

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Menurut Putu Edward Lim Junior, Eddy Muntina Dharma, dan Putu Trisna Hady Permana (2023) dalam jurnal *Smart Irrigation Berbasis Internet of Things* (IoT) Menggunakan Framework Firebase pada Tanaman Tomat (Studi Kasus di Desa Tegalcangkring, Kabupaten Jembrana), sistem irigasi otomatis untuk tanaman tomat dikembangkan menggunakan sensor kelembaban tanah kapasitif dan sensor suhu/kelembaban DHT22 yang terhubung ke platform Firebase, serta antarmuka aplikasi Android yang dibangun dengan MIT App Inventor untuk memantau kondisi lapangan secara *real-time*, ketika sensor mendeteksi bahwa kelembaban tanah turun di bawah ambang batas dan suhu sesuai, sistem secara otomatis mengaktifkan pompa untuk menyiram tanaman, dengan logika kontrol didasarkan pada metode fuzzy Tsukamoto, dan menunjukkan akurasi tinggi: sensor kelembaban tanah $\pm 95,22\%$ ($\sigma \pm 5,4$), sensor suhu $\pm 98,65\%$ ($\sigma \pm 0,14$) dan kelembaban udara $\pm 98,72\%$ ($\sigma \pm 0,38$), serta waktu tanggap sistem hanya 2 detik dari deteksi hingga pompa aktif.[1]

Menurut Ahmed A. Abdelmoneim, Hilda N. Kimaita, Christa M. Al Kalaany, Bilal Derardja, Giovanna Dragonetti, dan Roula Khadra (2025) dalam jurnal *IoT Sensing for Advanced Irrigation Management: A Systematic Review of Trends, Challenges, and Future Prospects*, integrasi Internet of

Things (IoT) dalam manajemen irigasi sangat penting untuk pertanian berkelanjutan. Studi ini, melalui analisis bibliometrik, mengidentifikasi peningkatan minat penelitian yang signifikan dari tahun 2020 hingga 2022, dengan fokus pada aplikasi interdisipliner dan terapan. Penelitian ini menemukan bahwa sistem irigasi cerdas berbasis IoT banyak menggunakan berbagai sensor, unit mikrokontroler umum, dan protokol komunikasi berdaya rendah untuk akuisisi data waktu nyata, yang memungkinkan wawasan *real-time* tentang kondisi tanah dan cuaca. Meskipun ada tantangan terkait privasi data dan ketersediaan infrastruktur di daerah pedesaan, jurnal ini menyoroti potensi besar IoT untuk mengoptimalkan penggunaan air dan memprediksi kebutuhan irigasi. Ke depannya, disarankan adanya integrasi IoT dengan Kecerdasan Buatan (AI), *edge computing*, dan *blockchain* untuk mengembangkan sistem pendukung keputusan yang lebih canggih dan strategi irigasi prediktif yang lebih efektif.[2]

Menurut Andi Bajiel Rifaat, Fany Sephiani, Ridwang, dan Adriani (2024) dalam jurnal Pengembangan Sistem Penyiram Tanaman Otomatis Berbasis IoT Menggunakan Sensor Suhu, Kelembapan Udara dan Kelembapan Tanah, sistem penyiraman tanaman otomatis berbasis Internet of Things (IoT) telah dikembangkan menggunakan mikrokontroler NodeMCU ESP8266 yang terintegrasi dengan sensor kelembapan tanah kapasitif, sensor kelembapan udara DHT11, dan sensor suhu DS18B20. Data dari sensor ini dikirim secara *real-time* ke aplikasi Blynk, yang juga memungkinkan penyiraman manual. Sistem ini secara otomatis mengaktifkan pompa

penyiraman ketika kelembapan tanah terdeteksi kurang dari 50% RH dan menghentikannya saat kelembapan mencapai lebih dari 65% RH. Pengujian pada tanah humus dan aluvial menunjukkan bahwa sensor suhu dan kelembapan udara memberikan hasil yang stabil dan akurat, sementara sensor kelembapan tanah memberikan informasi presisi untuk pengambilan keputusan penyiraman, secara signifikan meningkatkan efisiensi penggunaan air dan mengoptimalkan penyiraman tanaman.[3]

Menurut Wahyudi, Afu Ichsan Pradana, dan Hanifah Permatasari (2024) dalam jurnal Implementasi Sistem Irigasi Otomatis Berbasis IoT untuk Pertanian Greenhouse, sistem irigasi otomatis berbasis *Internet of Things* (IoT) telah dikembangkan untuk pertanian greenhouse menggunakan mikrokontroler ESP32 dan sensor kelembapan tanah. Data sensor dikirim ke platform *ThingSpeak* untuk pemantauan *real-time*, dan sistem mengontrol penyiraman berdasarkan tingkat kelembapan tanah yang terdeteksi. Hasil penelitian menunjukkan bahwa sistem ini mampu mengurangi penggunaan air hingga 30% dibandingkan metode konvensional, serta meningkatkan efisiensi energi melalui optimalisasi siklus penyiraman, menjadikannya solusi strategis untuk meningkatkan keberlanjutan pertanian dan menarik minat petani muda terhadap teknologi agrikultur.[4]

2.2 Landasan Teori

2.2.1 Internet of Things (IoT)

Internet of Things (IoT), merupakan sebuah konsep yang bertujuan untuk memperluas manfaat dari konektivitas internet yang tersambung secara terus-menerus yang memungkinkan kita untuk menghubungkan mesin, peralatan elektronik, dan benda fisik lainnya dengan menggunakan sensor jaringan dan aktuator untuk memperoleh data dan mengelola kinerjanya sendiri, sehingga memungkinkan perangkat tersebut untuk berkolaborasi dan bahkan bertindak berdasarkan informasi baru yang diperoleh secara independent dan dapat dikendalikan dari jarak jauh.[6]

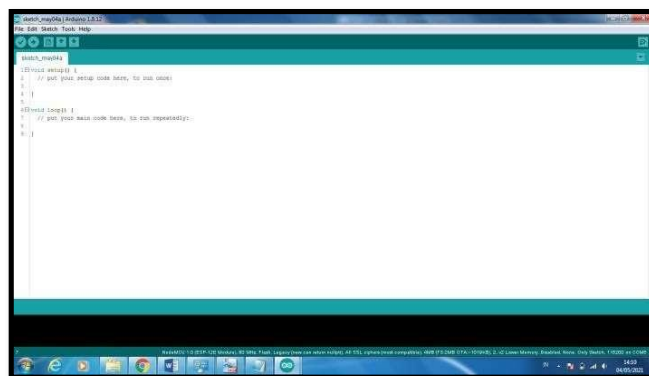
2.2.2 Monitoring

Monitoring merupakan sebuah proses yang dilakukan secara rutin untuk melakukan pengumpulan sebuah data atau dalam proses penyajian informasi yang berguna untuk mencapai suatu tujuan tertentu yang bersifat spesifik serta sistematis, yang dilakukan dengan memfokuskan pada proses dan hasilnya. Monitoring dapat di implementasikan pada alam dan lingkungan, salah satunya pada sektor pertanian guna untuk memonitoring intensitas cahaya, kelembaban suhu, pH pada suatu tanah, mengukur ketinggian suatu volume air dan kelembaban tanah secara *real-time*. Sistem monitoring dapat di terapkan dalam berbagai bidang pertanian antara lain pertanian cabai, bawang merah, tebu, padi, kacang hijau, dan lain

sebagainya. Sehingga para petani dapat mengambil keputusan yang tepat dan mengetahui pengelolaan lahan yang diperlukan bagi para pelaku usaha tani.[7]

2.2.3 Arduino Software IDE

Arduino IDE (*Integrated Development Environment*) merupakan lingkungan terintegrasi yang digunakan untuk melakukan pengembangan program. Disebut sebagai lingkungan karena melalui *software* inilah Arduino, ESP8266, Raspberry Pi pico diprogram untuk melakukan fungsi-fungsi yang dinamakan melalui sintak pemrograman. Di bawah ini merupakan tampilan Arduino IDE.[8]



Gambar 2. 1 Text Editor Arduino IDE

2.2.4 MYSQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL atau DBMS yang *multithread*, *multi-user*. Sebagai *database server*, MySQL dapat dikatakan lebih unggul dibandingkan *database server* lainnya dalam *query* data. Hal ini terbukti untuk *query* yang dilakukan oleh *single user*, kecepatan *query* MySQL bisa sepuluh kali

lebih cepat dari PostgreSQL dan lima kali lebih cepat dibandingkan *Interbase*. [9]



Gambar 2. 2 Logo Mysql

2.2.5 HTML

HyperText Markup Language (HTML) adalah bahasa markup utama yang digunakan browser untuk membuat dan menampilkan halaman web melalui struktur elemen-elemen tag. HTML digunakan untuk mengorganisir dan memformat dokumen di web, membentuk struktur dasar situs seperti heading, paragraf, gambar, dan tautan. [10]



Gambar 2. 3 Logo HTML

2.2.6 PHP

PHP (*Hypertext Preprocessor*) yaitu bahasa pemrograman berbasis web *server-side* yang bersifat *open source*. PHP merupakan script yang terintegrasi dengan HTML dan berada pada *server (server side HTML embedded scripting)*. PHP dirancang untuk dapat

bekerja sama dengan *database server* dan dibuat sedemikian rupa sehingga pembuatan dokumen HTML yang dapat mengakses database menjadi begitu mudah.[11]



Gambar 2. 4 Logo PHP

2.2.7 Codeigniter

Framework CodeIgniter adalah sebuah kerangka kerja bahasa pemrograman PHP *open source* berarsitektur MVC yang bermanfaat untuk membuat aplikasi web yang memiliki banyak fitur. Sejak tahun 2014, programmer dari EllisLab, yang sekarang menjadi proyek dari *British Columbia Institute of Technology*, adalah yang pertama kali membuat kerangka kerja ini. Dengan banyak fitur berguna.[9]



Gambar 2. 5 Logo Codeigniter

2.2.8 Bootstrap

Bootstrap adalah sebuah *tools* atau framework css yang membantu pengembang membuat situs web yang 20 menarik dan responsif. Tidak konsistensinya terhadap aplikasi tertentu membuat

pengembangan dan pemeliharaannya sulit. Bootstrap adalah CSS, tetapi dibuat dengan LESS, sebuah *pre-processor* yang memberi fleksibilitas lebih dari CSS biasa. Ini memberikan solusi yang konsisten untuk masalah *interface* yang biasa dihadapi pengembang.[9]



Gambar 2. 6 Logo Bootstrap

2.2.9 UML

Unified Modeling Language (UML) adalah metode pemodelan secara visual sebagai sarana untuk merancang dan atau membuat software berorientasi objek.[11]

Beberapa diagram yang digunakan di UML (*Unified Modeling Language*) :

1. Use Case Diagram

Fungsionalitas yang diharapkan dari sebuah sistem digambarkan dalam use case diagram. Sebuah use case menunjukkan interaksi antara aktor dan sistem, bukan bagaimana. Use case adalah tugas tertentu, seperti mengakses sistem, membuat daftar belanja, dll. Orang atau aktor adalah mesin atau manusia yang berinteraksi dengan sistem untuk melakukan tugas tertentu

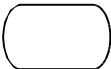
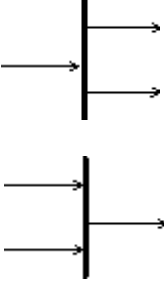
Tabel 2. 1 Metode Permodelan

Gambar	Nama	Keterangan
	Actor	menentukan peran yang dimainkan oleh pengguna saat berinteraksi dengan kasus.
	Association	apa yang menghubungkan satu objek ke objek lainnya.
	System	menentukan paket yang memiliki sistem terbatas.
	Use Case	Deskripsi dari urutan aksi aksi yang ditampilkan sistem yang dapat menghasilkan suatu hasil yang terukur bagi suatu aktor .

2. *Activity Diagram*

Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang. Bagaimana masing-masing alir berawal, decision yang mungkin terjadi, dan cara mereka berakhir. Selain itu, aktivitas diagram dapat menunjukkan proses paralel yang mungkin terjadi pada berbagai eksekusi. *Activity* diagram adalah state diagram khusus di mana sebagian besar state adalah tindakan dan sebagian besar transisi dipicu oleh selesainya state sebelumnya (proses internal). Akibatnya, aktivitas diagram tidak secara akurat menggambarkan perilaku internal sistem, serta interaksi antar subsistem, tetapi lebih banyak menggambarkan proses-proses dan jalur-jalur aktivitas dari tingkat atas secara keseluruhan. Satu atau lebih use case dapat digunakan untuk melakukan aktivitas. Use case menunjukkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas, sementara aktivitas menunjukkan proses yang berjalan. Decision digunakan untuk menggambarkan tindakan dalam situasi tertentu. Titik sinkronisasi, yang dapat berupa titik, garis horizontal atau vertical, digunakan untuk menunjukkan proses paralel (*fork* dan *join*).

Tabel 2. 2 *Activity* Diagram

Gambar	Nama	Keterangan
	<i>Activity</i>	memberikan gambaran tentang bagaimana masing-masing kelas antarmuka berinteraksi satu sama lain.
	<i>Action</i>	State sistem yang menunjukkan eksekusi tindakan
	<i>Initial Node</i>	Bagaimana objek dimulai atau dibentuk
	<i>Final Node</i>	Bagaimana objek dibentuk dan dihancurkan.
	<i>Fork/Join</i>	Digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu