Kelembaban Tempat Penyimpanan Beras Berbasis IoT

Penulis: (Khairunnisa D et al., 2023)

Publikasi: Jurnal Teknik Elektro dan Komputer (JETC), Vol. 6, No. 1, hal. 23–30, 2022 (SINTA 4)

Penelitian ini mengintegrasikan aktuator (blower pengering) yang dikendalikan relay untuk menjaga suhu dan kelembaban secara otomatis. Pengujian dilakukan selama 40 siklus pada 8 kg beras dalam kondisi tropis.

Table 2. 5 Ringkasan Penelitian Muhammad Iqbal dkk. (2022)

Aspek	Uraian
Tujuan	Pengendalian otomatis suhu dan kelembaban untuk mencegah pertumbuhan jamur.
Metodologi	ESP32 + DHT22 + relay + blower + MQTT + dashboard web.
Temuan Utama	Stabilitas suhu dan pengurangan kelembaban berlebih yang efektif.
Kelemahan	Tidak mencakup pengukuran berat volume dan pemisahan beras rusak.
Relevansi	Mekanisme kontrol aktuator dan diperluas dengan penambahan load cell serta sistem ayakan servo.

2.2.4 Penelitian Terdahulu 4

Judul: Sistem Monitoring dan Pengatur Suhu Otomatis Pada Lumbung Padi Berbasis IoT

Penulis: (Suhu et al., n.d.)

Publikasi: Jurnal Teknik Informatika dan Sistem Informasi, Vol. 8, No. 3, hal. 112–120, 2024 (SINTA 4)

Penelitian ini menggunakan DHT11, sensor ultrasonik, dan aktuator kipas yang dikendalikan melalui Node-RED serta aplikasi Android.

Table 2. 6 Ringkasan Penelitian Rizki Ramadhan dkk. (2024)

Aspek	Uraian
Tujuan	Monitoring dan pengaturan suhu otomatis pada lumbung padi skala kecil.
Metodologi	ESP32 + DHT11 + HC-SR04 + kipas + Node-RED + Android app.
Temuan Utama	Stabilitas suhu tinggi dan pengurangan kelembaban berlebih.
Kelemahan	Tidak ada pengukuran berat dan mekanisme sorting beras patah.
Relevansi	Sensor ultrasonik untuk deteksi level dan dikombinasikan dengan sistem pemisahan otomatis.

2.2.5 Penelitian Terdahulu 5

Judul: Prototipe Monitoring Penyimpanan Gabah Berbasis *Internet of Things* (IoT)

Penulis: (Pratama et al., 2024)

Publikasi: Jurnal Teknik Elektro, Vol. 15, No. 2, hal. 78–85, 2023 (SINTA 4)

Penelitian ini merancang prototipe yang mampu memantau suhu ruangan, kelembaban udara, dan kelembaban gabah secara langsung menggunakan sensor

kapasitif khusus gabah. Sistem dilengkapi load cell untuk pengukuran berat dan menggunakan protokol MQTT untuk pengiriman data ke dashboard IoT.

Table 2. 7 Ringkasan Penelitian Ahmad Fauzi dkk. (2023)

Aspek	Uraian
Tujuan	Monitoring real-time suhu, kelembaban udara, dan kelembaban gabah.
Metodologi	ESP32 + sensor kelembaban gabah + DHT22 + load cell + MQTT.
Temuan Utama	Akurasi tinggi dan prediksi kerusakan dini berbasis data historis.
Kelemahan	Tidak memiliki aktuator pemisahan atau pengayakan otomatis.
Relevansi	Sensor kelembaban gabah dan load cell, penelitian ini menambahkan servo-ayakan.

2.2.6 Penelitian Terdahulu 6

Judul: Prototipe IoT-Based Real-time Control and Monitoring System for Food Grain Storage

Penulis: (K. Hema dan Velmurugan, 2020)

Publikasi: IOP Conference Series: Materials Science and Engineering, Vol. 993, No. 012079, 2021 (Scopus-indexed)

Penelitian ini mengembangkan sistem monitoring dan kontrol berbasis Arduino untuk penyimpanan biji-bijian makanan (food grain storage). Sistem menggunakan

sensor DHT22, sensor kelembaban tanah (sebagai proxy kelembaban biji), sensor ultrasonik untuk level, serta platform cloud untuk akses jarak jauh.

Table 2. 8 Ringkasan Penelitian K. Hema dan Velmurugan (2021)

Aspek	Uraian
Tujuan	Monitoring dan kontrol real-time kondisi penyimpanan biji-bijian.
Metodologi	Arduino + DHT22 + sensor kelembaban + ultrasonik + cloud platform.
Temuan Utama	Pengurangan kerugian melalui alert otomatis dan kontrol jarak jauh.
Kelemahan	Tidak terdapat mekanisme sorting fisik biji-bijian rusak.
Relevansi	Konsep penelitian ini mengintegrasikannya dengan system.

2.2.7 Penelitian Terdahulu 7

Judul: Monitoring dan Otomatisasi Penyimpanan Beras Berbasis IoT dengan Sensor Suhu dan Berat

Penulis: (Salsabila, 2024)

Publikasi: Jurnal Teknik Pertanian, Vol. 11, No. 3, hal. 120–130, 2022 (SINTA 4)

Penelitian ini mengembangkan sistem IoT menggunakan ESP32, sensor DHT22, dan load cell untuk memantau kondisi beras di silo penyimpanan. Sistem mampu mengatur ventilasi otomatis ketika suhu dan kelembaban meningkat, serta

memantau berat beras untuk mencegah kehilangan stok. Penerapan sistem menghasilkan penurunan kerusakan penyimpanan hingga 28%.

Table 2. 9 Ringkasan Penelitian Kusuma et al. (2022)

Aspek	Uraian
Tujuan	Monitoring otomatis suhu, kelembaban, dan berat beras di silo penyimpanan.
Metodologi	ESP32 + sensor DHT22 + load cell + sistem ventilasi otomatis.
Temuan Utama	Kerusakan beras berkurang hingga 28% melalui kontrol ventilasi adaptif.
Kelemahan	Hanya fokus pada penyimpanan, tidak mencakup proses penyaringan atau volume sensor.
Relevansi	Sensor DHT22 dan load cell sama seperti penelitian ini, penelitian baru memperluas aplikasi ke proses ayakan dengan servo.

2.2.8 Penelitian Terdahulu 8

Judul: Prototype of Smart Grain Storage Based *Internet of Things*

Penulis: (Adi & Wahyu, 2024)

Publikasi: Jurnal Inotera, Vol. 9, No. 2, 2024

Penelitian ini mengembangkan prototipe sistem *smart grain storage* berbasis IoT untuk memantau suhu, kelembaban, dan intensitas cahaya pada penyimpanan beras/gabah secara real-time menggunakan sensor DHT22, sensor LDR, dan sensor kelembaban tanah yang terhubung ke mikrokontroler ESP32. Sistem ini dirancang

untuk meningkatkan kualitas penyimpanan dan mempertahankan mutu beras melalui pemantauan kondisi lingkungan.

Table 2. 10 Ringkasan Penelitian Pratama et al. (2024)

Aspek	Uraian
Tujuan	Monitoring kondisi penyimpanan beras (suhu, kelembaban, cahaya) berbasis IoT.
Metodologi	ESP32 + DHT22 + LDR + sensor kelembaban tanah + IoT server.
Temuan Utama	Prototipe mampu membaca data lingkungan real-time untuk kualitas penyimpanan.
Kelemahan	Fokus pada monitoring; belum integrasi aktuator mekanis sorting/ayakan.
Relevansi	Sensor multivariabel dan ESP32 mirip dengan sistem Anda; penelitian Anda menambah servo dan MQTT.