# BAB II RELATED WORK

* 1. **State Of The Art**

|  |
| --- |
| **C:\Users\Asus\Downloads\Purple Blue Simple Fishbone Diagram Chart (1).jpg** |
| Gambar 2.1 : Fishbone |

* 1. **Deskripsi Topik yang Sama**

Hidroponik adalah budidaya tanaman dengan memanfaatkan air dan tidak menggunakan tanah sebagai media tanam, tetapi lebih kepada penekanan kebutuhan nutrisi bagi tanaman [20]. Teknik NFT adalah teknik yang menggunakan larutan nutrisi untuk menguras di daerah akar. Solusi nutrisi sangat penting untuk didefinisikan keberhasilan budidaya hidroponik[21]. IOT dikenal sebagai salah satu bidang teknologi masa depan yang paling penting dan mendapat perhatian luas dari berbagai industri[22]. Suhu adalah salah satu faktor utama yang mempengaruhi pertumbuhan tanaman[23].Suhu air dianggap sebagai salah satu parameter penting yang mempengaruhi tingkat pertumbuhan dan pengembangan tanaman serta distribusi di dalam lingkungan kolam[10].Pertumbuhan setiap harinya dihitung dari pengukuran suhu udara dan merupakan kondisi cuaca yang sering digunakan untuk menggambarkan waktu proses biologis[24]. Respon tanaman terhadap kenaikan suhu udara tergantung pada nilai suhu [11].Suhu merupakan salah satu faktor yang secara signifikan menentukan pertumbuhan dan perkembangan tanaman[12]. Suhu rendah memungkinkan tanaman memiliki kendala dalam pertumbuhan[15]. Suhu yang paling ideal untuk fase tanaman contohnya kecambah berkisar antara 25◦ -30◦C, sedangkan suhu ideal untuk pembentukan warna buah berkisar antara 24◦ -28◦C[13]. Kontrol suhu adalah variabel yang menarik dalam pertumbuhan tanaman air yang sehat[14]. Kontrol suhu adalah pemandangan yang sangat umum dalam kendali industri[25].Pengatur suhu yang mempertahankan suhu air pada nilai yang telah ditentukan dilekatkan air yang merupakan wadah[26]. Dalam teknologi kontrol suhu tradisional, pemanasan dan pendinginan seringkali bersifat diskrit[27]. Kontrol suhu memiliki peran yang sangat penting dalam aplikasi industri / rumah dan sulit diimplementasikan melalui teknik kontrol biasa[28].

* 1. **Deskripsi Metode yang Sama**

Saat ini Fuzzy Logic banyak digunakan di berbagai bidang, salah satunya adalah di bidang control [29]. Prinsip logika fuzzy dibangun di atas seperangkat aturan bahasa manusia yang dipasok pengguna [30]. Logika fuzzy telah menjadi sangat populer, terutama karena proses kontrol logika fuzzy dapat mewujudkan strategi pengendalian manusia[31]. Logika fuzzy dapat menganalisa data ketidakpastian sehingga membuatnya fleksibel[32]. Logika fuzzy telah ditemukan sebagai praktik yang paling sesuai karena dapat menawarkan semua keadaan transisi antara ’0’ dan ’1’[33]. Kontroler fuzzy dapat mengendalikan model proses nonlinier dan model proses delay-time secara signifikan lebih baik daripada pengendali klasik[34]. Sistem fuzzy didasarkan pada pengetahuan dan pengalaman manusia, dengan menggunakan seperangkat aturan bentuk IF-THEN untuk menentukan output dari pengontrol yang diberikan satu set input[16]. Dalam logika fuzzy aturan bisa dibingkai sesuai kondisi[35]. Kontrol Fuzzy Logic adalah strategi kerangka kerja pengendalian pemikiran kritis yang digunakan[36]. Strategi pengendalian berdasarkan logika fuzzy yang serupa dengan penalaran manusia mentolerir ketidakpastian dan ketidaktepatan[17]. Fuzzy Logic adalah teknik pengontrol yang lebih berguna untuk menghindari variasi frekuensi dalam sistem hybrid[37]. Teknik Logika Fuzzy merupakan subset dari kecerdasan komputasi[38]. Contoh Aturan dasar dalam mengatur suhu ruangan hanya terdiri dari empat aturan yang mampu menutupi masalah kontrol set point[39]. Logika fuzzy membantu untuk sampai pada kesimpulan yang berbeda, tergantung pada input sensor informasi yaitu pengalaman pengetahuan para ahli[40]. Logika fuzzy adalah pemecahan masalah metodologi sistem kontrol yang digunakan dalam berbagai aplikasi karena kemampuan meniru logika kontrol manusia dan kemudahan modifikasi[18].

* 1. **Landasan Teori**
     1. **Pengertian Implementasi**

Implementasi secara umum diartikan sebagai suatu tindakan atau pelaksana rencana yang telah disusun secara cermat dan rinci (matang). Sedangkan Whitten, Bentle, & Barlow (1993) mengatakan bahwa “Implementasi adalah suatu proses untuk menempatkan dan menerapkan informasi dalam operasi”.

* + 1. **Pengertian Mempertahankan**

Mempertahankan disebut sebagai mengusahakan sesuatu supaya tetap tidak berubah dari keadaan semula.

* + 1. **Hidroponik**

Hidroponik merupakan teknik bertanam tanpa menggunakan media tanah. Teknik ini mampu meningkatkan hasil tanaman per satuan luas sampai lebih dari sepuluh kali, bila dibandingkan dengan teknik pertanian konvensional[4].

* + 1. **DSRM**

*Design Science Research Methodology* yang dikemukakan oleh Ken Peffers, Tuure Tuunanen, Marcus A.Rothenberger, dan Samir Chatterjee pada jurnal berjudul “*A Design Science Research Methodology for Information Systems Research*” yang terdiri dari enam tahap yaitu identifikasi masalah, penetapan tujuan, perancangan, studi kasus, evaluasi dan komunikasi. Berikut gambar alur tahapan penelitian yang dilakukan : [11]

|  |
| --- |
|  |
| *Gambar 5.1 Diagram Alur DSRM* |

Tahapan - tahapan diagram alur metodologi penelitian DSRM antara lain :

* 1. Identifikasi Masalah dan Motivasi
  2. Penentuan Tujuan dari Solusi Penelitian
  3. Perancangan dan Pengembangan Solusi
  4. Demonstrasi
  5. Evaluasi / Pengujiam
  6. Komunikasi
     1. **UML**
        1. **Pengertian UML**

Menurut Nugroho (2010:6), UML (*Unified Modeling Language*) adalah ‘bahasa’ pemodelan untuk sistem atau perangkat lunak yang berparadigma ‘berorientasi objek”. Pemodelan (*modeling*) sesungguhnya digunakan untuk penyederhanaan permasalahan-permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipelajari dan dipahami.

* + - 1. **Model UML (Unified Modeling Language)**

Menurut Widodo (2011:10), “Beberapa literature menyebutkan bahwa UML menyediakan sembilan jenis diagram, yang lain menyebutkan delapan karena ada beberapa diagram yang digabung, misanya diagram komunikasi, diagram urutan dan diagram pewaktuan digabung menjadi diagram interaksi”. Namun demikian model-model itu dapat dikelompokkan berdasarkan sifatnya yaitu statis atau dinamis. Jenis diagram itu antara lain:

1. Diagram kelas (*Class Diagram*)

Bersifat statis, Diagram ini memperlihatkan himpunan kelas-kelas, antarmuka-antarmuka, kolaborasi-kolaborasi, serta relasi-relasi. Diagram ini umum dijumpai pada pemodelan sistem berorientasi objek. Meskipun bersifat statis, sering pula diagram kelas memuat kelas-kelas aktif.

1. Diagram paket (*Package Diagram*)

Bersifat statis. Diagram ini memperlihatkan kumpulan kelas-kelas, merupakan bagian dari diagram komponen.

1. Diagram *use-case* (*Usecase Diagram*)

Bersifat statis. Diagram ini memperlihatkan himpunan u*se-case* dan aktor-aktor (suatu jenis khusus dari kelas). Diagram ini terutama sangat penting untuk mengorganisasi dan memodelkan perilaku suatu sistem yang dibutuhkan serta diharapkan pengguna.

1. Diagram interaksi dan sequence (*Sequence Diagram*)

Bersifat dinamis. Diagram urutan adalah iterasiksi yang menekankan pada pengiriman pesan dalam suatu waktu tertentu.

1. Diagram komunikasi (*Communication Diagram*)

Bersifat dinamis. Diagram sebagai pengganti diagram kolaborasi UML yang menekankan organisasi struktural dari objek - objek yang menerima serta mengirim pesan.

1. Diagram statechart (*Statechart Diagram*)

Bersifat dinamis. Diagram status memperlihatkan keadaan-keadaan pada sistem, memuat status (*state*), transisi, kejadian serta aktivitas.

1. Diagram aktivitas (*Activity Diagram*)

Bersifat dinamis. Diagram aktivitas adalah tipe khusus dari diagram status yang memperlihatkan aliran dari suatu suatu aktivitas ke aktivitas lainnya dalam suatu sistem. Diagram ini terutama penting dalam pemodelan fungsi-fungsi suatu sistem dan memberi tekanan pada aliran kendali antar objek.

1. Diagram komponen (*Component Diagram*)

Bersifat statis. Diagram komponen ini memperlihatkan organisasi serta kebergantungan sistem/perangkat lunak pada komponen-komponen yang telah ada sebelumnya.

1. Diagram deployment (*deployment diagram*)

Bersifat statis. Diagram inimemperlihatkan konfigurasi saat aplikasi dijalankan (*run-time*). Memuat simpul-simpul beserta komponen-komponen yang di dalamnya. Kesembilan diagram ini tidak mutlak harus digunakan dalam pengembangan perangkat lunak, semuanya dibuat sesuai kebutuhan. Pada UML dimungkinkan kita menggunakan diagram-diagram lainnya misalnya data *flow diagram, entity relationship diagram*, dan sebagainya.