

## BAB II LANDASAN TEORI

### 2.1 Tinjauan Pustaka

Pada penelitian ini, penulis membutuhkan *Literature* yang didapat dari penelitian sebelumnya dan bertujuan untuk dapat mendukung penelitian penulis. Adapun penelitian terdahulu yang relevan dengan penelitian ini adalah :

**Tabel 2.1** Penelitian Terdahulu

No. Litertur	Penulis	Tahun	Judul
<i>Literature 1</i>	Paridah Ardianto Pranata Dedi Setiawan	2020	Sistem Monitoring dan Pemberian TDS Pada Tanaman Aeroponik Menggunakan <i>Internet of Things</i> Berbasis <i>NodeMCU</i> .
<i>Literature 2</i>	Charisma Aulia Jamhari Wahyu Kunto Wibowo Aulia Rahma Annisa Teuku Muhammad Roffi	2020	<i>Design and Implementation of IoT System for Aeroponic Chamer Temperature Monitoring.</i>
<i>Literature 3</i>	Lenyn Lucero Dora Lucero Elianna Ormeno-Meija Gerardo Collaguazo	2020	<i>Automated Aeroponics Vegetable Growing System Case Study Lettuce.</i>
<i>Literature 4</i>	Mathawee Fuangthong Part Pramokchon	2018	<i>Automatic Control of Elevation Conductivity and PH Using Fuzzy Logic for Hydroponics System.</i>
<i>Literature 5</i>	Gurmu M. Debele Xiao Qian	2020	<i>Automatic Room Temperature Control System Using Arduino Uno R3 and DHT11 Sensor.</i>

### **2.1.1 Literature 1**

Penelitian ini dilakukan oleh (Studi dan Tinggi, 2019), penelitian ini berjudul sistem monitoring dan pemberian *TDS* pada tanaman aeroponik menggunakan *Internet of Things (IoT)* berbasis *NodeMCU*. Penelitian ini membangun sistem aeroponik untuk memonitoring dan pemberian nutrisi pada tanaman kentang menggunakan aplikasi *blynk*. Untuk tanaman kentang nilai ppm yang baik adalah 1000-1750 ppm. Apabila sensor mendeteksi lebih dari 1750 ppm maka dapat dikategorikan sebagai nilai *TDS* yang tinggi dan sistem akan mengirimkan notifikasi pada lcd. Sedangkan ppm mendeteksi kurang dari 1000 ppm maka dapat dikategorikan sebagai nilai *TDS* yang rendah dan sistem juga akan mengirimkan notifikasi pada lcd. Proses pengaktifan pompa dapat dilakukan dengan menekan tombol *button* pada aplikasi *blynk*.

### **2.1.2 Literature 2**

Penelitian ini dilakukan oleh (Jamhari *et al.*, 2020), penelitian ini berjudul perancangan dan implementasi sistem IoT untuk pemantauan suhu ruang aeroponik. Penelitian ini membangun sistem aeroponik skala *laboratorium* yang menggunakan *Internet of Things*. Penelitian ini dibuat dikarenakan masyarakat yang mulai memperhatikan kualitas produk untuk dikonsumsi dengan menggunakan urban farming. Salah satu urban farming yang sedang naik daun adalah metode aeroponik. Penelitian ini menyebutkan bahwa suhu untuk ruang aeroponik ini berkisar 25 – 30 °C dengan tingkat kelembaban diatas 60%. Sedangkan untuk paparan sinar matahari berkisar 29 - 32,9 °C. Sistem yang dibuat pada penelitian ini mampu mendinginkan suhu

ruang akar hingga rata-rata 28,8 °C dan sudah termasuk ideal untuk pertumbuhan tanaman sayuran.

### **2.1.3 Literature 3**

Penelitian yang dilakukan oleh (Lucero *et al.*, 2020), penelitian ini berjudul studi kasus sistem penanaman sayuran aeroponik otomatis. Pada penelitian ini disebutkan untuk meningkatkan produktivitas dan mengurangi kerusakan lingkungan yang disebabkan oleh petani *konvensional* maka pada penelitian ini menjelaskan penerapan sistem budidaya selada daun hijau. Penelitian ini difokuskan untuk mengontrol parameter suhu, kelembaban dan waktu irigasi dengan penyemprotannya berbasis *Internet of Things* sehingga pengguna dapat melakukan pemantauan dari jarak jauh.

### **2.1.4 Literature 4**

Penelitian ini dilakukan oleh (Fuangthong dan Pramokchon, 2018), penelitian ini berjudul control otomatis konduktivitas listrik dan ph menggunakan logika *fuzzy* untuk sistem hidroponik. Pada penelitian ini dilakukan penerapan menggunakan logika *fuzzy* untuk mengendalikan nilai EC dan PH pada penanaman hidroponik secara otomatis. Namun, untuk mengontrol kedua parameter ini membutuhkan keterampilan dan pengetahuan untuk menghindari pemborosan pada sumber daya seperti tenaga kerja dan nutrisi.

### **2.1.5 Literature 5**

Penelitian yang dilakukan oleh (Debele dan Qian, 2020), penelitian ini berjudul sistem control suhu ruangan otomatis menggunakan sensor Arduino

Uno R3 dan DHT11. Penelitian ini pengguna dapat mengatur rentang suhu referensi minimum dan maksimum dari *keypad*. Dengan menggunakan sensor DHT11 peneliti menggunakan sensor tersebut untuk mendeteksi suhu dan kelembaban yang nantinya juga ditampilkan pada *Liquid Crystal Display* (LCD). Jika suhu ruangan terdeteksi melampaui batas maksimum maka otomatis kipas akan menyala. Kecepatan kipas akan dikontrol dengan teknik *modulasi* lebar pulsa (PWM) berdasarkan perbedaan suhu antara pembacaan sensor dan ambang batas maksimum. Sehingga semakin besar perbedaan suhu, semakin besar siklus kerja kipas, dan cepat kecepataannya.

## 2.2 Komponen Pendukung

### 2.2.1 Kabel *Jumper*

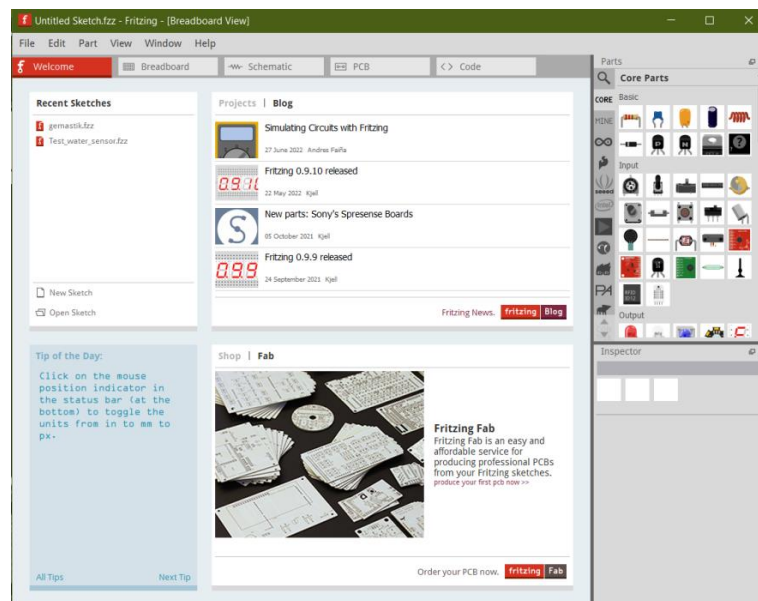
Kompnen penting dalam pembangunan rangkaian ini adalah kabel *jumper NodeMCU ESP32* dan Arduino UNO R3. Kabel *jumper* adalah kabel listrik dengan pin konektor di setiap ujungnya dan memungkinkan untuk mengkoneksi dua komponen yang terkait dengan *NodeMCU ESP32* tanpa penyolderan. Ada dua jenis konektor di ujung kabel, yaitu konektor *male* dan *female*(Theodorus S Kalengkongan *et al.*, 2018). Di bawah ini adalah gambar kabel *jumper*.



**Gambar 2.1** Kabel *Jumper*  
Sumber : (Dani, 2018)

### 2.2.2 Fritzing

*Fritzing* adalah perangkat lunak yang beradaptasi dengan *e-learning*. *Software Fritzing* adalah *software* yang bisa digunakan oleh pecinta elektronik. Perangkat lunak *Fritzing* dapat digunakan pada sistem *Windows* atau *Linux*(Prabowo *et al.*, 2020). Dalam penelitian ini, *fritzing* digunakan untuk merancang alat penanaman kentang.



**Gambar 2.2** *Fritzing*

### 2.2.3 Hardware

*Hardware* adalah istilah untuk menyebut perangkat keras yang erat dengan komponen baik komponen komputer, laptop atau alat-alat yang digunakan dalam pembuatan alat di penelitian ini. Ada beberapa jenis perangkat keras salah satunya adalah *NodeMCU*, masing-masing dari *NodeMCU* memiliki kelebihan dan kekurangannya sendiri. *NodeMCU ESP32* yang digunakan harus sesuai dengan desain yang disyaratkan. Penambahan fungsi pada setiap *board* inilah yang membuat setiap *NodeMCU*

berbeda(Mahesa *et al.*, 2019). Dalam penelitian ini, *NodeMCU* akan digunakan sebagai mikrokontroler.

#### 2.2.4 Software

*Driver IDE* adalah *driver* perangkat lunak masih perangkat lunak lain yang sangat berguna. *Integrated Development Environment (IDE)* program komputer khusus untuk membuat desain atau program sketsa untuk papan *Arduino* dan *NodeMCU*(Samsugi *et al.*, 2020). *Arduino IDE* terdiri dari :

1. Editor Program

*Windows* untuk menulis dan mengedit program dalam bahasa pemrosesan.

2. *Compiler*

Fungsi yang mengkompilasi garis besar tanpa mengunggah kepapan dapat digunakan untuk memeriksa kesalahan sintaks garis besar. Modul yang mengubah kode program menjadi biner setelah mikrokontroler yang sama tidak akan dapat memahami bahasa pemrosesan.

3. *Uploader*

Untuk mengunggah *sketsa* yang dikompilasi kepapan target. Pesan kesalahan akan muncul di layer *log*.

4. *New Sketch*

Untuk membuka *window* dan membuat *sketch* baru.

5. *Save Sketch*

Untuk menyimpan *sketch* yang sudah dibuat.

#### 6. *Open Sketch*

Untuk membuka sketsa yang dibuat sebelumnya. *Sketsa* yang dibuat dengan *Arduino IDE* akan disimpan dalam folder pilihan kita.

#### 7. *Serial Monitor*

Untuk membuka antarmuka untuk komunikasi *serial*, kita akan membahas lebih detail di bagian selanjutnya.

#### 8. Keterangan Aplikasi

Pesan aplikasi akan muncul di beberapa bagian seperti kompilasi dan unggah lengkap saat kami mengkompilasi dan mengunggah *sketsa* ke papan Arduino.

#### 9. *Konsol log*

Ini adalah notifikasi yang digunakan aplikasi dan notifikasi *sketsa* akan muncul di bagian ini. Misalnya, ketika aplikasi mengkompilasi atau ketika ada kesalahan pada *sketsa* yang sedang kita kerjakan, kesalahan dan informasi baris akan diisi di bagian ini.

#### 10. Baris *Sketch*

Bagian yang mewakili posisi saat ini dari garis *cursor* dalam *sketsa*.

#### 11. Informasi *Board* dan *port*



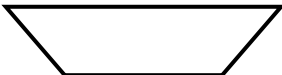

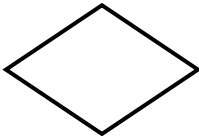
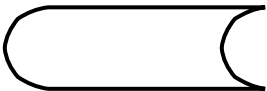
Bagian ini menyediakan port yang digunakan oleh board Arduino.

### **2.2.5 *Flowchart***



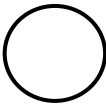
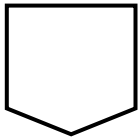

*Flowchart* memiliki fungsi yang ditentukan atau referensi yang memungkinkan penulis untuk melakukan urutan langkah demi langkah dari proses yang akan dieksekusi oleh aplikasi dan mikrokontroler yang akan

dieksekusi nanti. *Flowchart* sangat berpengaruh terhadap berjalan atau tidaknya sistem. Langkah ini merupakan pondasi awal sebelum sebuah sistem atau alat dikdanung. Jika pembuatan atau pembuatan skema tidak baik, maka dapat dipastikan bahwa sistem atau alat yang dibuat tidak baik atau sempurna. Oleh karenanya, sangat penting bagi kita untuk mengikuti proses dasar ini, sehingga sistem atau alat yang dihasilkan akan jauh lebih baik(Khesya, 2021).

**Tabel 2.2** Simbol *Flowchart*

NAMA	SIMBOL	KETERANGAN
Terminal		Simbol yang berfungsi untuk menunjukkan proses awal atau akhir suatu proses.
Porses		Simbol yang berfungsi untuk menunjukkan proses suatu sistem
Proses		Simbol proses yang dilakukan secara manual.
Proses		Simbol yang digunakan oleh manusia dan komputer seperti memasukan data kekomputer.
Decision		Simbol pengambilan keputusan bagaimana alur dalam <i>flowchart</i> berjalan selanjutnya berdasarkan pernyataan.
Stored Data		Simbol informasi yang disimpan kedalam media penyimpanan umum.



Databased		Untuk basis data atau <i>database</i> .
Predefined Process		Untuk proses yang telah kita jelaskan lebih rinci di dalam <i>flowchart</i> tersendiri.
Koneksi		Pengganti garis penghubung.
Penghubung		Koneksi yang dipakai pada halaman lain, sebagai pengganti garis penghubung.
Garis		Garis penghubung aliran algoritma.

### 2.2.6 NodeMCU ESP32

*NodeMCU* ESP32 merupakan salah satu mikrokontroler yang masuk kedalam keluarga *NodeMCU*. *NodeMCU* bersifat *opensource* dari sebuah *platform IoT* yang terdiri dari perangkat keras seperti *system on chip* ESP32. *NodeMCU* ESP32 ini memiliki WiFi mode g dan 2.4 GHz dan koneksi nirkabel BT. Selain itu mikrokontroler ini terintegrasi SRAM 512 kB dan *memory* sebesar 4MB, 2x DAC, 15x ADC, 1x SPI, 1x I2C, 2x UART. PWM diaktifkan di semua pin digital (Imran dan Rasul, 2020).



**Gambar 2.3** *NodeMCU* ESP32

Sumber : (Dani, 2018)

### 2.2.7 Real Time Clock / RTC

RTC adalah sebuah modul yang berfungsi untuk menjalankan fungsi waktu dan tanggal atau kalender secara *realtime* berbasis DS3231 dengan dilengkapi *battery* untuk tetap menjaga agar RTC masih tetap aktif (Selamet *et al.*, 2022). Berikut adalah fitur- fitur yang terdapat pada RTC DS3231 antara lain:

- a. *Real Time Clock Counts Seconds, Minutes, Hours, Date of the Month, Month, Day of The Week, dan Year with Leap-Year.*
- b. *56-Byte, Battery-Backed, General-Purpose RAM with Unlimited Writes*
- c. *I2C Serial Interface*
- d. *Programmable Square-Wave Output Signal*
- e. *Automatic Power-Fail Detect dan Switch Circuitry*
- f. *Consumes Less Than 500nA in Battery-Backup Mode with Oscillator Running*

Pada modul ini terdapat 6 pin utama, yaitu sebagai berikut :

- a. *SCL : Serial Clock Input* untuk komunikasi data I2C
- b. *SDA : Serial Data Input / Output* untuk komunikasi data I2C
- c. *VCC : Input Supply 5V*
- d. *GND : Input Supply Ground*
- e. *SQW : Square Wave Output*
- f. *32K : 32 Oscillator Output*

Dibawah ini adalah gambar *RTC DS3231* yang akan digunakan oleh penulis.

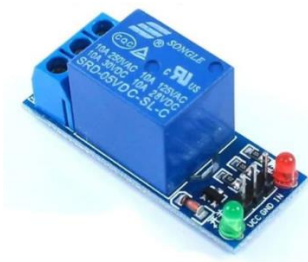


**Gambar 2.4** *RTC DS3231*

Sumber : (Anon, 2017)

### 2.2.8 Relay

*Relay* memiliki prinsip kerja yang sama dengan saklar (*switch*). *Relay* adalah komponen *Elektromechanical* (Elektromekanikal) yang terdiri dari dua bagian utama yaitu Elektromagnet (*Coil*) dan Mekanikal (seperangkat kotak *switch*)(Nasution *et al.*, 2019).



**Gambar 2.5** *Relay*

Sumber : (webstudi, 2020)

### 2.2.9 Sensor DHT22

DHT22 merupakan sensor yang dapat mendeteksi suhu dan kelembaban lingkungan atau ruangan. Sensor DHT22 ini memiliki pengaturan yang sangat akurat dengan bacaan suhu ruang dengan nilai yang tersimpan yang ada di memori OTP terpadu. Selain itu sensor ini mampu mendistribusikan sinyal keluaran via kabel dengan panjang hingga mencapai 20 meter(Saputra *et al.*, 2020).



**Gambar 2.6** DHT22

Sumber : (Abdurahman Rasyid, S.Pd., 2019)

#### 2.2.10 Sensor PH 4502C

Sensor ini berfungsi untuk mendeteksi tingkat ph air yang dimana memiliki *output* berupa tegangan analog. Sensor ini memiliki tegangan analog berada berkisar 0 – 3V DC dengan *input power supply* 3.3 – 5.5 V DC (Mufida *et al.*, 2020).



**Gambar 2.7** Sensor PH 4502C

Sumber : (Dani, 2018)

#### 2.2.11 Sensor TDS Meter

Sensor TDS Meter merupakan singkatan dari *Total Dissolved Solids*. Sensor ini merupakan alat yang digunakan untuk mengukur partikel yang ada pada air yang tidak bisa dilihat dengan mata telanjang. Pada setiap air minum pasti mengandung suatu partikel yang dapat larut dan tidak tampak oleh mata manusia (Dewa dan Sasmoko, 2016).



**Gambar 2.8** Sensor *TDS Meter*  
Sumber : (Aldy Razor, 2022)

### 2.2.12 Sensor *Water Level*

Sensor water level merupakan suatu alat yang digunakan untuk mengukur tingkat ketinggian air. Sensor water level alat yang mampu digunakan untuk memberikan *signal* kepada alarm / *automation panel* bahwa permukaan air telah mencapai *level* tertentu. Sensor akan memberikan *signal dry contact* (NO/NC) ke panel pendeteksi *level* ketinggian air dengan membaca nilai tegangan yang dihasilkan (Terbaik dan Ilmiah). Dibawah ini adalah sensor water level yang digunakan oleh penulis.



**Gambar 2.9** Sensor *Water Level*  
Sumber : (Dani, 2018)

### 2.2.13 *Internet Of Things*

*Internet of Things*, juga dikenal sebagai *IoT*, adalah sebuah konsep yang memperluas manfaat konektivitas internet yang selalu aktif yang memungkinkan kita menghubungkan mesin, perangkat, dan objek fisik

lainnya ke jaringan sensor dan perangkat. Kinerja sendiri, memungkinkan mesin untuk berkolaborasi dan bahkan beroperasi secara independen pada informasi yang baru diperoleh. *Internet of Things* atau yang biasa dikenal dengan *IoT* adalah sebuah gagasan bahwa semua objek dunia nyata dapat berkomunikasi satu sama lain sebagai bagian dari sistem yang terintegrasi dengan menggunakan *internet* sebagai penghubung (Sujadi *et al.*, 2021). Misalnya, pengawasan video yang dipasang di sepanjang jalan yang terhubung ke koneksi *internet* dan dipasang di ruang control mungkin berjarak puluhan kilometer. Atau rumah pintar yang dapat dikelola melalui *smartphone* menggunakan koneksi *internet*. Pada dasarnya perangkat *IoT* terdiri dari sensor sebagai media pengumpulan data, koneksi *internet* sebagai media komunikasi dan *server* sebagai pengumpul informasi yang diterima oleh sensor dan untuk keperluan analisis. Ide awal dari *Internet of Thing* pertama kali dikemukakan oleh Kevin Ashton pada tahun 1999 dalam salah satu presentasinya (Wilianto dan Kurniawan, 2018). Saat ini banyak perusahaan besar yang mulai menjajaki *Internet of Things*, yaitu *Intel*, *Microsoft*, *Oracle* dan masih banyak lagi. Banyak orang memprediksi bahwa pengaruh *Internet of Things* merupakan “*the next big thing*” dalam dunia teknologi informasi, karena *IoT* menawarkan begitu banyak potensi yang dapat digali. Contoh sederhana dari manfaat dan implementasi *Internet of Things* adalah lemari es yang dapat memberitahu pemilik melalui teks atau email bahwa makanan dan minuman habis dan perlu diisi ulang (Surya dan Winardi, 2021).

### 2.2.14 Sketch Arduino

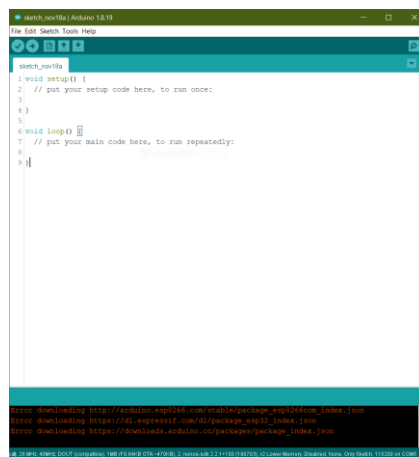
Pada arduino bahasa pemrograman yang digunakan adalah bahasa pemrograman C++. Program yang terdapat pada arduino menjadi 2 bagian utama yaitu :

#### a. *Setup()*

Fungsi *setup* ini dipanggil pertama kali pada saat menjalankan *sketch*. Pada *setup* ini digunakan sebagai tempat inisialisasi *variable*, *pin mode*, penggunaan *library* dan lainnya. Fungsi ini akan dijalankan sekali saja ketika *board* dinyalakan atau di *reset*(Novelan, 2019).

#### b. *Loop()*

Setelah membuat fungsi *setup()* selanjutnya adalah fungsi *loop()*. Seperti namanya, fungsi *loop()* ini akan menjalankan program secara berulang-ulang dan berurutan(Novelan, 2019).



**Gambar 2.10** Arduino IDE

Sumber : (Anon, 2017)

### 2.2.15 Lampu UV

Lampu UV atau *Ultraviolet* merupakan lampu yang menghasilkan sinar *ultraviolet*. Sinar UV memiliki banyak manfaat untuk kesehatan tubuh manusia. Selain itu sinar UV sangat dibutuhkan untuk kelangsungan hidup

untuk hewan maupun tumbuhan. Bagi tumbuhan dengan adanya sinar UV tumbuhan dapat melakukan *fotosintesis* dan berkembang biak dengan baik(SILALAH *et al.*, 2021). Berikut adalah gambar lampu UV yang digunakan oleh penulis.



**Gambar 2.11** Lampu UV  
Sumber : (Webstudi, 2020)

### **2.2.16 Water Pump R385**

Menurut Mucthar (2021) *water pump* r385 adalah jenis pompa air yang menggunakan motor DC dan tegangan searah sebagai sumber tagangannya. Motor akan berputar satu arah apabila tegangan yang diberikan pada kedua terminal berbeda dan apabila pola tegangan yang diberikan dibalik maka arah putaran motor akan terbalik juga(Hasanah, 2020). Dibawah ini adalah gambar *water pump* yang digunakan oleh penulis.



**Gambar 2.12** *Water Pump* R385  
Sumber : (Aldy Razor, 2022)

### **2.2.17 Arduino Uno R3**

Arduino Uno R3 adalah sebuah board minimum *system* mikrokontroller yang di dalamnya terdapat mikrokontroller AVR seri ATMega 328 yang merupakan produk dari Atmel. Umumnya arduino memiliki 14 pin *input/output* yang terdiri dari :



1. 6 pin dapat digunakan sebagai *output* PWM
2. 6 pin sebagai *analog input*
3. Osilator Kristal 16 MHz
4. Sebuah Power Jack
5. Sebuah Koneksi USB
6. Sebuah ICSP Header
7. Tombol Reset

Oleh karena itu, arduino uno mampu mensupport mikrkontroller secara mudah terhubung dengan kabel *power* USB atau kabel *power supply* adaptor AC ke DC maupun dengan batterai(Zanofa *et al.*, 2020). Berikut adalah gambar arduino uno r3 yang digunakan oleh penulis.



**Gambar 2.13** Arduino Uno R3  
Sumber : (Dani, 2018)

### 2.2.18 Solenoid Valve

*Solenoid valve* adalah katup yang digerakan oleh energi listrik melalui solenoida. Solenoida memiliki kumparan yang berfungsi untuk menggerakkan piston yang dapat digerakan oleh arus AC maupun DC. *Solenoid Valve* mempunyai lubang masukan dan lubang keluaran. Tugas *solenoid valve* adalah untuk mematikan, *release, dose, distribute* atau *mix fluids*(La Raufun

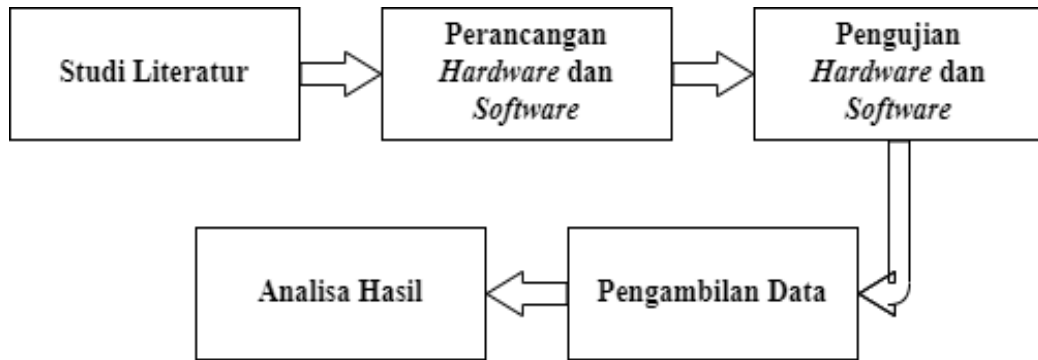
dan Sdani Ardiasyah, 2018). Berikut adalah gambar solenoid valve yang digunakan oleh penulis.



**Gambar 2.14** *Solenoid Valve*  
Sumber : (Aldy Razor, 2022)

### 2.3 Metode Penelitian

Pada penelitian ini penulis menggunakan metode eksperimen yang artinya metode ini membutuhkan penelitian atau implementasi secara langsung kemitra atau tempat penelitian. Penelitian ini dibagi menjadi beberapa tahap atau bagian yaitu studi *Literature*, perancangan dan pembuatan perangkat lunak (*software*) dan perangkat keras (*hardware*), pengujian, pengambilan data dan analisis hasil. Studi *Literature* dilakukan dengan mengumpulkan informasi yang berkaitan dengan penelitian ini. Dengan menggunakan metode eksperimen penulis membuat alat Penerapan Metode Aeroponik Pada Penanaman Kentang Berbasis *Internet Of Things*. Dibawah ini adalah gambar dari metode eksperimen yang digunakan oleh penulis.

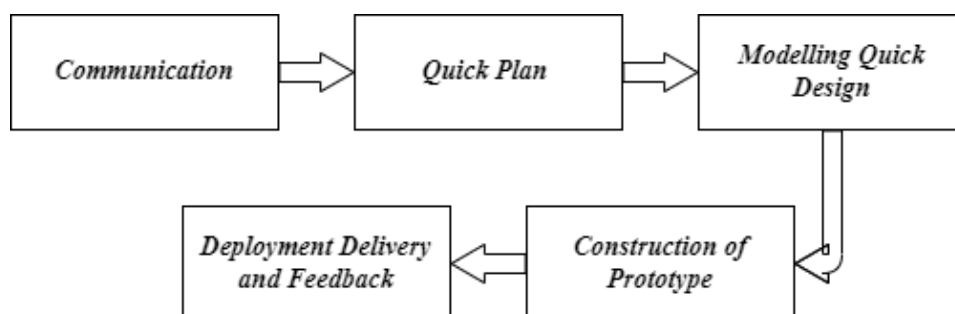


Gambar 2.15 Metode Penelitian

## 2.4 Metode Pengembangan Sistem

### 2.4.1 Metode *Prototype*

Metode *prototype* adalah metode yang dapat digunakan untuk mengembangkan sebuah perangkat yang akan dikembangkan kembali. Metode ini dimulai dengan pengumpulan kebutuhan pengguna dalam hal ini pengguna dari perangkat yang dikembangkan adalah otomatisasi suhu, kelembaban, pH air, dan pengisian penampungan air. Kemudian membuat sebuah rancangan kilat yang selanjutnya akan dievaluasi kembali sebelum diproduksi secara benar. *Prototype* bukanlah merupakan sesuatu yang lengkap, tetapi merupakan sesuatu yang harus dievaluasi dan dimodifikasi kembali. Segala perubahan dapat terjadi saat *prototype* dibuat untuk memenuhi kebutuhan pengguna dan pada saat yang sama memungkinkan pengembang untuk lebih memahami kebutuhan pengguna secara lebih baik. Berikut ini adalah gambar *prototype* yang digunakan oleh penulis.



### Gambar 2.16 Metode *Prototype*

1. *Communication*/komunikasi pengembangan perangkat lunak (*software*) melakukan pertemuan dengan pengguna untuk menentukan kebutuhan perangkat lunak (*software*) yang saat itu diketahui dan untuk menggambarkan area-area dimana definisi lebih lanjut untuk iterasi selanjutnya.
2. *Quick Plan* / perencanaan secara cepat dalam perencanaan ini iterasi pembuatan *prototype* dilakukan secara cepat. Setelah itu dilakukan pemodelan dalam bentuk “rancangan cepat”.
3. *Modelling Quick Design* / model rancangan cepat pada tahap ini memodelkan perancangan menggunakan *tools yed graph editor* yaitu *flowchart* untuk mendefinisikan fungsi dari sistem dan alat.
4. *Construction of Prototype* / pembuatan *prototype* dalam membuat rancangan cepat berdasarkan pada representasi aspek-aspek perangkat lunak (*software*) yang akan terlihat oleh pengguna.
5. *Deployment Delivery dan Feedback* / penyerahan dan memberikan umpan balik terhadap pengembangan *prototype* yang telah dibuat sebelumnya dan memberikan umpan balik yang akan digunakan untuk memperbaiki spesifikasi kebutuhan. Iterasi terjadi saat pengembangan melakukan perbaikan terhadap *prototype* tersebut.

### 2.5 Tampilan Data Sensor Pada Telegram

Dari 4 parameter sebelumnya akan dilakukan pengiriman data melalui jaringan dan disimpan kedalam *variabel* yang sudah dibuat lalu diproses dan

akan ditampilkan pada telegram jika pengguna memilih pilihan yang tersedia pada halaman *bot* telegram.



**Gambar 2.17** Tampilan Data Sensor Pada Telegram