

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Adanya penelitian dari Rizqi Cahyo M Putra (2024) yang berjudul “Perancangan Sistem Kontrol & Monitoring Berbasis IoT Pada Irigasi Pertanian Bawang Merah Menggunakan Pembangkit Listrik Hybrid”. Penelitian ini mengembangkan Sistem Monitoring dan Controlling berbasis Internet of Thing (IoT). Parameter yang nantinya di monitoring adalah tegangan, arus, daya yang dihasilkan pembangkit lalu untuk monitoring area pertanian adalah suhu dan kelembapan. Beban yang dikontrol adalah pompa air irigasi pertanian. Komponen pendukung sistem ini sangat banyak diantaranya, panel surya 50Wp, kincir angin, sensor tegangan, sensor arus INA219, sensor DS18B20, sensor YL-69, dan sensor irradiansi BH1750.[4]

Penelitian yang dilakukan Alexander Sinaga dan Aswardi (2020) dengan penelitiannya yang berjudul “Rancangan Alat Penyiram Dan Pemupukan Tanaman Otomatis Menggunakan RTC Dan Soil Moisture Sensor Berbasis Arduino”. Penelitian ini membuat sistem otomatis bekerja berdasarkan informasi beberapa sensor mengenai kondisi tanah pada tanaman dengan pengontrolan mikrokontroller arduino. Penyiraman tanaman dilakukan ketika sensor soil moisture membaca kelembaban tanah sesuai dengan set point yang telah ditentukan yaitu pada range >700 , dan penyiraman akan berhenti ketika tanah pada tanaman dalam keadaan basah

dan sesuai dengan set point yang telah ditentukan pada range >650. Dan untuk pemberian pupuk cair pada tanaman menggunakan RTC yang berfungsi sebagai penjadwalan pemberian pupuk cair yang terlebih dahulu dilakukan setting hari dan waktu pada program sesuai dengan set point yang telah dilakukan [5].

Pada penelitian lainnya yang dilakukan oleh Vina Septiana Windyarsari dan Pandu Azas Bagindo (2022) pada penelitian yang berjudul “Rancang Bangun Alat Penyiraman Dan Pemupukan Otomatis Dengan Sistem Monitoring Berbasis Internet Of Things”. Pada penelitian ini dapat melakukan penyiraman sesuai jadwal yang ditentukan, pompa akan aktif setiap pukul 08:00 pagi dan 16:00 sore selama 0 detik dan untuk proses pemupukan alat menggunakan motor servo yang akan membuka tempat pengeluaran pupuk selama 10 detik pada hari dan waktu yang sama [6].

Adapula penelitian yang dilakukan oleh Liyyin Putra Arif Wicaksana, Alexander Setiawan, dan Resmana Lim (2022) yang berjudul “Aplikasi Monitoring Pada Tanaman Aglaonema Menggunakan IoT”. Pada penelitian tersebut ph tanah untuk kelembapan tanaman aglaonema harus antara 6-7, hasil pengujian SensorBH1750 memiliki tingkat akurasi 97.6% dan pengujian sensor PH probe memiliki tingkat akurasi 99.5% [7].

2.2 Landasan Teori

2.2.1 Bawang Merah

Bawang merah pada Gambar dibawah adalah salah satu sayuran unggulan yang telah lama dibudidayakan secara *intensif* oleh petani. Sayuran ini menjadi sumber pendapatan dan lapangan kerja yang signifikan, memberikan kontribusi besar bagi perkembangan ekonomi daerah. Karena nilai ekonominya yang tinggi, budidaya bawang merah kini tersebar hampir di seluruh Indonesia. Meskipun minat petani terhadap bawang merah sangat tinggi, mereka masih menghadapi berbagai kendala dalam proses budidayanya, baik dari segi teknis maupun ekonomis.[8].



Gambar 2. 1 Bawang Merah

2.2.2 Internet Of Things

Internet of things adalah sebuah data dapat dikirim antar objek di jaringan tanpa memerlukan komunikasi komputer atau manusia ke komputer. Perkembangan IoT dapat ditelusuri kembali ke titik dimana teknologi *internet*, *nirkabel*, dan mikroelektromekanis. IoT dan komunikasi mesin-ke-mesin saat ini paling erat terkait dalam industri

industri dan minyak dan gas. Produk berkemampuan komunikasi M2M sering disebut sebagai sistem cerdas [9].

2.2.3 Website

Serangkaian halaman yang sering menjadi bagian dari nama *domain* atau *subdomain* di *WWW* disebut sebagai situs Web. Halaman web adalah sebuah *protocol* yang memiliki dokumen dalam format *HTML* yang hampir selalu dapat diakses melalui *HTTP* yang mentransmisikan data dari server situs web lalu ditampilkan ke *browser* web pengguna, baik statis maupun dinamis [10].

2.2.4 Sistem Monitoring

Proses mengumpulkan dan mengevaluasi data tentang suatu kegiatan atau program berdasarkan indikator yang diidentifikasi secara sistematis dan berkesinambungan sehingga tindakan perbaikan dapat dilakukan untuk meningkatkan kegiatan berikutnya [11].

2.2.5 XAMPP

XAMPP adalah server web lokal yang memungkinkan untuk membangun dan menguji website tanpa koneksi internet. Dengan *XAMPP*, dapat mengelola *database* di komputer Anda (*localhost*) tanpa perlu mengakses *internet*. Hal ini menjadi solusi ideal ketika koneksi *internet* terputus dan tetap ingin bekerja pada *website*.

2.2.6 Visual Studio Code

Visual Studio Visual Studio Code (VS Code) adalah *editor source code multiplatform* dari *Microsoft* yang mendukung *Windows*, *macOS*, dan *Linux*. *Editor* ini ideal untuk *extension* yang luas seperti *JavaScript*, *TypeScript*, dan *Node.js*, yang memiliki berbagai bahasa pemrograman dan *runtime*, seperti *C++*, *C#*, *Java*, *Python*, *PHP*, *Go*, dan *.NET*.

Fitur Utama VS Code:

- *IntelliSense*: *Fitur* cerdas yang memberikan saran kontekstual saat mengetik untuk membantu *developer* menulis *coding* lebih cepat dan minim resiko *error*.
- *Run and Debug*: Jalankan dan *debug* kode secara langsung untuk berbagai bahasa pemrograman. Atur *breakpoint*, periksa *variabel*, dan telusuri kode untuk mengidentifikasi dan memperbaiki masalah.
- *Built-in Git*: Kelola *version control* secara langsung di dalam *editor*. Atur dan ubah *repositori* tanpa perlu *Git client* terpisah.
- *Extensions*: Ribuan *extension* tersedia untuk menambahkan dukungan bahasa pemrograman, *framework*, dan alat lainnya, memungkinkan *developer* untuk menyesuaikan *editor* dengan kebutuhan spesifiknya.

2.2.7 Arduino IDE

Sebuah *software* yang bagaikan gerbang untuk membuka dunia pemrograman bagi para pemula dan ahli elektronik. Fungsinya sebagai media untuk membuat *sketch* pemrograman, mengedit, dan mengupload kode ke *board* Arduino yang ingin diprogram. Dengan Arduino IDE, Anda dapat menciptakan program unik untuk mengendalikan berbagai perangkat elektronik.

Software ini dilengkapi dengan *library C/C++ (wiring)* yang mudah untuk beroperasi *input/output*. Hal ini memungkinkan untuk berkreasi dengan berbagai sensor-sensor, aktuator, dan komponen elektronik lainnya dengan cukup mudah.

2.2.8 Database

Basis data (*database*) adalah tempat penyimpanan data. Kata "basis" mengacu pada gudang atau tempat penyimpanan, sedangkan "data" adalah representasi dari berbagai objek dalam dunia nyata, Data ini dapat direkam dalam berbagai format, termasuk angka, huruf, teks, gambar, simbol, bunyi, dan juga seperti manusia, hewan, benda, konsep, dan peristiwa. atau kombinasi dari format-format lainnya.

2.2.9 MySql

MySql Dengan kemampuan multi-pengguna, multi-utas, dan skalabilitasnya yang tinggi, *MySQL* menjadi pilihan tepat untuk mengelola *database* dengan volume data besar dan kompleks.

Kemampuannya yang *fleksibel* dan komunitasnya yang besar menjadikan *MySQL* sebagai salah satu *DBMS* paling populer di dunia.

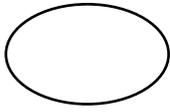
2.2.10 UML (*Unified Modeling Language*)

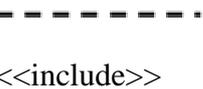
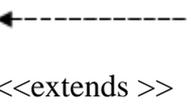
Salah satu *tools* yang sering kita gunakan untuk membuat dokumentasi piranti perangkat lunak yaitu *Unified Modeling Language* atau yang sering disebut *UML*. *UML* merupakan sebuah perangkat lunak dengan berbagai jenis dan fungsi sebagai berikut.

1. *Use Case Diagram*

Diagram dibawah ini menunjukkan kumpulan *use case* dan aktor-aktor, dan juga sangat penting dalam mengorganisir dan memodelkan perilaku sistem yang dibutuhkan serta diharapkan oleh manusia.

Tabel 2. 1 *Use Case Diagram*

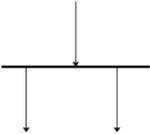
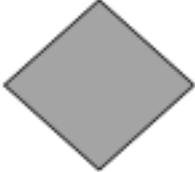
| No | Simbol | Keterangan |
|----|-------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1. |  | <i>Use case</i> merupakan deskripsi fungsional yang telah disediakan oleh sistem sebagai entitas yang menghasilkan hasil yang terukur untuk suatu <i>actor</i> . |
| 2. |  | <i>Actor</i> merupakan himpunan peran untuk berinteraksi dengan <i>usecase</i> |

| | | |
|----|-------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------|
| 3. |  | <i>Association</i> adalah yang menghubungkan objek/actor dari satu dengan objek yang lainnya |
| 4. |  | <i>Include</i> merupakan gambaran jika <i>usecase</i> dipanggil oleh <i>usecase</i> lain |
| 5. |  | <i>Dependecy</i> merupakan garis panah yang menunjukan jika <i>actor</i> berinteraksi secara pasif |
| 6. |  | <i>Extend</i> merupakan gambar jika memperluas <i>usecase</i> target. |

2. *Activity Diagram*

Diagram dibawah ini adalah jenis khusus dari *diagram state* yang menampilkan aliran aktivitas dari satu aktivitas ke aktivitas lainnya dalam sebuah sistem yang dibuat. Diagram ini juga memiliki peran penting untuk memodelkan fungsi-fungsi dalam suatu sistem dan menekankan aliran kendali dari objek ke antar objek lain.

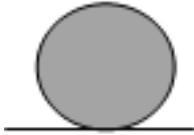
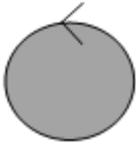
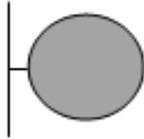
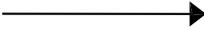
Tabel 2. 2 Activity Diagram

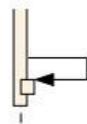
| No | Simbol | Keterangan |
|----|-------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------|
| 1. |  | <i>End point</i> atau <i>final Node</i> merupakan gambaran akhir dari suatu aktivitas |
| 2. |  | <i>Start Point</i> merupakan awal dari suatu aktivitas yang peletakannya pada pojok kiri atas |
| 3. |  | <i>Fork</i> atau <i>join</i> digunakan untuk memarallelkan suatu kegiatan atau penggabungan 2 kegiatan parallel menjadi satu |
| 4. |  | <i>Activity</i> merupakan gambaran dari suatu proses |
| 5. |  | <i>Decision</i> merupakan pilihan pengambilan suatu keputusan <i>false</i> or <i>true</i> . |

3. *Sequence Diagram*

Diagram sequence dibawah ini adalah diagram yang berinteraksi menyoroti pengiriman pesan pada waktu tertentu di sistem. Diagram ini bersifat dinamis.

Tabel 2. 3 *Sequence Diagram*

| No | Simbol | Keterangan |
|----|-------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1. |  | <i>Entity Class</i> adalah yang membentuk deskripsi awal sistem dan berisi kumpulan kelas dalam bentuk entitas yang mendasari untuk membuat database |
| 2. |  | <i>ControlClass</i> merupakan gambaran penghubung antara <i>Boundaryclass</i> dengan suatu table |
| 3. |  | <i>Boundary Class</i> merupakan gambaran dari penggambaran table |
| 4. |  | Pesen atau <i>message</i> menunjukkan pengiriman pesen antar <i>class</i> |

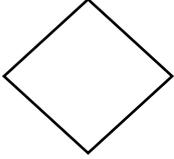
| | | |
|----|-----------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------|
| 5. |  | <i>Self message</i> memberikan suatu pesan ke objek itu sendiri |
| 6. |  | <i>Activation</i> menggambarkan suatu objek yang melakukan sebuah aksi/eksekusi operasi |
| 7. |  | <i>Lifeline</i> adalah sebuah garis titik-titik yang terhubung, disepanjang garis lifeline yang memiliki aktivitas ke objek |

4. *Class Diagram*

Diagram class dibawah ini bersifat statis. Yang memperlihatkan relasi, kolaborasi, antar muka, dan himpunan kelas.

Tabel 2. 4. *Class Diagram*

| No | Simbol | Keterangan |
|----|-------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------|
| 1. |  | <i>Generallization</i> , merupakan dimana objek <i>descendent</i> membagikan perilaku dan struktur data objek induknya |
| 2. |  | <i>Class</i> , adalah kumpulan objek yang saling berbagi. |

| | | |
|----|-------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------|
| 3. |  | <p><i>Nary Association</i>, digunakan untuk asosiasi terhindar dengan objek lainnya.</p> |
| 4. |  | <p><i>Collaboration</i> adalah deretan titik-titik urutan aksi yang menampilkan suatu sistem tertentu yang memiliki konsekuensi terukur bagi actor</p> |
| 5. |  | <p>Merupakan sebuah tanda panah titik-titik objek yang valid</p> |
| 6. |  | <p>Merupakan garis panah yang menunjukkan jika actor berinteraksi secara pasif</p> |
| 7. |  | <p><i>Association</i> merupakan garis penghubung antar objek</p> |