

SISTEM PENGENDALIAN SUHU DAN KELEMBABAN BERBASIS WIRELESS EMBEDDED SYSTEM

SKRIPSI

Diajukan untuk memenuhi persyaratan
memperoleh gelar Sarjana Komputer



Disusun oleh :

Yusuf Oktofani

NIM : 0910683101

**KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
PROGRAM STUDI TEKNIK INFORMATIKA/ILMU KOMPUTER
PROGRAM TEKNOLOGI INFORMASI DAN ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA**

MALANG

2014

LEMBAR PERSETUJUAN

SISTEM PENGENDALIAN SUHU DAN KELEMBABAN BERBASIS WIRELESS EMBEDDED SYSTEM

SKRIPSI

Diajukan untuk memenuhi persyaratan
memperoleh gelar Sarjana Komputer



Disusun oleh :

Yusuf Oktofani

NIM : 0910683101

Skripsi ini telah disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Arief Andy Soebroto, ST., M.Kom.

NIP. 19720425 199903 1 002

Aswin Suharsono, ST., MT.

NIK. 840919 06 1 1 0251

LEMBAR PENGESAHAN
SISTEM PENGENDALIAN SUHU DAN KELEMBABAN
BERBASIS WIRELESS EMBEDDED SYSTEM

SKRIPSI

Diajukan untuk memenuhi persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :

Yusuf Oktofani

NIM. 0910683101

Skripsi ini telah diuji dan dinyatakan lulus pada

Tanggal 24 Januari 2014

Penguji I

Penguji II

Adharul Muttaqin, ST., MT.

Agung Setia Budi, ST., MT.

NIP. 197601212005011001

-

Penguji III

Eko Sakti Pramukantoro, S.Kom., M.Kom.

NIK. 860805 06 1 1 0252

Mengetahui

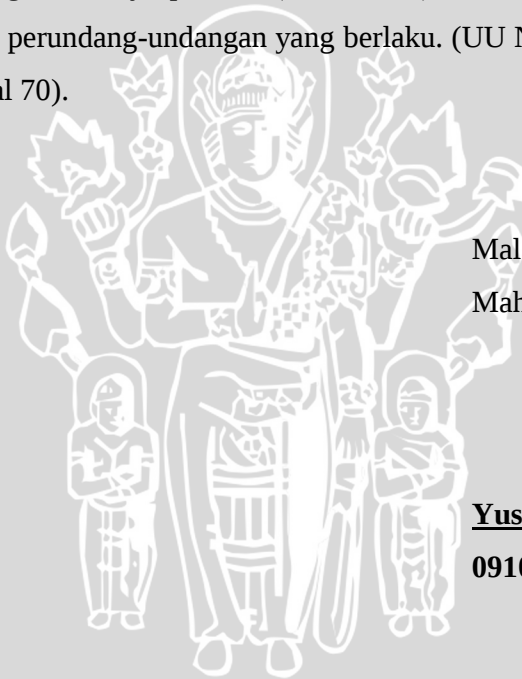
Ketua Program Studi Teknik Informatika/Illmu Komputer

Drs. Marji, MT.

NIP. 19670801 199203 1 001

PERNYATAAN ORISINALITAS SKRIPSI

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah SKRIPSI ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka. Apabila ternyata didalam naskah SKRIPSI ini dapat dibuktikan terdapat unsur-unsur PLAGIASI, saya bersedia SKRIPSI ini digugurkan dan gelar akademik yang telah saya peroleh (SARJANA) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku. (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).



Malang, Januari 2014

Mahasiswa,

Yusuf Oktofani

0910683101

KATA PENGANTAR

Dengan nama Allah SWT Yang Maha Pengasih dan Penyayang. Segala puji bagi Allah SWT karena atas rahmat dan hidayahNya-lah penulis dapat menyelesaikan Tugas Akhir yang berjudul “Sistem Pengendalian Suhu dan Kelembaban Berbasis *Wireless Embedded System*“. Tugas Akhir ini disusun untuk memenuhi sebagian persyaratan memperoleh gelar Sarjana Komputer di Program Studi Teknik Informatika, Program Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya Malang.

Penulis menyadari bahwa tugas akhir ini dapat terselesaikan dengan baik berkat bantuan, bimbingan dan petunjuk dari banyak pihak selama penulisan tugas akhir ini. Oleh karena itu, penulis ingin menyampaikan rasa hormat dan terima kasih penulis kepada :

1. Kepada Allah S.W.T atas semua karuniamu, rahmat mu serta anugerahmu ini saya dapat menuliskan tugas akhir dengan baik
2. Bapak Basar, Ibu Sofiyatun Hikmah, Mas Aghus Irfan Miftakhi dan seluruh keluarga besar yang senantiasa tiada henti-hentinya memberikan dukungan moril, materil dan doa'a demi terselesainya tugas akhir ini.
3. Bapak Ir.Sutrisno, M.T, Bapak Ir.Heru Nurwarsito, M.Kom, Bapak Himawat Aryadita, S.T, M.Sc, dan Bapak Eddy Santoso, S.Kom selaku Ketua, Wakil Ketua 1, Wakil Ketua 2, Wakil Ketua 3 Program Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya.
4. Bapak Drs. Marji, MT. dan Bapak Issa Arwani S.Kom, MSc. selaku Ketua Program dan Sekretaris Program Studi Teknik Informatika Universitas Brawijaya.
5. Bapak Arief Andy Soebroto, ST., M.Kom. dan Aswin Suharsono, ST., MT. selaku dosen pembimbing yang telah banyak memberikan bimbingan, ilmu dan saran dalam penyusunan tugas akhir ini.
6. Bapak Ir. Heru Nurwasito, M.Kom. selaku dosen penasehat akademik yang telah memberikan nasehat kepada penulis selama menempuh masa studi.
7. Seluruh Bapak/Ibu Dosen beserta Staff Laboratorium, Administrasi dan Perpustakaan Program Teknologi Informasi Dan Ilmu Komputer.

8. Teman – teman seperjuangan : Arif Hidayatullah, S.Kom, Aрга Suwastika P.P, S.Kom, Thierry Rahman Aziz, S.Kom, K.H. Septa Hendra, S.kom., BHA, Himawan Aditya Perdana, S.Kom., Antarangga Bhawika S.Kom., Moh. Halimi S.Kom., Rizky Ananda Aditya S.Kom., Hafidz Rozaq, S.Kom., Dhimas Sawung, S.Kom dll.
9. Teman Kos Mbetek Jaya : Zunan Ahzan Algoni, ST., Adi Wahyu Purwanto, S.Sos., Rizky Hanani, Lauma Yuda, ST., Lauma Dwi, SE., Galih Satya, SH., Gilang, SH., Yana Adi, Erwin Tidar, S.St., Dicky Arinta, S.Pd., Doni Sukmawan, SE.
10. Semua rekan penulis di Teknik Informatika angkatan 2009 (TIF 09).
11. Semua pihak yang tidak dapat penulis sebutkan satu per satu yang terlibat baik secara langsung maupun tidak langsung demi terselesaikannya tugas akhir ini.

Hanya doa yang bisa penulis berikan dan semoga Allah SWT memberikan pahala serta balasan kebaikan yang berlipat. Amin.

Penulis menyadari bahwa tugas akhir ini masih banyak kekurangan dan jauh dari sempurna, karena keterbatasan materi dan pengetahuan yang dimiliki penulis. Akhirnya semoga tugas akhir ini dapat bermanfaat bagi pembaca dan dapat berguna sebagai acuan bagi penelitian lebih lanjut.

Malang, Januari 2014

Penulis,

ABSTRAK

Yusuf Oktofani. 2014. Sistem Pengendalian Suhu dan Kelembaban Berbasis *Wireless Embedded System*. Program Teknologi Informasi dan Ilmu Komputer, Universitas Brawijaya, Malang. Dosen Pembimbing : Arief Andy Soebroto, S.T, M.kom. dan Aswin Suharsono, ST., MT.

Indonesia sebagai negara agraris memiliki potensi yang besar sekali dalam bidang pertanian. Salah satu produk tanaman pertanian indonesia adalah tomat. Dalam perkembangannya tanaman tomat perlu perawatan khusus karena tanaman tomat rentan terhadap perubahan suhu dan kelembaban. Untuk mendapatkan hasil panen yang memiliki kualitas dan produktifitas perlu adanya pengkondisian lingkungan tanaman agar tercipta atmosfer yang menunjang pertumbuhan tanaman. *Embedded system* atau sistem tertanam merupakan sistem komputer yang dirancang khusus untuk melakukan tujuan atau fungsi tertentu. Penelitian ini bertujuan untuk membuat sistem yang mampu mengendalikan suhu dan kelembaban serta dapat memantau perubahan kondisi lingkungan tanaman secara nirkabel. Sistem ini menggunakan perangkat *embedded system* yang praktis dan hemat daya. Perangkat *embedded* yang digunakan adalah arduino uno sebagai pusat pemrosesan data dan WiFi *shield* sebagai modul pengiriman data. Dari hasil pengujian sistem mampu memberi tindakan terhadap perubahan suhu dan kelembaban yang melebihi atau kurang dari batas yang ditentukan. Sistem mampu mengirimkan nilai-nilai sensor ke dalam database yang berada pada *server* secara *realtime*.

Kata Kunci : arduino, *embedded system*, pengendali suhu dan kelembaban, dan *wireless node* sensor.

ABSTRACT

Yusuf Oktofani. 2014. *Temperature and Humidity Control System Based Wireless Embedded Systems*. Information Technology and Computer Science Program, Brawijaya University, Malang. Advisors : Arief Andy Soebroto, S.T, M.kom. and Aswin Suharsono, ST., MT.

Indonesia as an agricultural country has great potential in the field of agriculture. One of the product of indonesia agricultural are tomatoes. In the progress of tomato plants need special care because of tomato plants susceptible to changes of temperature and moisture. To get the harvest having the quality of and productivity needed conditioning environment of plants in order to have the atmosphere that support the growth of plants. Embedded system is computer systems that specially designed to conduct a purpose or a specific function. This research aims to make system can control temperatures, moisture and monitor the change of the weather conditions plant wirelessly. This system uses practical embedded systems devices and power saving. Embedded devices that use are arduino uno as the data processing center and WiFi shield as data transmission module. The result of testing system capable of affording action on changes of temperature and moisture that which exceeds or less than a prescribed limit. The system is able to transmit the values of sensor into database on server in realtime.

Keywords : *arduino, embedded system, controlling the temperature and humidity, and wireless node sensor.*

DAFTAR ISI

KATA PENGANTAR	i
ABSTRAK	iii
ABSTRACT	iv
DAFTAR ISI	v
DAFTAR GAMBAR	ix
DAFTAR TABEL	xii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah	4
1.4 Tujuan Penelitian	4
1.5 Manfaat Penelitian	5
1.6 Sistematika Penulisan	5
BAB II TINJAUAN PUSTAKA	7
2.1 Kajian Pustaka	7
2.2 <i>Controll System</i> (Sistem kendali)	10
2.3 <i>Embedded System</i> (Sistem Tertanam)	11
2.4 <i>Wireless sensor network</i> (WSN)	12
2.4.1 Topologi <i>wireless sensor network</i>	14
2.5 Protokol Komunikasi Sistem <i>Wireless Sensor Network</i>	16
2.5.1 Model <i>Open System Interconnection / OSI Layer</i>	16
2.5.2 Internet Protokol (IP)	17
2.5.3 Alamat <i>broadcast</i>	19
2.5.4 <i>Subnet Mask</i>	20

2.6	Sensor DHT11	20
2.6.1	Karakteristik DHT11.....	20
2.7	Sensor SEN0114.....	21
2.8	Relay.....	23
2.8.1	Prinsip Kerja Relay	23
2.9	Arduino.....	25
2.9.2	Arduino IDE.....	26
2.9.3	Pemrograman Arduino.....	27
2.10	Modul Arduino WiFi	31
BAB III METODOLOGI PENELITIAN.....		32
3.1	Studi Literatur.....	33
3.2	Analisis Kebutuhan	33
3.2.1	Analisis kebutuhan <i>Input</i>	33
3.2.2	Analisis Kebutuhan <i>Output</i>	34
3.2.3	Analisis Fungsi dan Kinerja	34
3.3	Perancangan.....	34
3.4	Implementasi	35
3.5	Pengujian dan Analisis	36
3.6	Kesimpulan dan Saran.....	36
BAB IV PERANCANGAN SISTEM.....		37
4.1	Diskripsi Kerja Sistem Secara Keseluruhan.....	38
4.2	Analisis Kebutuhan	39
4.2.1	Analisis Kebutuhan Perangkat Keras.....	39
4.2.2	Analisis Kebutuhan Perangkat Lunak.....	39
4.3	Perancangan Perangkat <i>Node</i> Sensor	40

4.3.1	Perancangan Sistem pembacaan tingkat kelembaban tanah dengan menggunakan sensor SEN0114.....	41
4.3.2	Perancangan Sistem pembacaan suhu ruangan dengan menggunakan sensor DHT11	43
4.3.3	Perancangan Sistem relay aktuator	45
4.3.4	Perancangan Sistem pengiriman data secara <i>wireless</i> menggunakan arduino WiFi <i>shield</i>	46
4.3.5	Perancangan Sistem pengolahan data	49
4.4	Perancangan Perangkat Lunak	50
4.4.1	Perancangan Pemrograman Arduino.....	51
4.4.2	Perancangan <i>Datasink</i>	53
BAB V IMPLEMENTASI SISTEM.....		57
5.1	Batasan Implementasi.....	58
5.2	Implementasi <i>Node</i> Sensor.....	58
5.2.1	Implementasi Perangkat Keras <i>Node</i> Sensor	58
5.2.2	Pemrograman Arduino	62
5.3	Implementasi <i>Datasink</i>	69
5.3.1	Implementasi Database	69
5.3.2	Pemrograman PHP	70
BAB VI PENGUJIAN DAN ANALISIS		72
6.1	Lingkungan pengujian	72
6.2	Pengujian Unit.....	73
6.2.1	Pengujian Sensor Suhu.....	73
6.1.2	Pengujian Sensor SEN0114	75
6.1.3	Pengujian Relay	78
6.1.4	Pengujian WiFi <i>Shield</i>	80

6.2 Pengujian Sistem Secara Keseluruhan	82
6.2.1 Pengujian Konektifitas Jaringan Nirkabel pada <i>Node</i> sensor	83
6.2.2 Pengujian Aktuator Terhadap Nilai Sensor.....	85
6.2.3 Pengujian Ketepatan Data Transfer dan Kebenaran Data Terkirim	87
BAB VI PENUTUP	93
6.1 KESIMPULAN	93
6.2 SARAN	93
DAFTAR PUSTAKA	DP-1
LAMPIRAN.....	L-1



DAFTAR GAMBAR

Gambar 2. 1 Arsitektur WSN pada sistem monitoring polusi	8
Gambar 2. 2 Open loop system	10
Gambar 2. 3 Closed loop system	11
Gambar 2. 4 Susunan Standar <i>Embedded System</i>	11
Gambar 2. 5 WSN	13
Gambar 2. 6 Arsitektur WSN	13
Gambar 2. 7 Topologi Jaringan <i>Point to Point</i>	14
Gambar 2. 8 Topologi Jaringan <i>Multidrop</i>	15
Gambar 2. 9 Topologi Jaringan <i>Web</i>	16
Gambar 2. 10 Alamat Internet	18
Gambar 2. 11 Notasi Desimal	18
Gambar 2. 12 Kelas-kelas Alamat Internet	19
Gambar 2. 13 Konfigurasi pin sensor DHT11	21
Gambar 2. 14 Sensor DHT11	21
Gambar 2. 15 Skema SEN0114	22
Gambar 2. 16 Sensor SEN0114	23
Gambar 2. 17 Konsisi saklar relay pada posisi NO	24
Gambar 2. 18 Konsisi saklar relay pada posisi NC	24
Gambar 2. 19 Jenis-jenis Relay	25
Gambar 2. 20 Tampilan Papan Arduino Uno Tampak Atas	26
Gambar 2. 21 Menunjukkan Tampilan dari Arduino IDE	27
Gambar 2. 22 Struktur dasar Pemrograman Arduino	28
Gambar 2. 23 Fungsi Setup() pada Pemrograman Arduino	28
Gambar 2. 24 Fungsi While() pada Pemrograman Arduino	28
Gambar 2. 25 Pendeklarasian <i>Variable</i> pada Pemrograman Arduino	29
Gambar 2. 26 Contoh Pendeklarasian <i>Variable</i> pada Pemrograman Arduino	29
Gambar 2. 27 Contoh Penggunaan <i>Variable Scope</i> pada Pemrograman Arduino	30
Gambar 2. 28 Pendeklarasian Fungsi pada Pemrograman Arduino	30
Gambar 2. 29 Contoh Pendeklarasian Fungsi pada Pemrograman Arduino	30
Gambar 2. 30 Pin pada WiFi <i>Shield</i>	31

Gambar 3. 1 Diagram blok metodologi penelitian.....	32
Gambar 3. 2 Diagram blok perancangan sistem	35
Gambar 3. 3 Diagram blok tahapan implementasi sistem	36
Gambar 4. 1 Pohon Perancangan Sistem	37
Gambar 4. 2 Diagram blok kerja sistem.....	38
Gambar 4. 3 Perancangan perangkat keras <i>node</i> sensor	40
Gambar 4. 4 Sensor SEN0114	42
Gambar 4. 5 Skema hubungan arduino dengan sensor SEN0114.....	43
Gambar 4. 6 Sensor DHT11.....	44
Gambar 4. 7 Skema hubungan arduino dengan sensor DHT11	44
Gambar 4. 8 Skema hubungan arduino dengan relay	46
Gambar 4. 9 Pemasangan WiFi <i>shield</i> dengan arduino	46
Gambar 4. 10 Skema hubungan antara WiFi <i>shield</i> dengan arduino.....	47
Gambar 4. 11 Proses pengiriman data dari node sensor ke server.....	49
Gambar 4. 12 Skema pin masukan dan keluaran pada arduino	50
Gambar 4. 13 <i>Flowchart</i> Node Sensor.....	52
Gambar 4. 14 <i>Pseudocode</i> Algoritma Node Sensor.....	53
Gambar 4. 15 Skema proses pada <i>datasink</i>	54
Gambar 4. 16 ERD Data Sensor	54
Gambar 4. 17 <i>Flowchart</i> algoritma koneksi antara <i>datasink</i> dan <i>node</i> sensor	55
Gambar 4. 18 <i>Pseudocode</i> koneksi database	55
Gambar 4. 19 Desain <i>website</i>	56
Gambar 5. 1 Diagram Pohon Implementasi.....	57
Gambar 5. 2 Pemasangan kabel pada DHT11	59
Gambar 5. 3 Konfigurasi sensor SEN0114.....	59
Gambar 5. 4 Konfigurasi pin sensor pada arduino.....	60
Gambar 5. 5 Konfigurasi pin aktuator pada arduino.....	60
Gambar 5. 6 Pemasangan relay pada arduino	61
Gambar 5. 7 Perangkat node sensor.....	61
Gambar 5. 8 Potongan kode fungsi void setup	62
Gambar 5. 9 Potongan kode fungsi void loop.....	63

Gambar 5. 10 Potongan kode fungsi bacaSensor.....	63
Gambar 5. 11 Potongan kode fungsi SerialEvent	64
Gambar 5. 12 Potongan kode fungsi aktuator.....	65
Gambar 5. 13 Potongan kode fungsi Repeats	65
Gambar 5. 14 Potongan kode fungsi paket_data.....	66
Gambar 5. 15 Potongan kode fungsi paket_data.....	66
Gambar 5. 16 Potongan kode fungsi kirim_data.....	67
Gambar 5. 17 Pemberitahuan pada arduino IDE	68
Gambar 5. 18 Tampilan informasi node sensor melalui serial monitor.....	68
Gambar 5. 19 Struktur tabel pada database.....	69
Gambar 5. 20 <i>Source Code</i> inputSensor.php	70
Gambar 5. 21 Tampilan <i>web</i> sistem pantau	71
Gambar 6. 1 Desain model simulasi	73
Gambar 6. 2 Suhu terukur dari DHT11.....	74
Gambar 6. 3 Suhu terukur dari termometer pembanding.....	74
Gambar 6. 4 Grafik kurva kalibrasi kelembaban tanah.....	77
Gambar 6. 5 Relay aktif	79
Gambar 6. 6 Relay mati	79
Gambar 6. 7 Tampilan program pengujian WiFi shield	81
Gambar 6. 8 Pengujian konektifitas datasink dengan WiFi shield	81
Gambar 6. 9 Tampilan status WiFi shield.....	84
Gambar 6. 10 Pengujian konektifitas <i>node</i> sensor dengan komputer <i>server</i>	84
Gambar 6. 11 Perubahan nilai-nilai sensor pada serial monitor	89
Gambar 6. 12 Data-data sensor pada database.....	90

DAFTAR TABEL

Tabel 2. 1 Perbedaan Spesifikasi Sistem Monitoring	8
Tabel 2. 2 Pengujian sensor suhu	9
Tabel 2. 3 Perbedaan Spesifikasi Sistem Kendali	9
Tabel 2. 4 Tipe Sensor	12
Tabel 2. 5 OSI Layer	17
Tabel 2. 6 Alamat broadcast	19
Tabel 2. 7 Subnet Mask	20
Tabel 4. 1 Konfigurasi pin	41
Tabel 4. 2 Struktur tabel database	56
Tabel 5. 1 Tabel database arduino	69
Tabel 6. 1 Pengujian Sensor Suhu	75
Tabel 6. 2 Pengujian Sensor SEN0114	76
Tabel 6. 3 Nilai kelembaban dalam rentang 0-100%	78
Tabel 6. 4 Hasil pengujian WiFi shield	82
Tabel 6. 5 Jangkauan Nilai Sensor Untuk Masing-masing Kondisi	85
Tabel 6. 6 Hasil Pengujian Aktuator	86
Tabel 6. 7 Hasil pengujian integrasi	91

BAB I PENDAHULUAN

1.1 Latar Belakang

Indonesia sebagai negara agraris memiliki potensi yang besar sekali dalam bidang pertanian. Sumber alam yang berlimpah serta kondisi alam Indonesia yang menunjang menjadikan negara Indonesia memiliki peluang yang besar bagi kemajuan agroindustri. Salah satu produk pertanian Indonesia adalah tomat. Dari sekian banyak varietas tomat yang ada jenis tomat yang banyak ditanam petani adalah tomat varietas ratna, berlian, precious 206, kingkong dan intan. Sedangkan dari hasil survei yang telah dilakukan di lapangan varietas yang digunakan adalah varietas *Artaloka* [MIK-11].

Tanaman tomat membutuhkan perawatan yang khusus untuk pertumbuhan dan hasil optimal. Suhu udara rata-rata harian yang optimal untuk pertumbuhan tanaman tomat adalah suhu siang hari 18-29 °C dan pada malam hari 10-20 °C [MIK-11]. Tanaman tomat bisa tumbuh baik di dataran tinggi maupun di dataran rendah tergantung varietasnya. Tanaman tomat dapat tumbuh baik di dataran tinggi (lebih dari 700 m dpl), dataran medium (200 m - 700 m dpl), dan dataran rendah (kurang dari 200 m dpl). Faktor temperatur dapat mempengaruhi warna buah. Pada temperatur tinggi (diatas 32°C) warna buah tomat cenderung kuning, sedangkan pada temperatur tidak tetap warna buah cenderung tidak merata. Temperatur ideal dan berpengaruh baik terhadap warna buah tomat adalah antara 24°C - 28°C yang umumnya merah merata. Keadaan temperatur udara dan kelembaban yang tinggi berpengaruh kurang baik terhadap pertumbuhan. Kelembaban yang relatif diperlukan untuk tanaman tomat adalah 70 - 80 %. Tanaman tomat merupakan tanaman yang bisa tumbuh disegala tempat dari daerah dataran rendah sampai daerah dataran tinggi, untuk pertumbuhan yang baik tanaman tomat membutuhkan tanah yang gembur, kadar keasaman pH antara lain 5-6, tanah sedikit mengandung pasir, dan banyak mengandung hara, serta pengairan yang teratur [TMT-13].

Untuk mendapatkan hasil panen yang memiliki kualitas dan produktifitas tinggi serta hasil yang seragam perlu adanya pengkondisian lingkungan sistem produksi agar tercipta atmosfer yang menunjang pertumbuhan tanaman tomat. Salah satu metode pencapaiannya yaitu dengan cara bercocok tanam secara hidroponik dan menggunakan *green house* sebagai tempat budidaya tanaman produksi.

Green house atau rumah kaca adalah struktur berbentuk rumah kecil yang terbuat dari kaca, plastik atau bahan tembus kaca yang berfungsi untuk memanipulasi kondisi lingkungan agar tanaman didalamnya berkembang optimal. Manipulasi kondisi lingkungan ini dilakukan untuk menghindari kondisi lingkungan yang tidak dikehendaki seperti cuaca ekstrim dan menciptakan kondisi lingkungan yang dikehendaki [CLA-07].

Permasalahan yang dihadapi dalam pengkondisian keadaan lingkungan tersebut adalah perlunya monitoring dan kendali yang dilakukan secara berkelanjutan selama masa budidaya tanaman. Hal ini tentu sulit dilakukan oleh petugas ataupun orang yang berkepentingan dengan *green house* untuk terus menerus memantau dan mengontrol keadaan lingkungan selama 24 jam. Meninjau permasalahan tersebut maka perlu adanya sistem monitoring dan kendali parameter lingkungan yang menunjang dan memudahkan pengkondisian keadaan lingkungan. Berdasarkan uraian diatas, *embedded system* sangat cocok digunakan untuk diimplementasikan pada sebuah perangkat lunak dan perangkat keras untuk melakukan sebuah tugas atau pekerjaan secara khusus guna meminimalkan penggunaan sumber daya.

Perkembangan teknologi sistem tertanam (*Embedded system*) ini telah membawa dampak yang sangat besar dalam kehandalan dan kepraktisannya dalam mengolah data yang tidak terlalu kompleks, penggunaan sumber daya yang minimal dan harga yang murah merupakan keunggulan tersendiri menggunakan perangkat sistem tertanam [DDU-07]. Penggunaan teknologi sistem tertanam dapat memenuhi kebutuhan akuisisi data mentah pada pemantauan lahan pertanian. Dengan perangkat *embedded system* proses akuisisi data pada saat monitoring kondisi fisis akan lebih praktis dan membutuhkan sumber daya yang kecil.

Teknologi *Wireless Sensor Network* (WSN) ialah proses komunikasi wireless dengan menggunakan sensor dalam satu jaringan yang saling terhubung dalam satu *gateway*. Beberapa sensor tersebut dapat diletakkan di tempat – tempat yang berbeda untuk memonitoring suatu kondisi di lingkungan sekitarnya yang disebut dengan *node* sensor [IWS-09]. Salah satu kelebihan dari WSN ini ialah tanpa menggunakan kabel dan jangkauan WSN juga lebih luas.

Sistem pemantauan lahan pertanian membutuhkan data yang selalu diperbaharui setiap waktu. Oleh karena itu sistem ini menerapkan sistem informasi waktu nyata (*Real time information system*). Sebuah sistem waktu nyata adalah sistem dengan kebenarannya secara logis didasarkan pada kebenaran hasil-hasil keluaran sistem dan ketepatan waktu hasil-hasil tersebut dikeluarkan [RIS-12].

Berdasarkan paparan kebutuhan pengendalian suhu ruangan dan kelembaban tanah, maka diusulkan penelitian dengan judul Sistem Pengendalian Suhu dan Kelembaban Berbasis *Wireless Embedded System*. Pada penelitian ini penulis akan membangun suatu alat kendali otomatis untuk menciptakan kondisi lingkungan tumbuh tanaman yang ideal untuk tanaman tomat. Pengendalian tersebut dengan mengatur suhu dan kelembaban tanah dari lingkungan tumbuh tanaman tomat dengan menggunakan mikrokontroler. Data pemantauan suhu ruangan dan kelembaban tanah lahan pertanian tomat menerapkan teknologi *wireless sensor network* untuk sistem pemantauannya. Data akan ditransmisikan kedalam jaringan *server* untuk disimpan didalam database dan dapat diakses melalui *website*.

1.2 Rumusan Masalah

Dari latar belakang masalah dapat dirumuskan beberapa masalah yang akan dibahas antara lain :

1. Bagaimana desain dari sistem monitor dan kendali suhu serta kelembaban berbasis *wireless embedded system*
2. Bagaimana implementasi dari sistem monitor dan kendali suhu serta kelembaban berbasis *wireless embedded system*

3. Bagaimana pengujian dan analisis sistem monitor dan kendali suhu serta kelembaban berbasis *wireless embedded system*

1.3 Batasan Masalah

Untuk menghindari pembahasan yang terlalu luas, penulis dalam menyelesaikan tugas akhir ini memiliki batasan-batasan masalah yang dibuat tanpa bermaksud menghilangkan maksud dan tujuan awal. Pembahasan masalah tersebut diantaranya adalah:

1. Data yang diambil oleh sistem tertanam adalah data suhu ruangan dan kelembaban tanah.
2. Perangkat sistem tertanam yang digunakan adalah Arduino uno.
3. Perangkat sensor suhu yang digunakan adalah DHT11
4. Perangkat sensor kelembaban tanah yang digunakan adalah SEN0114 keluaran dfrobot
5. Media transmisi data menggunakan media *wireless*.
6. Database yang digunakan adalah mysql
7. Penelitian menggunakan sistem operasi Windows dan *software* Arduino IDE yang bersifat *open source* dan *portable* tanpa harus diinstall.

1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah merancang dan membuat sistem pengendalian kondisi lingkungan lahan pertanian tomat dengan memanfaatkan teknologi *wireless embedded system*. Dengan pemantauan parameter kondisi lingkungan yang diambil adalah suhu ruangan dan kelembaban tanah yang dipantau secara *wireless* melalui *website*. Sistem kendali yang dilakukan adalah dengan mengaktifkan kipas angin ketika suhu melebihi batas dan mengaktifkan sprayer air ketika kelembaban tanah kurang dari batas.

1.5 Manfaat Penelitian

a. Bagi Penulis :

1. Menerapkan ilmu yang telah diperoleh dari Teknik Informatika Universitas Brawijaya.
2. Mendapatkan pemahaman dan wawasan terkait dengan pengambilan data sensor dan kendali otomatis berbasis sistem tanam serta pengiriman data dari mikrokontroler ke komputer melalui jaringan *wireless*.

b. Bagi pengguna

1. Dengan adanya sistem monitoring keadaan lingkungan lahan pertanian, para petani atau pembudidaya tomat dapat mengetahui informasi perubahan suhu ruangan dan kelembaban tanah.
2. Mendapatkan sistem kendali otomatis apabila kondisi lingkungan tidak sesuai dengan kehendak dan menciptakan manipulasi kondisi keadaan lingkungan yang baik untuk tumbuh kembang tanaman.

1.6 Sistematika Penulisan

Penyusunan skripsi ini dibagi dalam enam bab, yang secara garis besar ini dari setiap bab sebagai berikut.

BAB I: PENDAHULUAN

Bab ini membahas latar belakang penelitian, identifikasi masalah, rumusan masalah, batasan masalah, tujuan dan manfaat penelitian dan sistematika penulisan tiap bab dari skripsi ini

BAB II: TINJAUAN PUSTAKA

Bab ini berisi teori dasar dan teori pendukung yang berhubungan dalam penyelesaian penelitian sistem pengendalian suhu dan kelembaban berbasis *wireless embedded system*. Kajian teori yang diambil berasal dari jurnal, buku, dan sumber referensi lainnya yang berhubungan dengan topik yang akan diteliti.

BAB III: METODOLOGI PENELITIAN

Bab ini berisi metode yang digunakan dalam penelitian yang terdiri dari studi literatur, perancangan, implementasi, pengujian dan analisis, serta pengambilan kesimpulan dan saran.

BAB IV: ANALISIS SISTEM PERANCANGAN APLIKASI

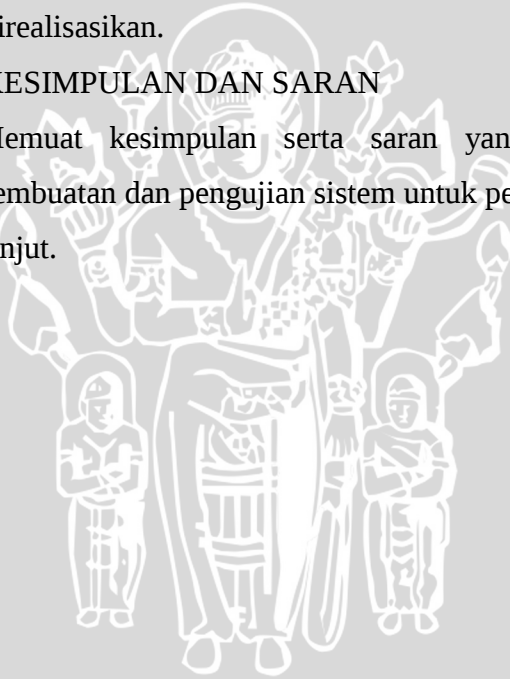
Membahas analisis kebutuhan dan perancangan system serta hasil yang diharapkan.

BAB V: IMPLEMENTASI DAN EVALUASI

Membahas tentang implementasi dari sistem dan memuat proses dan hasil pengujian terhadap sistem yang telah direalisasikan.

BAB VI: KESIMPULAN DAN SARAN

Memuat kesimpulan serta saran yang diperoleh dari pembuatan dan pengujian sistem untuk pengembangan lebih lanjut.



BAB II

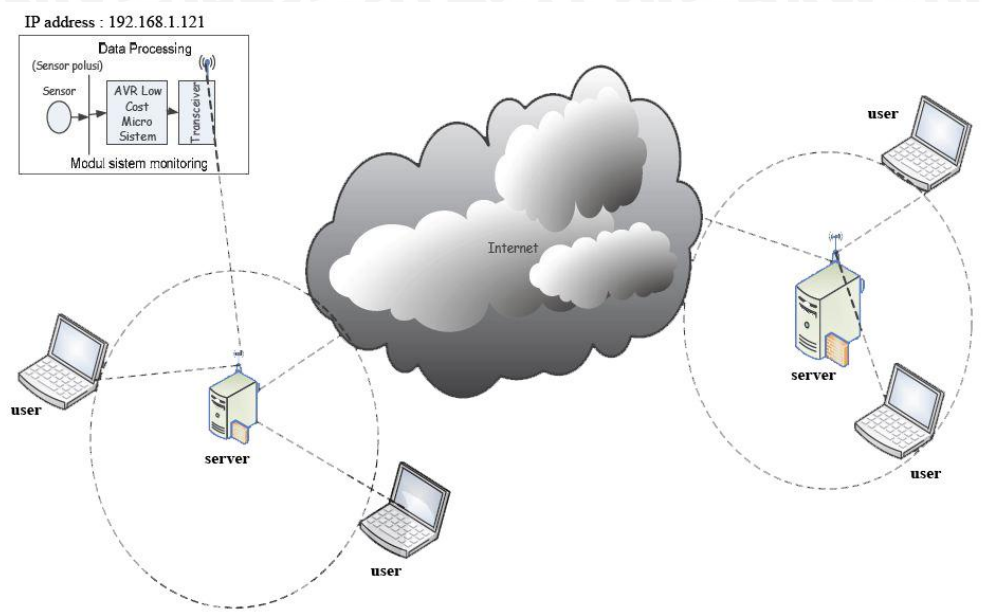
TINJAUAN PUSTAKA

Bab ini membahas tentang kajian pustaka dan dasar teori yang digunakan untuk menunjang penulisan skripsi mengenai Sistem Pengendalian Suhu dan Kelembaban Berbasis *Wireless Embedded System*. Beberapa dasar teori yang dimaksud adalah *controll system*, *embedded system*, *Wireless sensor network*, protokol komunikasi sistem *wireless sensor network*, sensor suhu ruangan, sensor kelembaban tanah, relay, arduino, dan *Wireless Fidelity (WiFi) shield*.

2.1 Kajian Pustaka

Sistem *wireless sensor network* berbasis *Internet Protocol (IP)* didesain sebagai sebuah jaringan komunikasi sensor yang terhubung secara nirkabel untuk memonitoring kondisi fisis atau kondisi lingkungan seperti keadaan suhu, kelembaban, radiasi dan lain-lain. Data pembacaan sensor tersebut dapat diakses secara nirkabel sesuai dengan *IP address* perangkat sensor tersebut. Pada penelitian sebelumnya telah dilakukan oleh Marimbun Sibarani dengan judul Implementasi Sistem *Wireless Sensor Network* Berbasis *Internet Protocol (IP)* Untuk Pemantauan Tingkat Polusi Udara [ISW-08]. Pada penelitian ini *wireless sensor network* digunakan untuk sistem memonitoring tingkat polusi udara dengan menggunakan sensor TGS 2600 *Air Quality Sensor*. Sensor tersebut digunakan untuk mendeteksi tingkat polusi udara atau tingkat kontaminasi udara terutama oleh kandungan karbon dioksida (CO_2). Perangkat yang digunakan sebagai pusat pemrosesan data menggunakan *DT-AVR Low Cost Micro System*. Data pembacaan sensor yang telah diolah didalam mikrokontroler selanjutnya dikirim kedalam perangkat *web server* menggunakan perangkat *web server* NM7010A-LF. Pemantauan dilakukan melalui *website* dengan mengakses *IP address* perangkat sensor tersebut.

Sistem kerja dari *wireless sensor network* berbasis IP ini ditunjukkan pada Gambar 2.1 yang menggambarkan sebuah sistem yang diimplemnatasikan menggunakan sensor polusi udara, perangkat pengambilan dan pengolahan data, serta pengiriman data ke dalam jaringan *server* internet.



Gambar 2. 1 Arsitektur WSN pada sistem monitoring polusi
Sumber : [ISW-08]

Pebedaan spesifikasi perangkat keras pada penelitian yang dilakukan oleh Marimbun Sibarani dengan penulis dijelaskan pada Tabel 2.1.

Tabel 2. 1 Perbedaan Spesifikasi Sistem Monitoring

Judul Penelitian	Implementasi Sistem <i>Wireless Sensor Network</i> Berbasis <i>Internet Protocol</i> (IP) Untuk Pemantauan Tingkat Polusi Udara	Sistem Pengendalian Suhu dan Kelembaban berbasis <i>Wireless Embedded System</i>
Jenis sensor	Sensor Polusi Udara TGS 2600 Air Quality Sensor	Sensor suhu DHT11, Sensor kelembaban tanah dfrobot SEN0114
Mikrokontroler	DT-Avr Low Cost Micro System	Arduino Uno R3
Modul pengiriman	NM7010A-LF	Arduino WiFi <i>Shield</i>
Komunikasi data pengiriman	Ethernet	<i>Wireless</i>
Relay	Tidak ada	Ada

Era modernisasi ikut berimbas terhadap modernisasi alat baik di industri maupun di pertanian. Peralatan di sebuah pertanian yang dulunya digerakkan manual oleh manusia kini mulai terotomatisasi yakni dikendalikan secara otomatis oleh mesin itu sendiri. Proses otomatisasi mesin dikenal dengan istilah sistem kontrol atau ada juga yang menyebut sistem pengendalian. Salah satu penerapan sistem kendali pada bidang pertanian adalah Sistem Pengendalian Ruang

Tanaman Anggrek Bulan Berbasis *Microcontroller* [PTA-08]. Pada penelitian ini peneliti memanipulasi keadaan lingkungan tanaman anggrek agar menyerupai dengan kondisi lingkungannya yang lembab. Pengaturan secara otomatis akan memudahkan para penggemar anggrek dan petani anggrek dalam menciptakan lingkungan yang sesuai dengan tanaman anggrek bulan. Pengaturan ini dilengkapi sensor suhu dan sensor kelembaban untuk mengetahui suhu dan kelembaban saat ini, dan juga pewaktu untuk membuka atap supaya tanaman mendapatkan sinar matahari tapi secara tidak langsung. Penelitian ini menggunakan sensor suhu dan pewaktu yang telah berjalan dengan baik, dengan error dari sensor suhu sebesar 0.44% dan error dari pewaktu sebesar 0%, sedangkan untuk sensor kelembaban belum dapat berjalan dengan baik sehingga untuk sensor kelembaban disimulasikan dengan menggunakan potensiometer. Berikut ini data pengujian sensor suhu ditunjukkan pada Tabel 2.2.

Tabel 2. 2 Pengujian sensor suhu

Termometer ruangan (°C)	Sensor LM35			Rata-rata sensor	Error
	S1	S2	S3		
21	21	22	22	21	0%
23	24	23	24	23	0%
25	25	26	25	25	0%
27	27	27	26	26	0%
29	29	29	29	29	0%
30	31	30	31	30	0%
33	32	32	33	33	0%
40	41	39	39	39	2.5%
43	43	44	44	43	0%
52	52	50	52	51	1.9%
Rata-rata error					0.44%

Sumber : [PTA-08]

Pebedaan spesifikasi perangkat keras pada penelitian yang dilakukan oleh Anik Nur Kholifah dengan penulis dijelaskan pada Tabel 2.3.

Tabel 2. 3 Perbedaan Spesifikasi Sistem Kendali

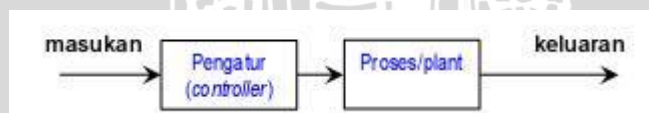
Judul Penelitian	Sistem Pengendalian Ruang Tanaman Anggrek Bulan Berbasis <i>Microcontroller</i>	Sistem Pengendalian Suhu dan Kelembaban berbasis <i>Wireless Embedded System</i>
Jenis sensor	Sensor suhu LM35, Sensor kelembaban ruangan RHKIAN	Sensor suhu DHT11, Sensor kelembaban tanah dfrobot SEN0114
Mikrokontroler	AT89S52	Arduino Uno R3

LCD	Ada	Tidak ada
RTC	Ada	Tidak ada
Modul pengiriman	Tidak ada	Arduino WiFi Shield
Relay	Ada	Ada

2.2 Controll System (Sistem kendali)

Sistem kendali dapat dikatakan sebagai hubungan antara komponen yang membentuk sebuah konfigurasi sistem yang akan menghasilkan tanggapan sistem yang diharapkan. Jadi harus ada yang dikendalikan, yang merupakan suatu sistem fisis, yang biasa disebut dengan kendalian (*plant*). Masukan adalah yang mempengaruhi kendalian, yang mengatur keluaran. Keluaran merupakan hal yang dihasilkan oleh kendalian, artinya yang dikendalikan [MCE-10].

Pada sistem kendali dikenal sistem lup terbuka (*open loop system*) dan sistem lup tertutup (*closed loop system*). Sistem kendali lup terbuka atau umpan maju (*feedforward control*) umumnya mempergunakan pengatur (*controller*) serta aktuator kendali (*control actuator*) yang berguna untuk memperoleh respon sistem yang baik. Sistem kendali ini keluarannya tidak diperhitungkan ulang oleh *controller*. Suatu keadaan apakah *plant* benar-benar telah mencapai target seperti yang dikehendaki masukan atau referensi, tidak dapat mempengaruhi kinerja kontroler.



Gambar 2. 2 Open loop system
Sumber : [MCE-10]

Pada sistem kendali kendali lup tertutup (*closed loop system*) memanfaatkan variabel yang sebanding dengan selisih respon yang terjadi terhadap respon yang diinginkan. Sistem seperti ini juga sering dikenal dengan sistem kendali umpan balik. Aplikasi sistem umpan balik banyak dipergunakan untuk sistem kemudi kapal laut dan pesawat terbang. Perangkat sehari-hari yang juga menerapkan sistem ini adalah penyetelan temperatur pada almari es, oven, tungku, dan pemanas air.

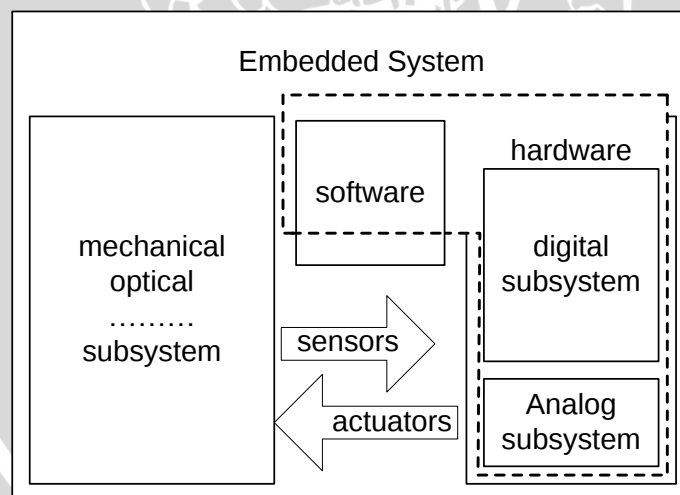


Gambar 2. 3 Closed loop system
Sumber : [MCE-10]

2.3 Embedded System (Sistem Tertanam)

Embedded system adalah suatu sistem komputer yang dibangun untuk melakukan tujuan atau fungsi tertentu. sistem tertanam ini memiliki *central processing unit* (CPU) yang umumnya dalam bentuk mikrokontroller ataupun *microprocessor*. Sistem tertanam dapat pula didefinisikan sebagai sistem dengan ukuran yang relatif kecil dan berbasis komputer yang dikemas dalam bentuk *chip*.

Sistem tertanam dirancang untuk suatu pekerjaan khusus yang dapat dikerjakan secara otomatis, terutama sebagai suatu sistem kontrol. Dengan adanya sistem tertanam, suatu sistem dapat dikerjakan lebih baik, bekerja secara otomatis, berukuran relatif kecil, dan memerlukan biaya yang lebih murah di bandingkan langsung menggunakan komputer [SYT-13].



Gambar 2. 4 Susunan Standar *Embedded System*
Sumber : [SYT-13]

2.4 Wireless sensor network (WSN)

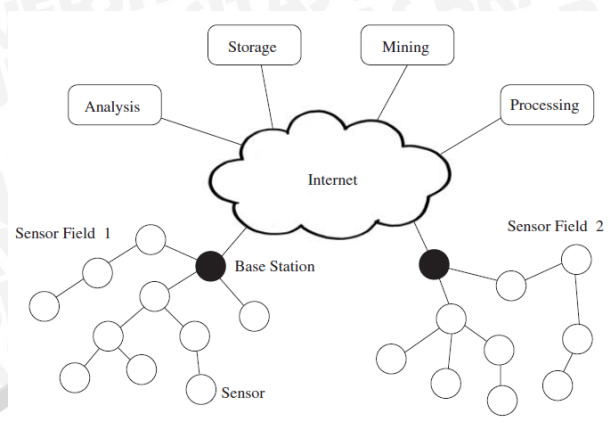
Wireless sensor network adalah sebuah jaringan komunikasi sensor yang terhubung secara *wireless* untuk memonitoring kondisi fisis atau kondisi lingkungan tertentu pada lokasi yang berbeda antara sensor dan pemrosesan datanya. Sensor disini digunakan untuk menangkap informasi sesuai dengan karakteristik kondisi fisis yang akan diamati. Contoh tipe sensor dapat dilihat di Tabel berikut ini [WSN-10]:

Tabel 2. 4 Tipe Sensor

Tipe Sensor	Contoh Sensor
Temperatur	Thermistor, thermocouple
Tekanan	Pressure gauge, barometer, ionization gauge
Optik	Photodiodes, phototransistors, infrared sensors, CCD sensors
Akustik	Piezoelectric resonators, microphones
Mekanik	Strain gauges, tactile sensors, capacitive diaphragms, piezoresistive cells
Gerakan dan Getaran	Accelerometers, gyroscopes, photo sensors
Posisi	GPS, ultrasound-based sensors, infrared-based sensors, inclinometers
Kelembaban	Capacitive and resistive sensors, hygrometers, MEMS-based humidity sensors
Radiasi	Ionization detectors, Geiger–Mueller counters

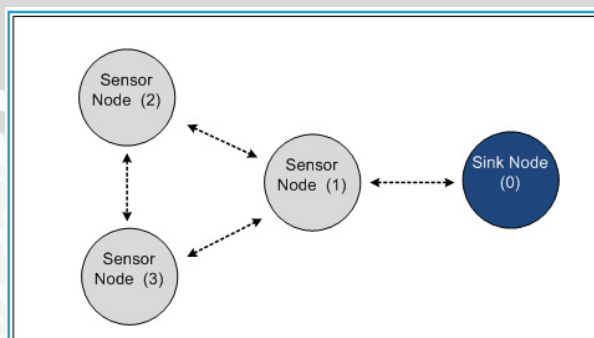
Sumber : [WSN-10]

Kemampuan sensor pada WSN secara luas digunakan untuk melakukan monitoring, WSN yang sederhana dapat menggunakan satu sensor untuk memonitoring suatu fenomena sedangkan untuk WSN yang lebih kompleks dapat digunakan lebih dari satu sensor untuk memonitoring suatu fenomena. WSN tidak hanya terdiri dari komponen sensor tetapi terdiri dari kemampuan memproses, komunikasi dan penyimpanan. WSN tidak hanya bertanggung jawab untuk mengumpulkan data, tetapi untuk analisis di dalam jaringan, korelasi antar *node-node* sensor. WSN yang terdiri dari banyak sensor saat memantau suatu kondisi lingkungan yang luas, *node-node* sensor tidak hanya berkomunikasi antara *node* sensor dengan *base* sensor tetapi berkomunikasi dengan antar *node* dengan menyebarluaskan data sensor seperti pada Gambar 2.5 [WSN-10].



Gambar 2. 5 WSN
Sumber : [WSN-10]

Jaringan ini menghubungkan perangkat-perangkat seperti *node* sensor, *router*, dan *sink node*. Perangkat ini terhubung secara *ad-hoc* dan mendukung komunikasi *multi-hop*. Istilah *ad-hoc* merujuk pada kemampuan perangkat untuk berkomunikasi satu sama lain secara langsung tanpa memerlukan infrastruktur jaringan seperti *router* atau *access point*. Beberapa contoh penerapan jaringan *ad-hoc* antara lain pembangunan jaringan komunikasi di medan perang untuk beberapa lokasi, pusat-pusat komunikasi di daerah bencana alam, sarana koneksi internet pada stand-stand suatu event/pameran dimana tidak dimungkinkan untuk membangun jaringan kabel atau ketidaktersediaan jaringan kabel. *Multi-hop* yaitu istilah yang merujuk pada komunikasi beberapa perangkat yang melibatkan perangkat antara (*intermediate*), *multi-hop* melibatkan perangkat antara seperti *router* untuk meneruskan sebuah paket dari satu *node* ke *node* lain dalam jaringan. Secara sederhana pengertian WSN dalam tulisan ini disajikan pada arsitektur berikut [WSN-10]:



Gambar 2. 6 Arsitektur WSN
Sumber : [WSN-10]

Pada Gambar 2.6 WSN yang terdiri dari tiga buah *node* dan satu buah *sink node* yang terhubung dan berkomunikasi dengan gelombang radio. *Node* sensor 3 dan 2 dapat berkomunikasi langsung secara *ad hoc*. Sementara apabila ingin berkomunikasi dengan *sink node*, kedua *node* dapat mengirim paket datanya ke *node* sensor 1, kemudian *node* sensor 1 akan meneruskan paket data tersebut ke *sink node*. Jadi *node* sensor 1 akan bertindak sebagai *router* atau perantara yang meneruskan paket data dan sebagai penyedia komunikasi *multihop*.

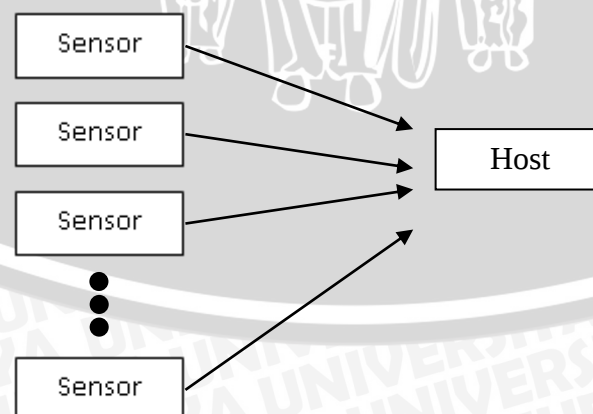
2.4.1 Topologi *wireless sensor network*

Dalam perkembangannya *wireless sensor network* telah dikembangkan dengan beberapa topologi jaringan seperti [ISW-08]:

1. Topologi jaringan *point to point*
2. Topologi jaringan *multidrop*
3. Topologi jaringan *web model*

2.4.1.1 Topologi jaringan *point to point*

Secara teori, topologi jaringan *point to point* memberikan *reliability* yang lebih baik dibandingkan dengan topologi *wireless sensor network* yang lain, karena masing masing sensor secara terpisah tersambung ke *host*. Topologi ini dapat dikembangkan dengan menambahkan *redundant host*. Topologi jaringan ini dapat dilihat seperti Gambar 2.7.

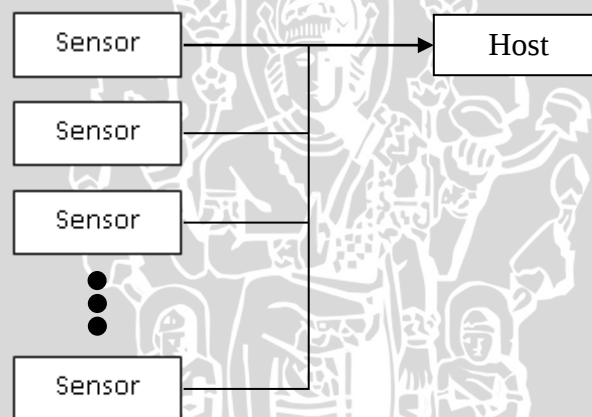


Gambar 2. 7 Topologi Jaringan *Point to Point*
Sumber : [ISW-08]

Pada topologi jaringan *point to point* ini membutuhkan biaya lebih mahal dan manajemen konfigurasi yang agak sulit, semua data dan konfigurasi hanya diproses oleh *host*, karena *node* sensor terhubung langsung ke *host*.

2.4.1.2 Topologi jaringan *Multidrop*

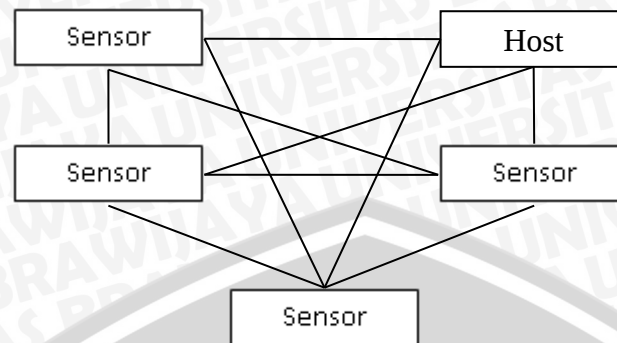
Pada topologi jaringan *multidrop*, setiap sensor memberikan informasi kepada *common medium* yang sebagai perantara antara *node-node* sensor ke *host*. Hal ini membutuhkan ketelitian dalam protokol *hardware* dan *software*. Sistem *multidrop* ini sebagai jaringan dengan sistem *bus*, dan jika dibandingkan dengan topologi *point to point* terdapat sambungan yang lebih sederhana dalam penyambungan ke *host*. Topologi jaringan ini dapat dilihat seperti pada Gambar 2.8.



Gambar 2. 8 Topologi Jaringan *Multidrop*
Sumber : [ISW-08]

2.4.1.3 Topologi *web network*

Pad topologi jaringan *web* ini, setiap sensor terhubung setiap saat. Salah satu keuntungan dari topologi ini adalah adanya kemungkinan pengembangan dari level *intelegence* setiap sensor. Topologi jenis ini dapat digambarkan seperti pada Gambar 2.9:



Gambar 2. 9 Topologi Jaringan Web
Sumber : [ISW-08]

2.5 Protokol Komunikasi Sistem *Wireless Sensor Network*

Sistem komunikasi *Wireless Sensor Network* berbasis *Internet Protocol* merupakan sebuah komunikasi sistem jaringan komputer, sehingga protokol komunikasi yang digunakan adalah protokol komunikasi yang telah dikembangkan untuk membentuk jaringan komputer. Jaringan komputer adalah kumpulan dari sejumlah perangkat berupa komputer, *hub*, *switch*, *router* atau perangkat jaringan lainnya yang terhubung dengan menggunakan media komunikasi tertentu. Perangkat yang terhubung dengan jaringan disebut juga sebagai *node*. Hal ini memungkinkan pengguna dapat bertukar dokumen dan data, mencetak pada *printer* yang sama, dan menggunakan sumber daya jaringan (*hardware* dan *software*) yang ada.

Sebuah jaringan komputer biasanya terdiri dari dua buah komputer atau lebih dan melakukan data *sharing* antar komputer. Informasi dan data bergerak melalui media komunikasi. Media komunikasi yang dipakai dalam membuat jaringan komputer antara lain adalah kabel, jaringan telepon, gelombang radio, satelit, *bluetooth* atau infra merah. Pemakaian media komunikasi ini akan tergantung pada kegunaan dan ukuran jaringan [MOD-12].

2.5.1 Model *Open System Interconnection / OSI Layer*

Model referensi OSI terdiri dari 7 lapisan, mulai dari lapisan fisik hingga aplikasi. Model referensi ini tidak hanya berguna untuk produk-produk LAN saja, tetapi juga sangat diperlukan dalam membangun jaringan internet. OSI menjelaskan bagaimana data dan informasi jaringan berkomunikasi dari sebuah

aplikasi pada sebuah komputer berjalan melalui jaringan menuju aplikasi di komputer lain. OSI menjelaskan melalui pendekatan pemecahan menjadi lapisan lapisan (*layer*). Analogi konsep *layer* adalah seperti dalam departemen / bidang dalam sebuah perusahaan, setiap departemen memiliki tugas yang berbeda dan hanya fokus pada hal tertentu sesuai pembagian tugas. Hubungan antara model referensi OSI dengan protokol internet dapat dilihat pada Tabel 2.5 Setiap *layer* mewakili suatu abstraksi yang berbeda dengan lainnya dan melaksanakan suatu fungsi yang telah didefinisikan [MOD-12].

Tabel 2. 5 OSI Layer

Nama Layer	Fungsi
Aplikasi (<i>Layer 7</i>)	Aplikasi yang saling berkomunikasi antar komputer. Aplikasi <i>layer</i> mengacu pada pelayanan komunikasi pada suatu aplikasi.
Presentasi (<i>Layer 6</i>)	Pada <i>layer</i> bertujuan untuk mendefinisikan format data, seperti ASCII text, binary dan JPEG.
Sesi (<i>Layer 5</i>)	Sesi <i>layer</i> mendefinisikan bagaimana memulai, mengontrol dan mengakhiri suatu percakapan (biasa disebut <i>session</i>)
Transport (<i>Layer 4</i>)	Pada <i>layer 4</i> ini bisa dipilih apakah menggunakan protokol yang mendukung <i>error recovery</i> atau tidak. Melakukan <i>multiplexing</i> terhadap data yang datang, mengurutkan data yang datang apabila datangnya tidak berurutan.
Network (<i>Layer 3</i>)	<i>Layer</i> ini mendefinisikan pengiriman data dari ujung ke ujung. Untuk melakukan pengiriman pada <i>layer</i> ini juga melakukan pengalamatan. Mendefinisikan pengiriman jalur (<i>routing</i>).
Data Link (<i>layer 2</i>)	<i>Layer</i> ini mengatur pengiriman data dari <i>interface</i> yang berbeda. Semisal pengiriman data dari ethernet 802.3 menuju ke <i>High-level Data Link Control</i> (HDLC), pengiriman data WAN.
Physical (<i>Layer 1</i>)	<i>Layer</i> ini mengatur tentang bentuk <i>interface</i> yang berbeda-beda dari sebuah media transmisi. Spesifikasi yang berbeda misal konektor, pin, penggunaan pin, arus listrik yang lewat, <i>encoding</i> , sumber cahaya dll

Sumber: [MOD-12]

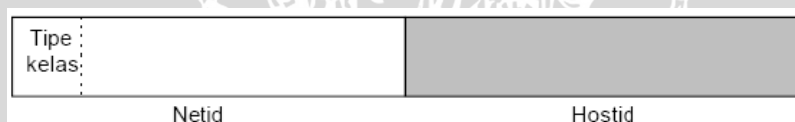
2.5.2 Internet Protokol (IP)

IP memiliki sifat yang dikenal sebagai *unreliable*, *connectionless*, datagram *delivery service* dalam melakukan pengiriman data pada jaringan komputer. *Unreliable* atau ketidakhandalan berarti tidak ada jaminan sampainya

data di tempat tujuan. *Connectionless* berarti dalam mengirim paket dari tempat asal ke tujuan, tidak diawali dengan perjanjian (*handshake*) antara pengirim & penerima. Datagram *delivery service* berarti setiap paket data yang dikirim adalah independen terhadap paket data yang lain. Jalur yang ditempuh antara satu data dengan yang lain bisa berbeda, sehingga kedatangannya pun bisa tidak teratur seperti urutan pengiriman.

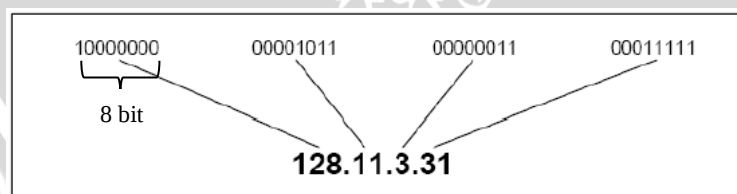
IP address (IPv4) memiliki 32 bit angka yang merupakan *logical address*. IP address bersifat unik, artinya tidak ada *device*, *station*, *host* atau *router* yang memiliki IP address yang sama tetapi setiap *host*, komputer atau *router* dapat memiliki lebih dari satu IP address. Setiap alamat IP memiliki makna netID dan hostID. NetID adalah pada bit-bit terkiri dan hostID adalah bit-bit selain netID (terkanan) [MOD-12].

Pembacaan alamat internet yang merupakan *logical address* lebih mudah dilakukan dengan menyusun IP address dalam bentuk desimal. Dimana setiap 8 bit diwakili satu bilangan desimal. Masing-masing angka desimal ini dipisahkan oleh tanda titik.



Gambar 2. 10 Alamat Internet
Sumber: [MOD-12]

Untuk mempermudah pembacaan, 32 bit alamat internet direpresentasikan dengan notasi desimal.



Gambar 2. 11 Notasi Desimal
Sumber: [MOD-12]

IP address diklasifikasikan menjadi 5 kelas, yaitu kelas A, kelas B, kelas C, kelas D dan kelas E. Kelas-kelas tersebut didesain untuk kebutuhan jenis-jenis organisasi.

	Mulai	Hingga
Kelas A	0 . 0 . 0 . 0	127.255.255.255
	Netid Hostid	Netid Hostid
Kelas B	128 . 0 . 0 . 0	191.255.255.255
	Netid Hostid	Netid Hostid
Kelas C	192 . 0 . 0 . 0	223.255.255.255
	Netid Hostid	Netid Hostid
Kelas D	224 . 0 . 0 . 0	239.255.255.255
	Alamat Multicast	Alamat Multicast
Kelas E	24- . 0 . 0 . 0	255.255.255.255
	Cadangan	Cadangan

Gambar 2. 12 Kelas-kelas Alamat Internet
Sumber: [MOD-12]

2.5.3 Alamat broadcast

Dalam kelas A, B dan C, jika *host id* semuanya di-set 1, alamat tersebut disebut sebagai *direct broadcast address*. Alamat ini digunakan oleh *router* untuk mengirim sebuah paket ke seluruh *host* dalam jaringan tertentu/khusus, sehingga seluruh *host* pada jaringan tertentu tersebut menerima paket dengan alamat ini.

Alamat *broadcast* menggunakan alamat tertinggi dalam jangkauan jaringan. Contoh untuk jaringan 192.168.1.0 dengan subnet mask 255.255.255.0, alamat broadcast akan menjadi 192.168.1.255. Alamat ini juga disebut sebagai *directed broadcast*. Ilustrasi alamat broadcast dapat dilihat dari Gambar 2.11 [MOD-12]:

Tabel 2. 6 Alamat broadcast

host id	192	168	1	20
	11000000	10101000	00000001	00010100
subnet mask	255	255	255	0
	11111111	11111111	11111111	0
network add	192	168	1	0
	11000000	10101000	00000001	00000000
broadcas add	192	168	1	255
	11000000	10101000	00000001	11111111

Sumber: [MOD-12]

2.5.4 Subnet Mask

Setiap jaringan TCP/IP memerlukan nilai *subnet* yang dikenal sebagai *subnet mask* atau *address mask*. Nilai *subnet mask* memisahkan *network id* dengan *host id*. Pembagian kelas IP berdasarkan *subnet mask* dapat dilihat pada Tabel 2.7.

Tabel 2. 7 Subnet Mask

Kelas	Subnet mask
A	255.0.0.0
B	255.255.0.0
C	255.255.255.0

Sumber: [MOD-12]

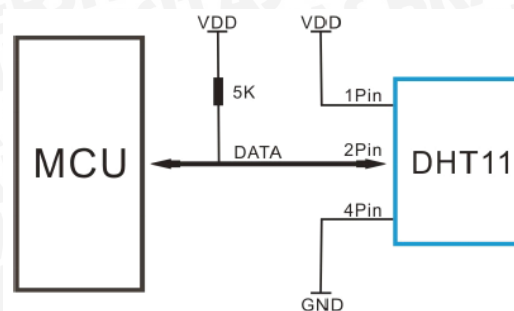
Subnet mask diperlukan oleh TCP/IP untuk memeriksa apakah jaringan yang dimaksud adalah jaringan lokal atau non lokal. Untuk jaringan non lokal berarti harus mentransmisikan paket data melalui sebuah *router*. Dengan demikian diperlukan *address mask* untuk menyaring IP *address* dan paket data yang keluar masuk jaringan tersebut.

2.6 Sensor DHT11

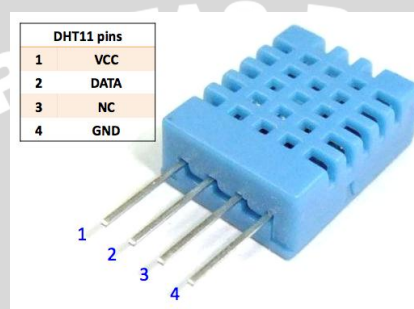
Sensor DHT11 merupakan sensor suhu dan kelembaban udara yang memiliki jangkauan pengukuran suhu antara 0-50°C dan jangkauan pengukuran kelembaban udara 20-95% RH. Dengan ukuran yang kecil dan memakan daya listrik yang kecil modul sensor ini mudah untuk diimplementasikan menggunakan mikrokontroler. DHT11 cukup ekonomis namun memadai untuk aplikasi monitoring suhu dan kelembaban udara [DSD-13].

2.6.1 Karakteristik DHT11

Sensor DHT11 memiliki fitur pengukuran suhu dan kelembaban udara yang sudah dikalibrasi dengan keluaran sinyal digital. Sensor DHT11 memiliki komponen 4 pin namun yang digunakan hanya 3 yaitu pin ground, data, dan 5v [DSD-13]. Gambar 2.13 dijelaskan mengenai konfigurasi pin sensor DHT11 serta Gambar 2.14 mengenai sensor DHT11.



Gambar 2. 13 Konfigurasi pin sensor DHT11
Sumber: [DSD-13]



Gambar 2. 14 Sensor DHT11
Sumber: [DSD-13]

2.7 Sensor SEN0114

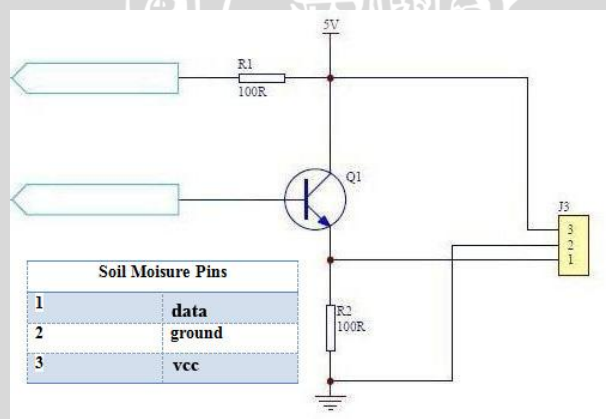
Kelembaban tanah adalah jumlah air yang ditahan di dalam tanah setelah kelebihan air dialirkan. Apabila tanah memiliki kadar air yang tinggi, maka kelebihan air tanah akan dikurangi melalui evaporasi, transpirasi dan transpor air bawah tanah. Kelembaban tanah dapat diketahui melalui berbagai macam metode, yaitu secara langsung melalui pengukuran perbedaan berat tanah (disebut metode *gravimetri*) dan secara tidak langsung melalui pengukuran sifat-sifat lain yang berhubungan erat dengan air tanah. Metode langsung secara *gravimetri* memiliki akurasi yang sangat tinggi namun membutuhkan waktu dan tenaga yang sangat besar. Kebutuhan akan metode yang cepat dalam memonitor fluktuasi kadar air tanah menjadi sangat mendesak sebagai jawaban atas tingginya waktu dan tenaga yang dibutuhkan oleh metode *gravimetri*. Oleh karena itu, dibutuhkanlah sebuah metode baru yang dapat secara langsung mendeteksi kadar air dan kelembaban tanah secara akurat, cepat dan sederhana. Metode tersebut merupakan metode langsung dengan menggunakan sensor [SDS-13].

Sensor SEN0114 adalah sensor kelembaban tanah yang bekerja dengan prinsip membaca jumlah kadar air dalam tanah di sekitarnya. Sensor ini merupakan sensor dengan teknologi rendah namun ideal untuk memantau kadar air tanah untuk tanaman. Sensor ini menggunakan dua konduktor untuk melewatkan arus melalui tanah, kemudian membaca nilai resistensi untuk mendapatkan tingkat kelembaban. Lebih banyak air dalam tanah akan membuat tanah lebih mudah menghantarkan listrik (nilai resistansi lebih kecil), sedangkan tanah kering akan mempersulit untuk menghantarkan listrik (nilai resistansi besar).

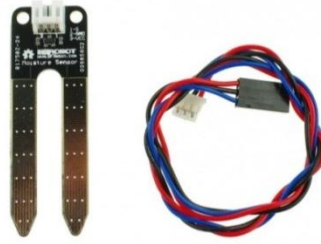
Sensor SEN0114 dalam penerapannya membutuhkan daya sebesar 3,3v atau 5 v dengan keluaran tegangan sebesar 0-4,2v. Sensor ini mampu membaca kadar air yang memiliki 3 kondisi yaitu [SSM-13]:

1. 0 ~300 : tanah kering/ udara bebas
2. 300 ~700 : tanah lembab
3. 700 ~950 : di dalam air

Sensor ini memiliki 3 pin yang terdiri dari pin *ground*, 5v dan data. Pada Gambar 2.15 dijelaskan mengenai skema sensor *soil moisture* serta Gambar 2.16 mengenai sensor *soil moisture*.



Gambar 2. 15 Skema SEN0114
Sumber: [DSS-13]



Gambar 2. 16 Sensor SEN0114
Sumber: [SSM-13]

2.8 Relay

Relay adalah sebuah saklar yang dikendalikan menggunakan arus listrik. Relay memiliki sebuah kumparan dengan tegangan rendah yang dililitkan pada sebuah inti. Terdapat sebuah *armature* yang akan tertarik menuju inti kumparan apabila dialiri listrik tegangan rendah. *Armature* ini dikaitkan pada tuas berpegas, ketika elektromagnet diberikan sumber tegangan maka medan magnet akan menarik *armature* sehingga kontak jalur bersama berubah posisinya dari kontak normal-tertutup ke kontak normal-terbuka [RLY-13].

2.8.1 Prinsip Kerja Relay

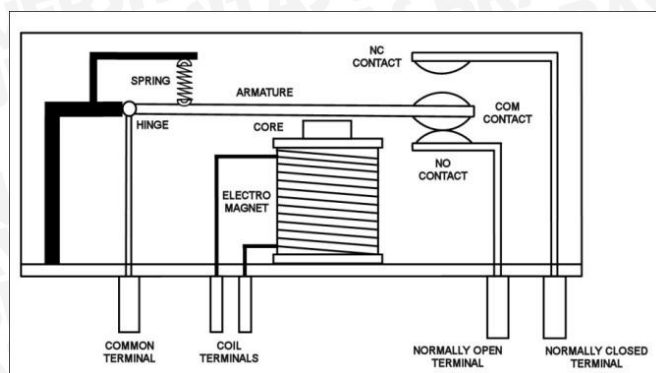
Bagian utama relay elektromagnetik adalah sebagai berikut :

1. Kumparan elektromagnet
2. Saklar atau kontaktor
3. *Swing armature*
4. *Spring* (pegas)

Relay elektromagnetik memiliki 3 kondisi saklar, ketiga posisi saklar atau kontaktor relay tersebut akan berubah ketika relay mendapatkan tegangan sumber pada elektromagnetnya. Ketiga posisi saklar tersebut adalah:

1. Posisi *Normally Open* (NO)

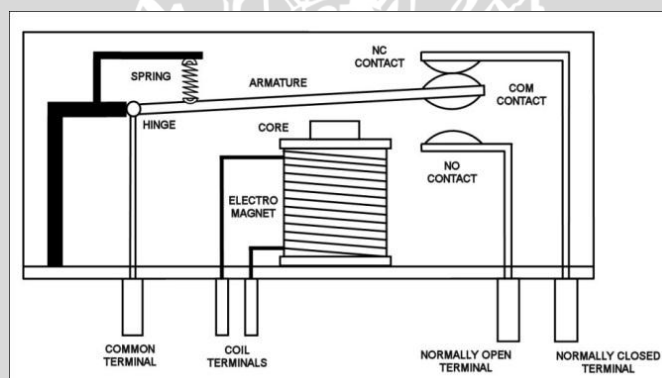
Posisi saklar relay terhubung ke terminal NO, kondisi ini akan terjadi pada saat relay mendapatkan sumber tegangan pada elektromagnetnya. Gambar 2.17 menjelaskan tentang kondisi saklar relay pada posisi NO.



Gambar 2. 17 Konsisi saklar relay pada posisi NO
Sumber: [RLY-13]

2. Posisi *Normally Close* (NC)

Pada posisi ini saklar relay terhubung ke terminal NC. Kondisi ini terjadi ketika relay tidak mendapatkan sumber tegangan pada elektromagnetnya. Gambar 2.18 menjelaskan tentang kondisi salar relay pada posisi NC.



Gambar 2. 18 Konsisi saklar relay pada posisi NC
Sumber: [RLY-13]

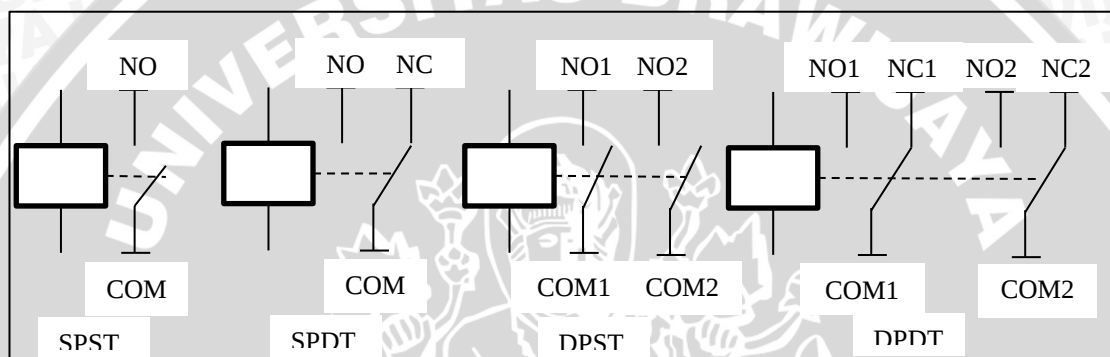
3. Posisi *Change Over* (CO)

Kondisi perubahan *armature* saklar relay yang berubah dari posisi NC ke NO atau sebaliknya dari NO ke NC. Kondisi ini terjadi ketika sumber tegangan diberikan ke elektromagnet atau ketika sumber tegangan diputus dari elektromagnetnya.

Relay memiliki tiga jenis kutub yaitu *COMMON* atau kutub acuan, *NC* (*Normally Close*), dan *NO* (*Normally Open*). Berdasarkan jumlah kutub pada relay, maka relay dibedakan menjadi 4 jenis :

1. SPST = *Single Pole Single Throw*
2. SPDT = *Single Pole Double Throw*
3. DPST = *Double Pole Single Throw*
4. DPDT = *Double Pole Double Throw*

Pole adalah jumlah *COMMON*, sedangkan *Throw* adalah jumlah terminal output (*NO* dan *NC*). Gambar 2.19 menunjukkan tentang jenis-jenis relay.



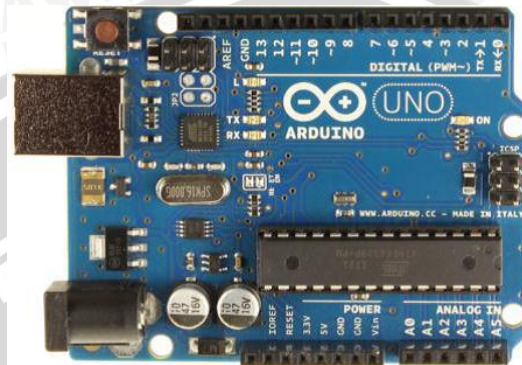
Gambar 2. 19 Jenis-jenis Relay
Sumber: [RLY-13]

2.9 Arduino

Arduino adalah kit elektronik atau papan rangkaian elektronik yang di dalamnya terdapat komponen utama yaitu sebuah *chip* mikrokontroler dengan jenis AVR dari perusahaan Atmel yang bersifat *open source*, fleksibel, dan mudah digunakan dalam hal perangkat keras dan perangkat lunak, dari sisi perangkat kerasnya arduino menggunakan prosesor Atmel AVR (Atmega8 atau Atmega168) dan memiliki pin yang dapat digunakan sebagai *input* maupun *output*. Sedangkan dari sisi perangkat lunaknya, arduino menggunakan bahasa pemrograman umum dan tersedia *boot-loader* yang berjalan pada modulnya [ARD-13].

Bahasa pemrograman Arduino adalah bahasa C tetapi bahasa ini sudah dipermudah menggunakan *library-library* sehingga pemula pun bisa mempelajarinya dengan cukup mudah. Papan Arduino UNO menggunakan mikrokontroler ATmega328P. Papan ini memiliki 14 pin *input/output digital* (enam diantaranya dapat digunakan untuk output PWM), enam buah input analog,

16 MHz *crystal oscillator*, sambungan USB, ICSP header, dan tombol *reset*. Dalam penggunaannya cukup dengan menghubungkan ke komputer dengan menggunakan kabel USB atau dengan memberikan daya menggunakan adapter AC ke DC atau dengan baterai [ARD-13]. Gambar 2.20 menunjukkan tampilan papan Arduino UNO dari atas.

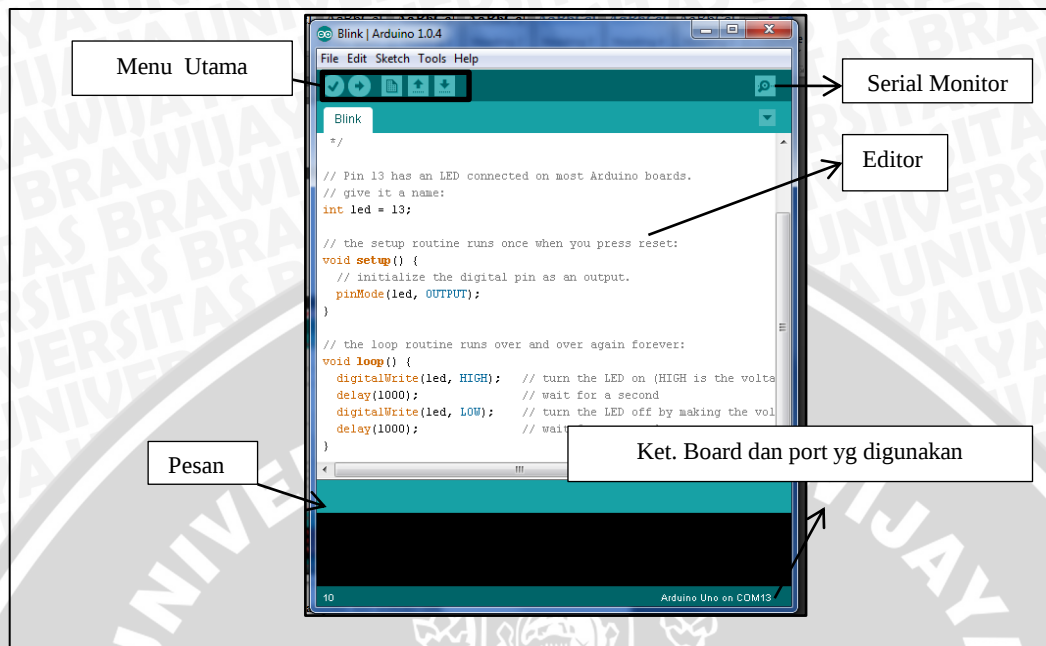


Gambar 2. 20 Tampilan Papan Arduino Uno Tampak Atas
Sumber: [ARD-13]

Arduino UNO mempunyai beberapa fasilitas untuk dapat berkomunikasi dengan komputer, Arduino lain, atau dengan mikrokontroler lain. Mikrokontroler ATmega328P pada Arduino UNO menyediakan komunikasi serial UART TTL (5V), yang tersedia pada pin 0 (RX) dan 1 (TX). Papan Arduino UNO menyalurkan komunikasi serial ini melalui USB dan dilihat sebagai *com port virtual* pada *software* di komputer. *Software Arduino Integrated Development Environment* (IDE) memiliki serial monitor yang memungkinkan data teks sederhana dikirim ke dan dari Arduino.

2.9.2 Arduino IDE

Arduino IDE adalah sebuah editor program yang digunakan untuk menulis program, mengcompile, dan mengunggah ke papan Arduino. *Arduino Integrated development environment* terdiri dari editor teks untuk menulis kode, area pesan, *toolbar* dengan tombol-tombol untuk fungsi umum, dan sederetan menu. Untuk memasukkan kode program ke papan arduino, pastikan konfigurasi jenis papan dan *port* yang digunakan pada arduino IDE. Tampilan IDE Arduino dapat dilihat pada Gambar 2.21 berikut.



Gambar 2. 21 Menunjukkan Tampilan dari Arduino IDE
Sumber: [ARD-13]

Toolbar yang terdapat pada arduino IDE terdiri dari:

1. *Verify*, digunakan untuk mengecek kesalahan pada kode program.
2. *Upload*, mengcompile kode program dan memasukkan kode ke dalam papan arduino.
3. *New*, untuk membuat kode program baru.
4. *Open*, untuk membuka file kode program yang telah disimpan sebelumnya.
5. *Save*, untuk menyimpan kode program.
6. *Serial Monitor*, untuk menyimpan kode program.

2.9.3 Pemrograman Arduino

Pada pemrograman arduino terdapat tiga bagian utama dalam bahasa pemrograman arduino, yaitu Struktur, Variabel, dan Fungsi [IAL-12].

Struktur dasar dari bahasa pemrograman arduino itu sederhana hanya terdiri dari dua bagian.

```
void setup( )
{
  //statement
}
void loop( )
{
  //statement
}
```

Gambar 2. 22 Struktur dasar Pemrograman Arduino
Sumber: [IAL-12]

Dimana *setup()* bagian untuk inisialisasi yang hanya dijalankan sekali di awal program, sedangkan *loop()* untuk mengeksekusi bagian program yang akan dijalankan berulang-ulang untuk selamanya.

➤ ***setup()***

Fungsi *setup()* hanya di panggil satu kali ketika program pertama kali di jalankan. Ini digunakan untuk pendefinisian *mode* pin atau memulai komunikasi serial. Fungsi *setup()* harus diikutsertakan dalam program walaupun tidak ada *statement* yang di jalankan.

```
void setup()
{
  pinMode(13,OUTPUT); // mengset 'pin' 13 sebagai output
}
```

Gambar 2. 23 Fungsi Setup() pada Pemrograman Arduino
Sumber: [IAL-12]

➤ ***loop()***

Setelah melakukan fungsi *setup()* maka secara langsung akan melakukan fungsi *loop()* secara berurutan dan melakukan instruksi-instruksi yang ada dalam fungsi *loop()*.

```
void loop()
{
  digitalWrite(13, HIGH); // nyalakan 'pin' 13
  delay(1000);           // pause selama 1 detik
}
```

Gambar 2. 24 Fungsi While() pada Pemrograman Arduino
Sumber: [IAL-12]

➤ ***{ } curly braces***

Curly brace mendefinisikan awal dan akhir dari sebuah blok fungsi. Baik blok void *setup()*, void *loop()* maupun blok fungsi harus diberi tanda kurung kurawal buka “{” sebagai tanda awal program di blok tersebut dan kurung

kurawal tutup “}” sebagai tanda akhir program. Tanda kurung kurawal buka dan tutup digunakan pada blok kontrol program, seperti if, if-else, for-loop, while-loop dan do-while-loop. Apabila ketika memprogram dan programmer lupa memberi *curly brace* tutup maka ketika di compile akan terdapat laporan *error* .

➤ **; semicolon**

Semicolon harus di berikan pada setiap *statement* program yang kita buat ini merupakan pembatas setiap *statement* program yang di buat.

➤ **/*...*/ blok comment**

Semua *statement* yang di tulis dalam *block comments* tidak akan dieksekusi dan tidak akan di *compile* sehingga tidak mempengaruhi besar program yang di buat untuk dimasukan dalam papan arduino.

➤ **// line comment**

Sama halnya dengan *block comments*, *line comments* pun sama hanya saja yang dijadikan komen adalah tiap baris.

➤ **Variable**

Variable adalah sebuah penyimpan nilai yang dapat digunakan dalam program. *Variable* dapat dirubah sesuai dengan instruksi yang kita buat. Ketika mendeklarasikan *variable* harus diikutsertakan tipe *variable* serta nilai awal *variable*.

```
Type variableName = 0;
```

Gambar 2. 25 Pendeklarasian *Variable* pada Pemrograman Arduino

Sumber: [IAL-12]

Contoh:

```
IntinputVariable = 0; // mendefinisikan sebuah
variable bernama inputVariable dengan nilai awal 0

inputVariable = analogRead(2); // menyimpan nilai
yang ada di analog pin 2 ke input Variable
```

Gambar 2. 26 Contoh Pendeklarasian *Variable* pada Pemrograman Arduino

Sumber: [IAL-12]

➤ **Variable scope**

Sebuah *variable* dapat dideklarasikan pada awal program sebelum void setup(), secara *local* di dalam sebuah *function*, dan terkadang di dalam sebuah

block statement pengulangan. Sebuah *variabel* global hanya satu dan dapat digunakan pada semua blok *function* dan *statement* di dalam program. *Variabel* global dideklarasikan pada awal program sebelum fungsi *setup()*. Sebuah *variabel* lokal dideklarasikan di setiap *block function* atau di setiap *block statement* pengulangan dan hanya dapat digunakan pada *block* yang bersangkutan saja.

```
void setup()
{
}
void loop()
{
  for (int i=0; i<20;) {
    i++;
  }
  float f;
}
```

Gambar 2. 27 Contoh Penggunaan Variable Scope pada Pemrograman Arduino
Sumber: [IAL-12]

➤ Function

Function (fungsi) adalah blok pemrograman yang mempunyai nama dan mempunyai *statement* yang akan di eksekusi ketika *function* di panggil.

Cara pendeklarasian *function*

```
type functionName(parameters) {
  // Statement;
}
```

Gambar 2. 28 Pendeklarasian Fungsi pada Pemrograman Arduino
Sumber: [IAL-12]

Contoh :

```
intdelayVal()
{
  int v; // membuat variable 'v' bertipe integer
  v = analogRead(pot); // bacaharga potentiometer
  v /= 4; // konversi 0-1023 ke 0-255
  return v; // return nilai v
}
```

Gambar 2. 29 Contoh Pendeklarasian Fungsi pada Pemrograman Arduino
Sumber: [IAL-12]

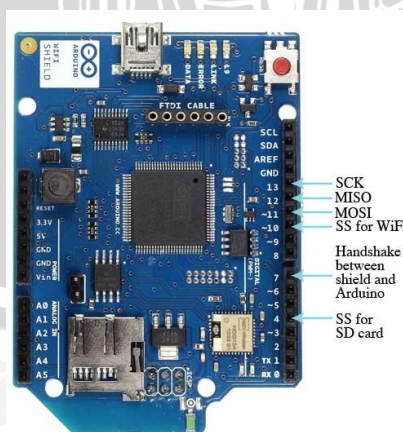
Pada contoh diatas fungsi tersebut memiliki nilai balik *int* (integer), karena jika tidak menghendaki adanya nilai balik maka tipe *function* harus *void*.

2.10 Modul Arduino WiFi

Dengan adanya modul WiFi memungkinkan sebuah papan arduino dapat terhubung ke internet menggunakan protokol standar 802.11 (WiFi). Sebuah Atmega 32UC3 menyediakan jaringan (IP *stack*) yang mampu menyediakan komunikasi TCP dan UDP. Untuk menggunakan WiFi *shield* pada penulisan *sketch* atau kode pada arduino IDE dibutuhkan *library* WiFi agar WiFi *shield* dapat dikenali oleh papan arduino. WiFi *shield* terhubung dengan papan Arduino menggunakan *header* dengan kaki yang panjang yang menembus *shield*. Hal ini untuk menjaga *layout* pin dan memungkinkan *shield* lain untuk ditumpukkan di atasnya [ARD-13].

WiFi *shield* dapat terhubung ke jaringan nirkabel dengan ketentuan harus sesuai dengan spesifikasi operasi pada protokol 802.11b dan 802.11g. Pada *shiled* tersebut terdapat *slot* kartu micro-SD yang dapat digunakan sebagai tempat penyimpanan (*storage*). *Shield* ini cocok untuk papan arduino UNO dan Mega, untuk mengoperasikan *slot* micro-SD digunakan *library* SD Card.

Arduino berkomunikasi dengan prosesor WiFi *shield* dan SD Card menggunakan bus SPI (termasuk *header* ICSP). Pin digital yang digunakan adalah pin 11,12, dan 13 untuk papan arduino UNO dan pin 50, 51, dan 52 untuk papan arduino Mega. Pada kedua papan tersebut pin 10 digunakan untuk HDG104 dan pin 4 untuk SD card. Pin 7 digunakan dalam proses *handshake* pin antara WiFi *shield* dan Arduino. Gambar 2.30 menjelaskan tentang pin pada WiFi *shield*.

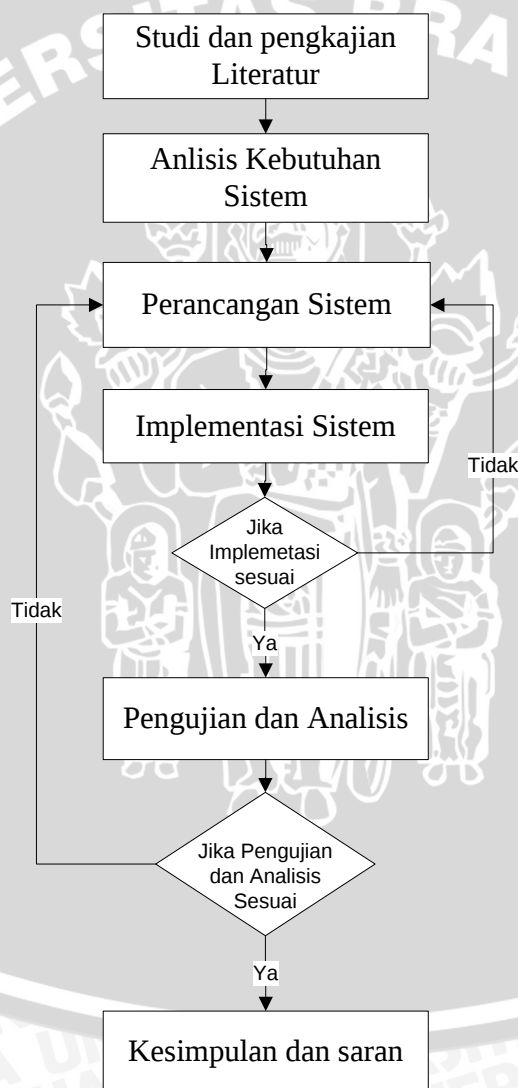


Gambar 2. 30 Pin pada WiFi Shield
Sumber: [ARD-13]

BAB III

METODOLOGI PENELITIAN

Bab ini membahas langkah-langkah yang akan dilakukan dalam penyusunan skripsi yang meliputi : perancangan, implementasi dan pengujian. Kesimpulan dan saran disertakan sebagai catatan dan kemungkinan ke arah pengembangan selanjutnya.



Gambar 3. 1 Diagram blok metodologi penelitian
Sumber: [Perancangan]

3.1 Studi Literatur

Studi literatur mempelajari mengenai penjelasan dasar teori yang digunakan untuk menunjang penulisan skripsi. Teori-teori pendukung tersebut diperoleh dari buku, artikel, jurnal, *e-book*, dan dokumentasi *project*. Teori-teori pendukung tersebut meliputi:

1. *Embedded System* (Sistem Tertanam)
2. *Controll System* (Sistem Kendali)
3. *Wireless sensor network*
4. Protokol komunikasi sistem *Wireless sensor network*
 - a. Model OSI layer
 - b. Arsitektur jaringan TCP/IP
 - c. Internet Protocol
 - d. Alamat *broadcast*
 - e. Subnet mask
5. Arduino
 - a. Pengenalan arduino
 - b. Arduino IDE
 - c. Pemrograman arduino
6. Sensor DHT11
7. Sensor SEN0114
8. Relay
9. Modul WiFi *shield*

3.2 Analisis Kebutuhan

Analisis kebutuhan bertujuan untuk mendapatkan semua kebutuhan yang diperlukan oleh sistem yang akan dibangun. Analisis kebutuhan ini disesuaikan dengan kebutuhan yang peneliti gunakan, kebutuhan tersebut meliputi :

3.2.1 Analisis kebutuhan *Input*

Input yang digunakan pada sistem pengendalian suhu dan kelembaban berbasis *wireless embedded system* sebagai berikut :

1. *Input* Sensor DHT11 adalah suhu udara di sekitar sensor

2. *Input* Sensor SEN0114 adalah kelembaban tanah di sekitar sensor
3. *Input* Arduino UNO R3 adalah data analog yang berasal dari sensor SEN0114 dan data digital suhu udara (DHT11).
4. *Input* WiFi *shield* adalah data digital berasal dari Arduino UNO R3.
5. *Input* Relay adalah data digital berasal dari Arduino UNO R3.

3.2.2 Analisis Kebutuhan Output

Output yang digunakan pada sistem pengendalian suhu dan kelembaban berbasis *wireless embedded system* sebagai berikut :

1. *Output* Sensor DHT11 adalah data digital untuk Arduino UNO R3.
2. *Output* Sensor SEN0114 adalah data analog untuk Arduino UNO R3.
3. *Output* Arduino UNO R3 adalah data digital untuk WiFi *Shield* dan Relay.
4. *Output* WiFi *shield* adalah paket data internet.

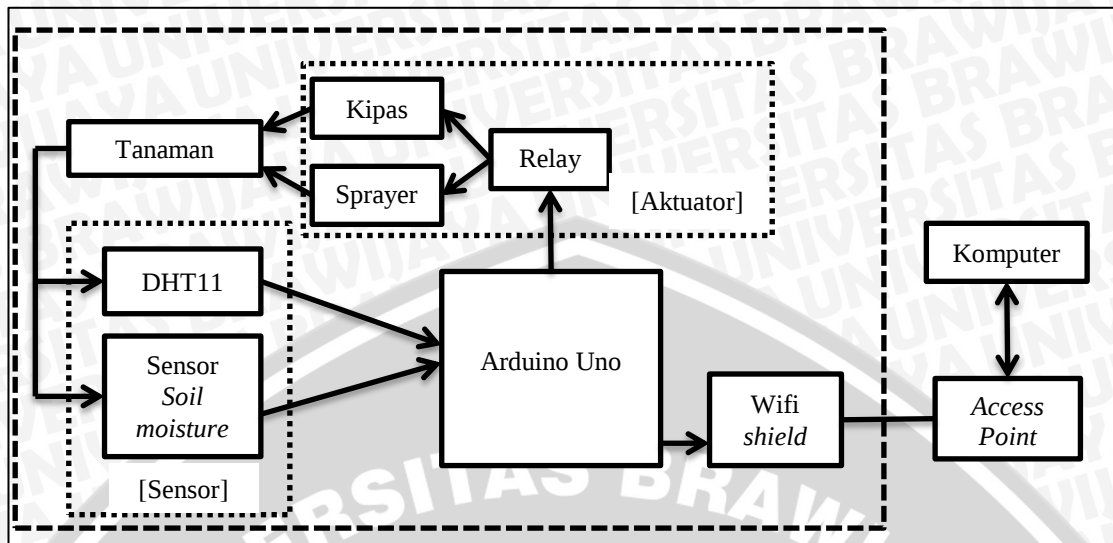
3.2.3 Analisis Fungsi dan Kinerja

Fungsi dan kinerja pada sistem pengendalian suhu dan kelembaban berbasis *wireless embedded system* adalah sebagai berikut:

1. Membaca *input* berupa suhu udara dan kelembaban tanah.
2. Memberikan tindakan dengan mengaktifkan aktuator sesuai dengan kondisi yang telah diatur sebelumnya.
3. Memberikan memberikan informasi tentang suhu udara dan kelembaban tanah melalui *web* secara *relatime* dan berkelanjutan.

3.3 Perancangan

Perancangan aplikasi dilakukan setelah semua kebutuhan sistem diperoleh melalui tahap analisis kebutuhan. Perancangan sistem digambarkan pada diagram blok Gambar 3.2 berikut ini:



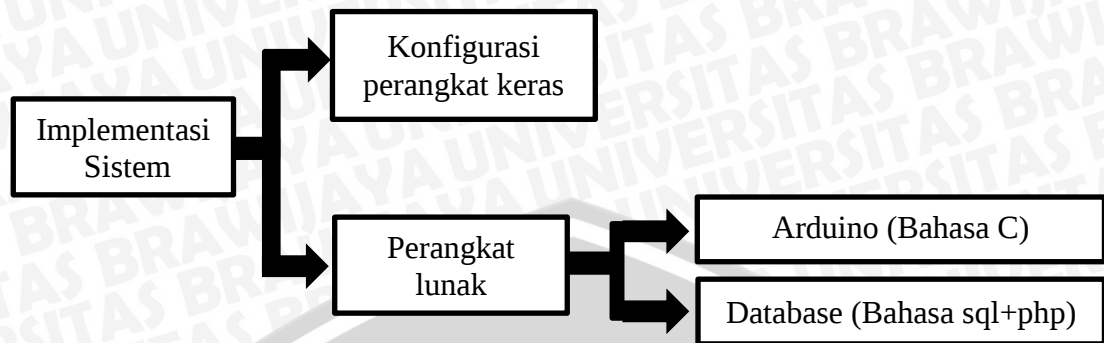
Gambar 3. 2 Diagram blok perancangan sistem

Sumber: [Perancangan]

Pada diagram blok diatas, sensor suhu udara (DHT11) dan kelembaban tanah (sensor SEN0114) berfungsi untuk menerima *input* dari luar yaitu suhu udara dan kelembaban tanah disekitar sensor. Data inputan tersebut diproses oleh arduino untuk digunakan sebagai nilai acuan untuk melakukan kontrol pada aktuator kipas dan sprayer yang diaktifkan oleh relay sebagai saklar elektrik, data suhu ruangan dan kelembaban tanah akan dikirim ke komputer melalui jaringan nirkabel secara *realtime* dan berkelanjutan.

3.4 Implementasi

Implementasi sistem dilakukan dengan mengacu kepada perancangan sistem. Tahapan implementasi dibagi menjadi 2 yaitu implementasi perangkat keras dan perangkat lunak. Pada tahapan implementasi perangkat keras hal yang dilakukan adalah mengkonfigurasi perangkat keras pada *node* sensor yang meliputi arduino, WiFi *shield*, sensor, aktuator, dan relay. Pada tahapan implementasi perangkat lunak pada sisi *node* sensor meliputi pemrograman arduino dengan bahasa c. Pada sisi data *sink* meliputi pemrograman database menggunakan bahasa sql dan untuk terhubung dengan database diperlukan aplikasi yang dibuat dengan menggunakan bahasa php. Tahapan implementasi sistem digambarkan pada diagram blok Gambar 3.3 berikut ini:



Gambar 3. 3 Diagram blok tahapan implementasi sistem
Sumber: [Perancangan]

3.5 Pengujian dan Analisis

Pengujian sistem pada penelitian ini dilakukan agar dapat menunjukkan bahwa sistem telah mampu bekerja sesuai dengan spesifikasi dari kebutuhan yang melandasinya. Pengujian yang dilakukan meliputi:

- Pengujian sensor suhu dan kelembaban tanah
- Pengujian komunikasi WiFi antara arduino dengan komputer
- Pengujian relay
- Pengujian sistem secara keseluruhan

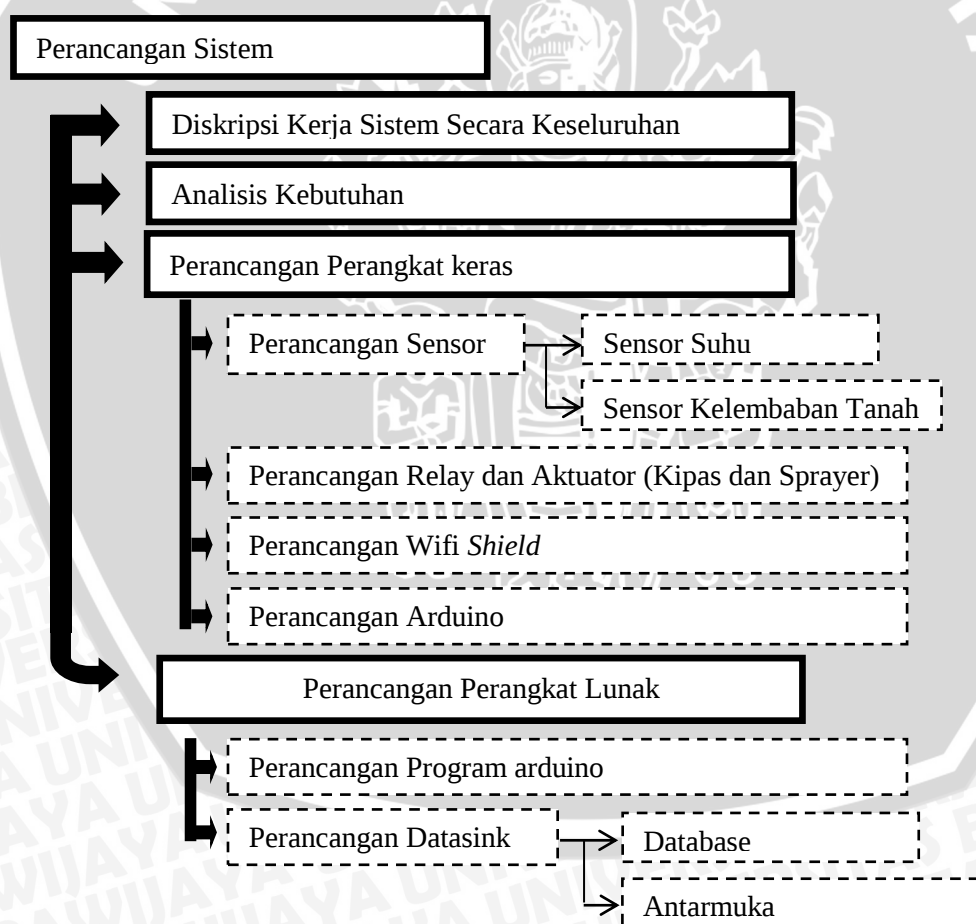
3.6 Kesimpulan dan Saran

Pengambilan kesimpulan dilakukan setelah semua tahapan perancangan, dan implementasi sistem telah selesai dilakukan. Kesimpulan diambil dari analisis terhadap rancangan yang dibangun. Tahap terakhir dari penulisan adalah saran yang dimaksudkan untuk memperbaiki kesalahan-kesalahan yang terjadi dan menyempurnakan penulisan serta untuk memberikan pertimbangan atas pengembangan sistem selanjutnya.

BAB IV

PERANCANGAN SISTEM

Perancangan dalam pembuatan sistem ini meliputi dua bagian, yaitu perancangan perangkat keras pada *node* sensor dan perancangan perangkat lunak. Pada perancangan perangkat keras meliputi perancangan arduino, WiFi *shield*, sensor suhu dan kelembaban tanah, aktuator dan relay. Sedangkan perancangan perangkat lunak meliputi pemrograman arduino, dan database. Perancangan serta pembuatan sistem dilakukan secara bertahap untuk memudahkan proses analisa sistem. Perancangan sistem ini diawali dengan pembuatan blok diagram dan prinsip kerja secara keseluruhan. Berikut pohon analisa dan perancangan sistem dijelaskan pada Gambar 4.1

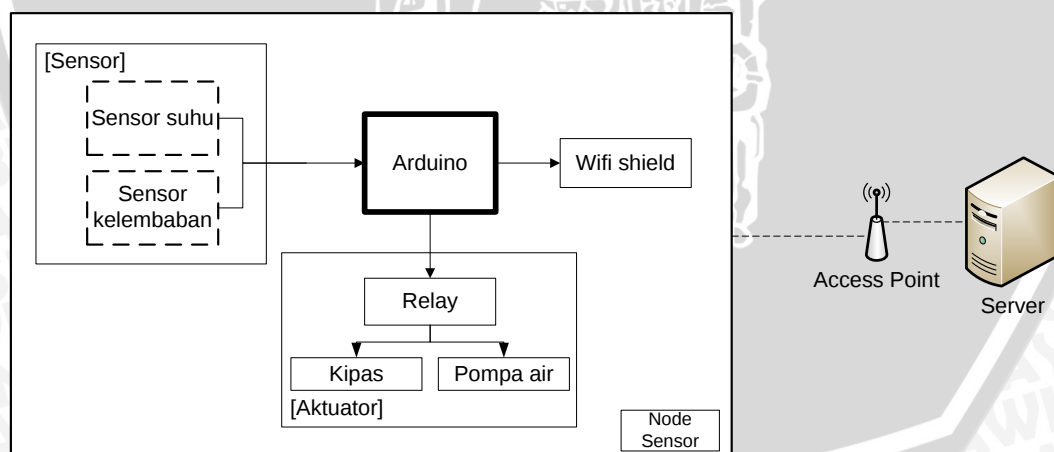


Gambar 4. 1 Pohon Perancangan Sistem
Sumber : [perancangan]

4.1 Diskripsi Kerja Sistem Secara Keseluruhan

Sistem pengendalian suhu dan kelembaban tanah merupakan sebuah jaringan sensor yang terhubung secara *wireless* untuk proses monitoring perubahan suhu udara dan kelembaban tanah. Sensor suhu udara dan kelembaban tanah sebagai inputan akan diambil dan diproses oleh mikrokontroler untuk dijadikan sebagai nilai acuan untuk mengambil tindakan. Sistem ini dapat mendeteksi perubahan suhu udara dan kelembaban tanah yang akan memicu aktuator untuk melakukan tindakan. Apabila nilai yang dibaca oleh sensor suhu melebihi batas maka mikrokontroler akan mengirimkan sinyal ke relay untuk mengaktifkan aktuator berupa kipas angin. Apabila nilai yang dibaca oleh kelembaban tanah kurang dari batas maka mikrokontroler akan mengirimkan sinyal ke relay untuk mengaktifkan aktuator berupa pompa air. Data-data sensor akan dikirim dan disimpan pada database *server* melalui jaringan nirkabel secara *realtime* dan berkelanjutan.

Pemantauan dapat dilihat secara *realtime* melalui *website* yang memberikan informasi mengenai status suhu ruangan, kelembaban tanah dan status kedua aktuator. Diskripsi kerja sistem ini dapat digambarkan dalam diagram blok seperti Gambar 4.2.



Gambar 4. 2 Diagram blok kerja sistem
Sumber : [perancangan]

4.2 Analisis Kebutuhan

Sistem pengendalian suhu dan kelembaban berbasis *wireless embedded system* dirancang dengan memanfaatkan teknologi *embedded* untuk melakukan tugas tertentu. Tugas tersebut meliputi proses pengambilan data fisis lingkungan pantau, kendali otomatis, pengiriman data, hingga memprosesnya untuk menghasilkan data yang dikehendaki. Berdasarkan diskripsi kerja sistem pengendalian suhu dan kelembaban, analisis kebutuhan meliputi analisis kebutuhan perangkat keras dan perangkat lunak.

4.2.1 Analisis Kebutuhan Perangkat Keras

Perangkat keras pada sistem pengendalian suhu dan kelembaban sebagai pengolah data dari *input* dan menghasilkan *output*. Beberapa perangkat keras yang dibutuhkan oleh sistem pengendalian suhu dan kelembaban berbasis *wireless embedded system* adalah sebagai berikut:

1. Mikrokontroler Arduino Uno R3
2. WiFi Shield
3. Sensor suhu udara (DHT11) dan kelembaban tanah (sensor SEN0114)
4. Relay
5. Kabel
6. Kipas
7. Sprayer atau pompa air

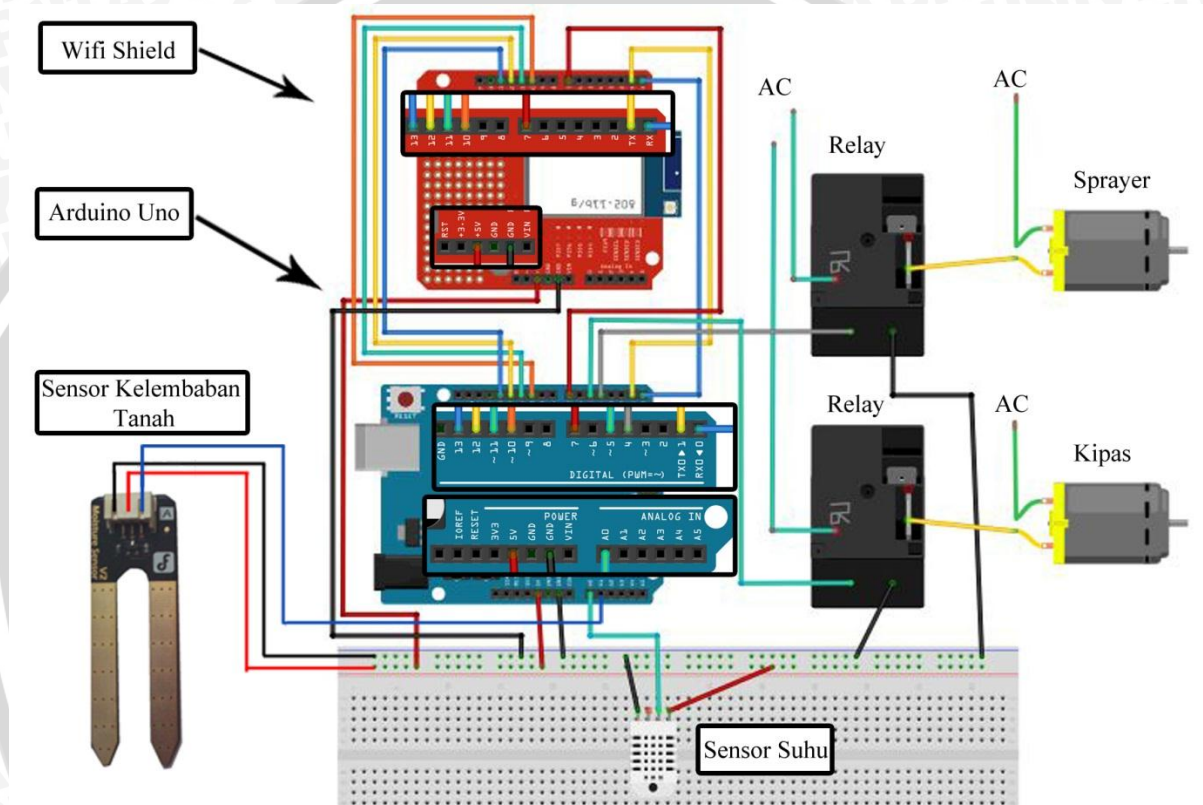
4.2.2 Analisis Kebutuhan Perangkat Lunak

Selain perangkat keras, sistem pengendali suhu dan kelembaban tanah membutuhkan perangkat lunak untuk mendukung kerja perangkat keras sebagai berikut:

1. Sistem Operasi, sistem operasi digunakan untuk menjalankan perangkat lunak. Sistem operasi yang digunakan adalah Windows 7 *professional*.
2. Arduino 1.0.4, sebuah perangkat lunak yang digunakan untuk membuat program yang dibutuhkan pada sistem pengendali suhu dan kelembaban tanah.

3. Xampp, Fungsinya adalah sebagai server yang berdiri sendiri (localhost), yang terdiri atas program Apache HTTP Server, MySQL database, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl.

4.3 Perancangan Perangkat *Node Sensor*



Gambar 4. 3 Perancangan perangkat keras *node sensor*
Sumber : [perancangan]

Dari Gambar 4.3 dapat ditunjukkan bahwa rangkaian terdiri dari beberapa bagian, yaitu sensor suhu dan kelembaban, arduino, modul WiFi dan relay. Sensor kelembaban tanah memiliki 3 pin yang terdiri dari pin *ground*, *vcc*, dan *data*, untuk pin *data* terhubung dengan pin A1 pada arduino, penjelasan lebih lanjut ditunjukkan pada subbab 4.3.1. Sensor suhu memiliki 4 pin namun yang digunakan hanya 3 pin yaitu pin *ground*, *vcc*, dan *data*, untuk pin *data* terhubung dengan pin A0 pada arduino, penjelasan lebih lanjut ditunjukkan pada subbab 4.3.2. Arduino sebagai pusat pemrosesan data dan kendali terhubung dengan semua bagian. Untuk proses pengiriman data sensor ke *server* melalui jaringan

nirkabel menggunakan modul WiFi. Modul WiFi terhubung dengan arduino menggunakan beberapa pin yaitu pin 0, 1, 7, 10, 11, 12, dan 13, penjelasan lebih lanjut ditunjukkan pada subbab 4.3.4. Relay sebagai saklar elektronik akan aktif apabila dipicu dengan arus kecil dari mikrokontroler, relay yang digunakan untuk mengaktifkan sprayer akan terhubung dengan pin D4 pada arduino. Sedangkan relay yang digunakan untuk mengaktifkan kipas terhubung dengan pin D5 pada arduino, penjelasan lebih lanjut ditunjukkan pada subbab 4.3.1. Penjelasan mengenai konfigurasi pin terdapat pada Tabel 4.1.

Tabel 4. 1 Konfigurasi pin

Pin Arduino	Pin Sensor kelembaban tanah	Pin Sensor suhu	Pin Modul WiFi	Pin Relay kipas	Pin Relay sprayer
A0		2 (Data)			
A1	1 (Data)				
Ground	2 (Ground)	4 (Ground)	Ground	Ground	Ground
5v	3 (5v)	1 (5v)	5v	5v	5v
D0			D0		
D1			D1		
D4					DATA 1
D5				DATA 2	
D7			D7		
D10			D10		
D11			D11		
D12			D12		
D13			D13		

Sumber : [perancangan]

4.3.1 Perancangan Sistem pembacaan tingkat kelembaban tanah dengan menggunakan sensor SEN0114

Sensor kelembaban tanah yang digunakan dalam penelitian ini adalah sensor SEN0114 dfrobot. Sensor ini memiliki spesifikasi sebagai berikut [SSM-13]:

1. Beroperasi pada *input* tegangan 3.3v atau 5v
2. Output tegangan 0~4.2v
3. Arus listrik 35mA
4. Memiliki 3 pin,yaitu:

- Analog output
- Ground
- Tegangan daya

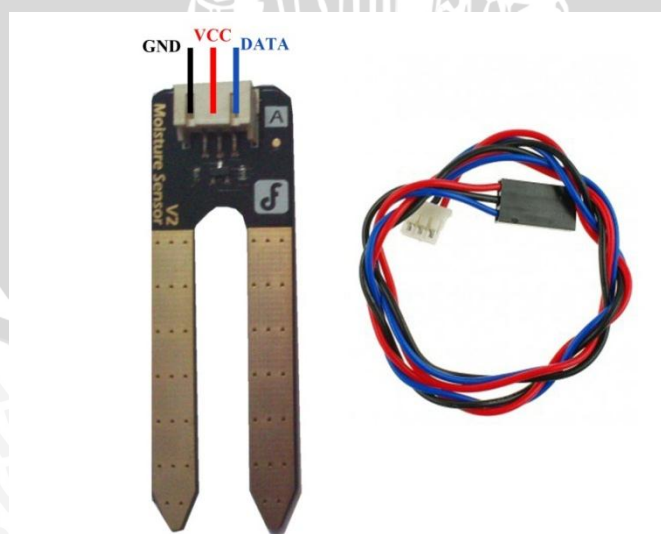
5. Dimensi 60x20x5mm

6. Nilai jangkauan :

1. 0~300 : tanah kering
2. 300~700 : tanah lembab
3. 700~950 : di dalam air

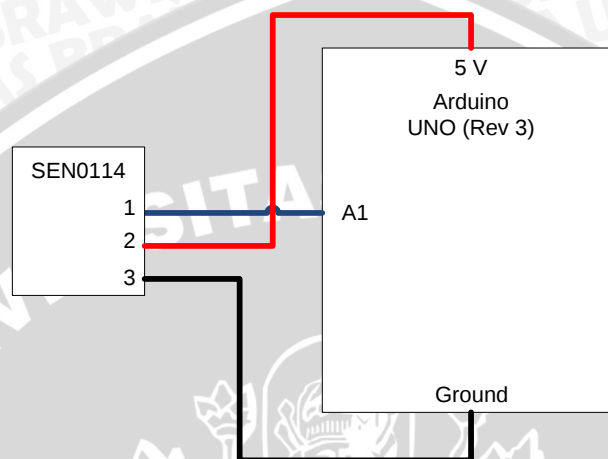
Sebelum menggunakan sensor SEN0114 untuk keperluan pengukuran kelembaban tanah, sensor ini perlu dikalibrasikan terlebih dahulu agar memudahkan dalam penggunaannya. Pengkalibrasian sensor dilakukan dengan mencari nilai terendah dan tertinggi sebagai pembanding untuk mencari persentase kelembaban tanah. Semakin banyak kadar air maka semakin tinggi nilai sensor yang dihasilkan, begitu juga sebaliknya. Sensor ini akan menghasilkan nilai tertinggi ketika media disekitar sensor dalam keadaan basah. Sedangkan sensor akan menghasilkan nilai terendah ketika sensor berada di udara bebas atau disekitar sensor tidak ada media yang dapat menghantarkan listrik dalam hal ini ialah air atau tanah.

Perangkat sensor kelembaban tanah yang digunakan dalam penelitian ini dapat dilihat pada Gambar 4.4.



Gambar 4. 4 Sensor SEN0114
Sumber: [perancangan]

Pada perancangan sistem ini sensor SEN0114 dihubungkan ke arduino dengan konfigurasi pin data disambungkan ke pin A1 pada arduino, pin *ground* dihubungkan ke pin *ground* pada arduino dan pin vcc dihubungkan ke pin 5v pada arduino. Skema hubungan antara arduino dengan sensor SEN0114 ditunjukkan pada Gambar 4.5.



Gambar 4. 5 Skema hubungan arduino dengan sensor SEN0114
Sumber: [perancangan]

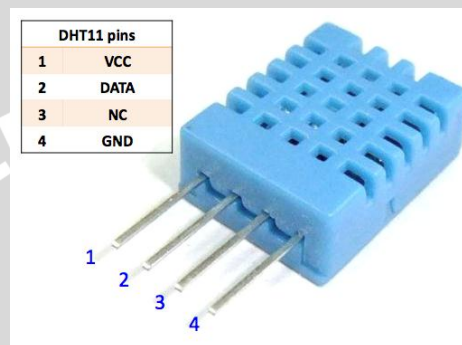
4.3.2 Perancangan Sistem pembacaan suhu ruangan dengan menggunakan sensor DHT11

Sensor yang digunakan dalam penelitian ini adalah sensor DHT11. Perangkat sensor dengan tipe *temperature and humidity sensor* didesain untuk mendeteksi tingkat suhu dan kelembaban udara yang ada disekitar. Sensor ini mengeluarkan sinyal digital sehingga data keluaran sensor sudah menunjukkan suhu yang sebenarnya. Sensor ini sangat mudah digunakan bersama dengan arduino. Memiliki tingkat stabilitas yang sangat baik serta fitur kalibrasi yang akurat. Koefisien kalibrasi disimpan dalam OTP program memori, sehingga ketika internal sensor mendeteksi sesuatu, maka modul ini menyertakan koefisien tersebut dalam kalkulasinya. Sensor DHT11 memiliki spesifikasi sebagai berikut [DSD-13]:

1. Beroperasi pada input tegangan 3-5.5v
2. Arus listrik 0.2-1mA
3. Waktu pengambilan data minimal 1 detik
4. Nilai jangkauan suhu 0-50°C, akurasi $\pm 1 \sim 2$ °C, resolusi 1 °C

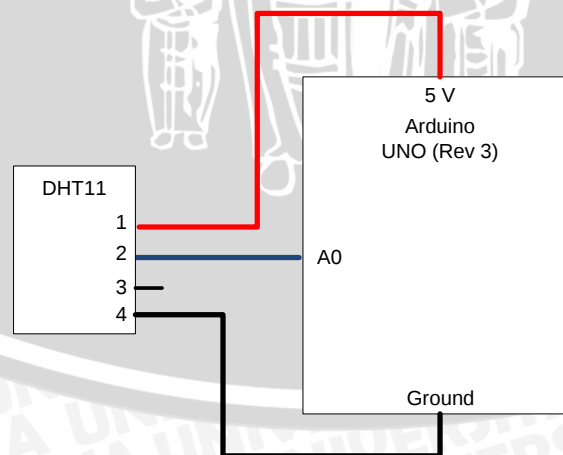
5. Nilai jangkauan kelembaban 20-90%RH, akurasi $\pm 5\%$ RH
6. Memiliki 4 pin, yaitu:
 1. Vdd
 2. Data
 3. Null

Perangkat sensor DHT11 yang digunakan dalam penelitian ini dapat dilihat pada Gambar 4.6.



Gambar 4. 6 Sensor DHT11
Sumber: [DSD-13]

Pada penelitian ini sensor dihubungkan ke arduino dengan menggunakan 3 pin. Pin vcc dihubungkan dengan pin 5v pada arduino, pin *ground* dihubungkan dengan *ground* pada arduino dan pin data dihubungkan ke pin A0 pada arduino. Skema hubungan arduino dengan sensor DHT11 ditunjukkan pada Gambar 4.7.



Gambar 4. 7 Skema hubungan arduino dengan sensor DHT11
Sumber: [perancangan]

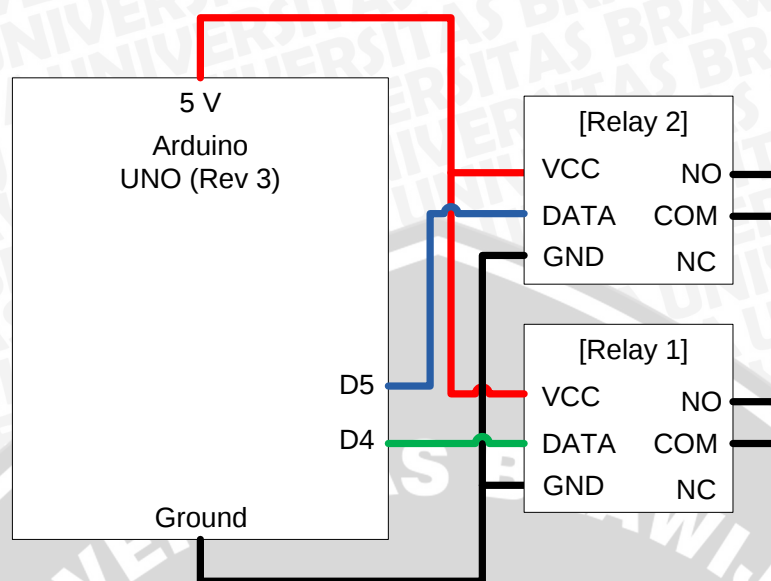
4.3.3 Perancangan Sistem relay aktuator

Perangkat relay yang digunakan pada penelitian ini adalah jenis relay SPDT dengan spesifikasi sebagai berikut:

1. Input tegangan 5v DC
2. Output tegangan maksimal 250v AC
3. Terdiri dari 2 relay
4. Port meliputi NC, NO dan *main terminal*
5. Terdapat 2 led untuk mengetahui status relay

Penggunaan relay pada penelitian ini digunakan sebagai saklar untuk menghidupkan atau mematikan aktuator kipas dan sprayer. Untuk mengendalikan relay aktuator sprayer, pin inputan pada relay di hubungkan pada pin D4 arduino. Sedangkan untuk mengendalikan relay aktuator kipas, pin inputan pada relay di hubungkan pada pin D5 arduino. Sedangkan pin *ground* dan vcc pada kedua relay tersebut di hubungkan pada pin *ground* dan 5v arduino. Pada relay aktuator sambungan tegangan AC di hubungkan ke terminal NO dan sambungan satunya di hubungkan ke terminal kontak inti.

Arduino sebagai pusat kendali dan pengolahan data akan menyeleksi kondisi dengan inputan parameter suhu pada DHT11 dan kelembaban tanah. Pada saat kondisi kelembaban kurang dari batas maka arduino akan mengaktifkan pin D4 menjadi *high* mengakibatkan relay pada aktuator sprayer menyala, sehingga relay akan berubah posisi dari NC ke NO. Apabila kondisi suhu melebihi batas maka arduino akan mengaktifkan pin D5 menjadi *high* mengakibatkan relay pada aktuator kipas menyala, sehingga relay akan berubah posisi dari NC ke NO. Relay sebagai saklar elektrik diaktifkan dengan tegangan yang kecil untuk menggerakkan *armature* yang mengakibatkan perubahan keadaan NC ke NO atau sebaliknya. Gambar 4.8 menjelaskan tentang hubungan antara arduino dengan kedua relay aktuator.



Gambar 4. 8 Skema hubungan arduino dengan relay
Sumber: [perancangan]

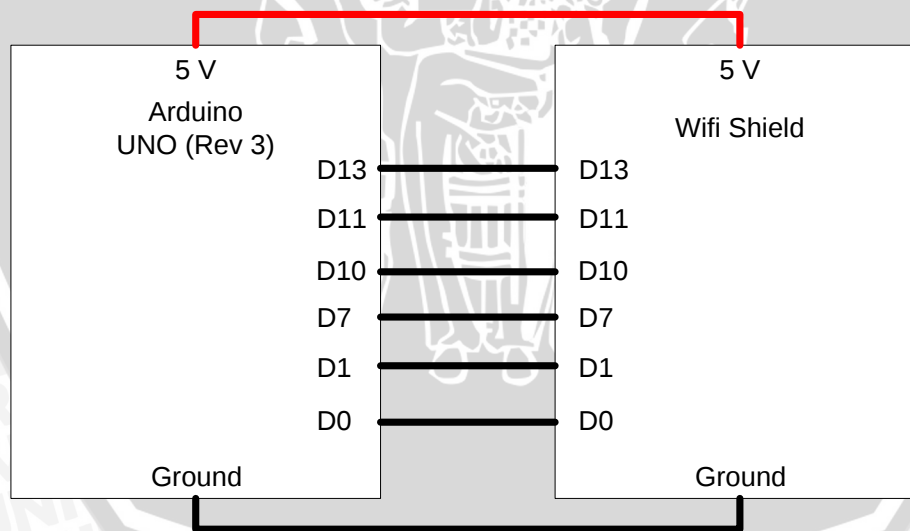
4.3.4 Perancangan Sistem pengiriman data secara *wireless* menggunakan arduino WiFi shield

Untuk melakukan proses pengiriman data pembacaan sensor didesain dengan menggunakan perangkat arduino WiFi shield. Pada penelitian ini perangkat arduino WiFi shield digunakan sebagai sebuah perangkat jaringan yang mentransmisikan data-data sensor dari *node* sensor ke *server*. Perangkat ini dipasang pada arduino dengan cara menghubungkan pin-pin antara arduino WiFi shield dengan mikrokontroler arduino, pemasangan WiFi shield dilakukan dengan cara menumpukkan arduino WiFi shield diatas arduino. Penjelasan mengenai pemasangan arduino WiFi shield dengan mikrokontroler arduino ditunjukkan pada Gambar 4.9.



Gambar 4. 9 Pemasangan WiFi shield dengan arduino
Sumber: [perancangan]

Arduino WiFi *shield* dilengkapi dengan *processor* Atmega 32UC3 yang menyediakan jaringan (*IP stack*) untuk komunikasi melalui jaringan internet. Untuk menggunakan perangkat WiFi *shield* pada arduino dibutuhkan *library* WiFi pada penulisan *sketch* atau kode pada arduino IDE. WiFi *shield* terhubung dengan papan arduino menggunakan *header* dengan kaki yang panjang yang menembus *shield*. Hal Ini untuk menjaga *layout* pin dan memungkinkan *shield* lain untuk ditumpukkan di atasnya. Pada penelitian ini perangkat lain seperti sensor dan aktuator yang terhubung dengan arduino dengan cara menghubungkan pada pin arduino WiFi *shield* yang terpasang tepat di atas arduino. Pin yang digunakan pada arduino WiFi *shield* untuk terhubung dengan arduino pada penelitian ini yaitu pin D11, D12, dan D13 yang digunakan sebagai SPI *bus*. Pin D10 digunakan untuk HDG104 dan pin D4 digunakan untuk SD *card*, namun pada penelitian ini modul SD *card* yang ada pada arduino WiFi *shield* tidak digunakan sehingga tidak perlu didefinisikan pin D4. Pin0 dan 1 merupakan pin yang digunakan untuk komunikasi serial. Penejelasan mengenai hubungan antara mikrokontroler arduino dengan arduino WiFi *shield* ditunjukkan pada Gambar 4.11.



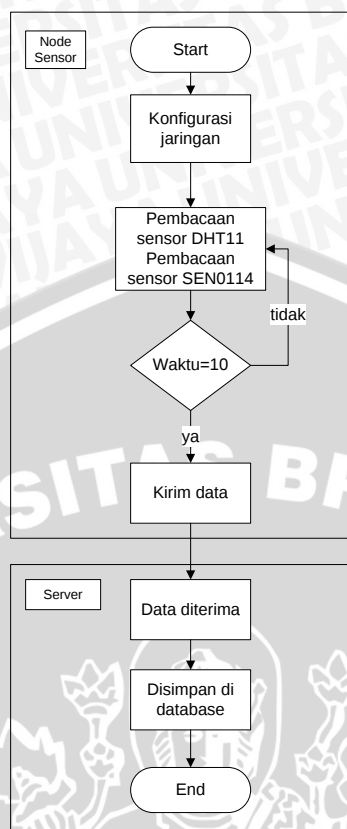
Gambar 4. 10 Skema hubungan antara WiFi *shield* dengan arduino
Sumber: [perancangan]

Spesifikasi perangkat arduino WiFi *shield* yang digunakan pada penelitian ini sebagai berikut [DSW-13]:

1. Beroperasi pada tegangan 5v (tersuplai dari papan arduino)
2. Terkoneksi dengan jaringan *wireless* 802.11b/g
3. Jenis enkripsi : WEP dan WPA2 *Personal*
4. Terkoneksi dengan arduino melalui *port* SPI
5. ICSP *headers*
6. Termasuk *micro* SD *slot*
7. Koneksi FTDI untuk serial *debugging* WiFi *shield*
8. Mini-USB untuk *update firmware* WiFi *shield*
9. Memiliki beberapa lampu led yang memberikan informasi sebagai berikut:
 - L9 (Kuning) : Led untuk pin 9
 - *LINK* (Hijau) : Mengindikasikan terhubung pada jaringan
 - *ERROR* (Merah) : Mengindikasikan bahwa terjadi gagal hubungan terhadap jaringan
 - *DATA* (Biru) : Mengindikasikan transfer data

Pada penelitian ini *node* sensor yang dilengkapi dengan WiFi *shield* terhubung dengan *access point* dengan mendaftarkan nama SSID *access point* pada *sketch* atau kode program. Setelah *node* sensor ini mengetahui nama SSID *access point*, *node* sensor akan mendapatkan IP *address* secara DHCP. Penggunaan metode DHCP dalam menentukan IP *address* memudahkan dalam pengalokasian IP *address* karena tidak perlu memberikan IP *address* *node* sensor secara manual.

Untuk memenuhi kebutuhan transfer data, *node* sensor ini dijadikan sebagai *client* yang akan mengirimkan data sensor suhu ruangan dan kelembaban tanah ke *server*. Dalam penelitian ini *server* tersebut adalah komputer yang terhubung dalam satu jaringan dengan *node* sensor. Pengiriman data sensor dilakukan setiap periode waktu 10 detik, data-data hasil pemantauan sensor terhadap suhu ruangan dan kelembaban tanah akan disimpan kedalam database. Penjelasan mengenai proses pengiriman data sensor ditunjukkan pada Gambar 4.11.



Gambar 4. 11 Proses pengiriman data dari node sensor ke server
Sumber: [perancangan]

4.3.5 Perancangan Sistem pengolahan data

Untuk melakukan proses pengambilan dan pengolahan data pembacaan sensor didesain dengan menggunakan perangkat mikrokontroler Arduino Uno R3. Arduino UNO merupakan mikrokontroler *plug and play*, sehingga dapat dengan mudah digunakan dan hanya dibutuhkan kode program atau *source code* untuk menjalankan program yang akan dibuat. Proses pemrograman dilakukan dengan menggunakan *software* Arduino IDE, selanjutnya program yang dibuat nantinya akan diupload ke modul mikrokontroler arduino dengan koneksi kabel USB.

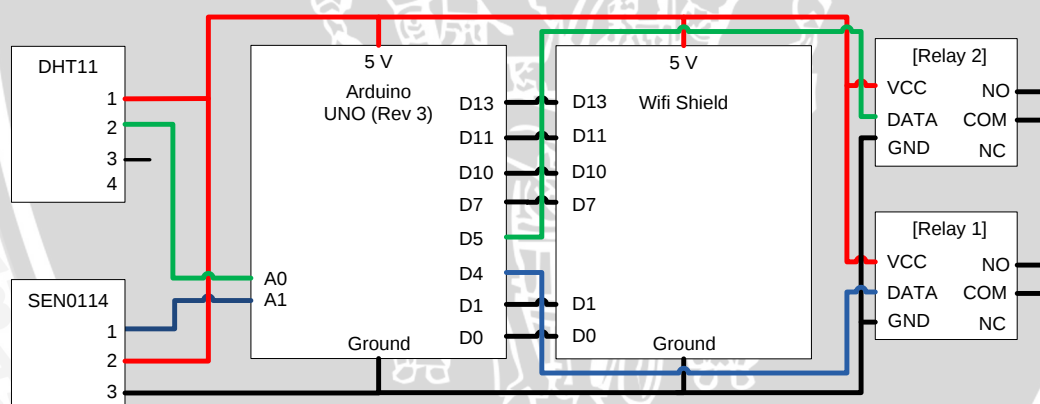
Arduino Uno memiliki *port* TX dan RX yang digunakan untuk komunikasi serial. WiFi *Shield* ini nantinya akan bertugas untuk mengirim data sensor. Berikut spesifikasi Arduino Uno yang digunakan :

- Microcontroller ATmega328
- Operasi dengan daya 5V
- Digital I / O Pins 14 (dimana 6 memberikan output PWM)

- Analog Input Pin 6
- DC Lancar per I / O Pin 40 mA
- Saat 3.3V Pin 50 mA DC
- *Flash Memory* 32 KB (ATmega328) yang 0,5 KB digunakan oleh *bootloader*
- SRAM 2 KB (ATmega328)
- EEPROM 1 KB (ATmega328)
- *Clock Speed* 16 MHz

Pada penelitian ini data inputan arduino berasal dari sensor DHT11 dan sensor SEN0114 yang terhubung pada pin A0 dan A1 modul arduino. Sebagai proses kendali suhu dan kelembaban tanah, aktuator dihubungkan dengan arduino melalui pin D4 dan D5. Proses pengiriman data sensor dilakukan oleh *WiFi shield* yang terhubung pada arduino melalui pin D0, D1, D7, D10, D11, D12, dan D13.

Penjelasan mengenai pin masukan dan keluaran mikrokontroler arduino pada penelitian ini ditunjukkan pada Gambar 4.12.



Gambar 4. 12 Skema pin masukan dan keluaran pada arduino
Sumber: [perancangan]

4.4 Perancangan Perangkat Lunak

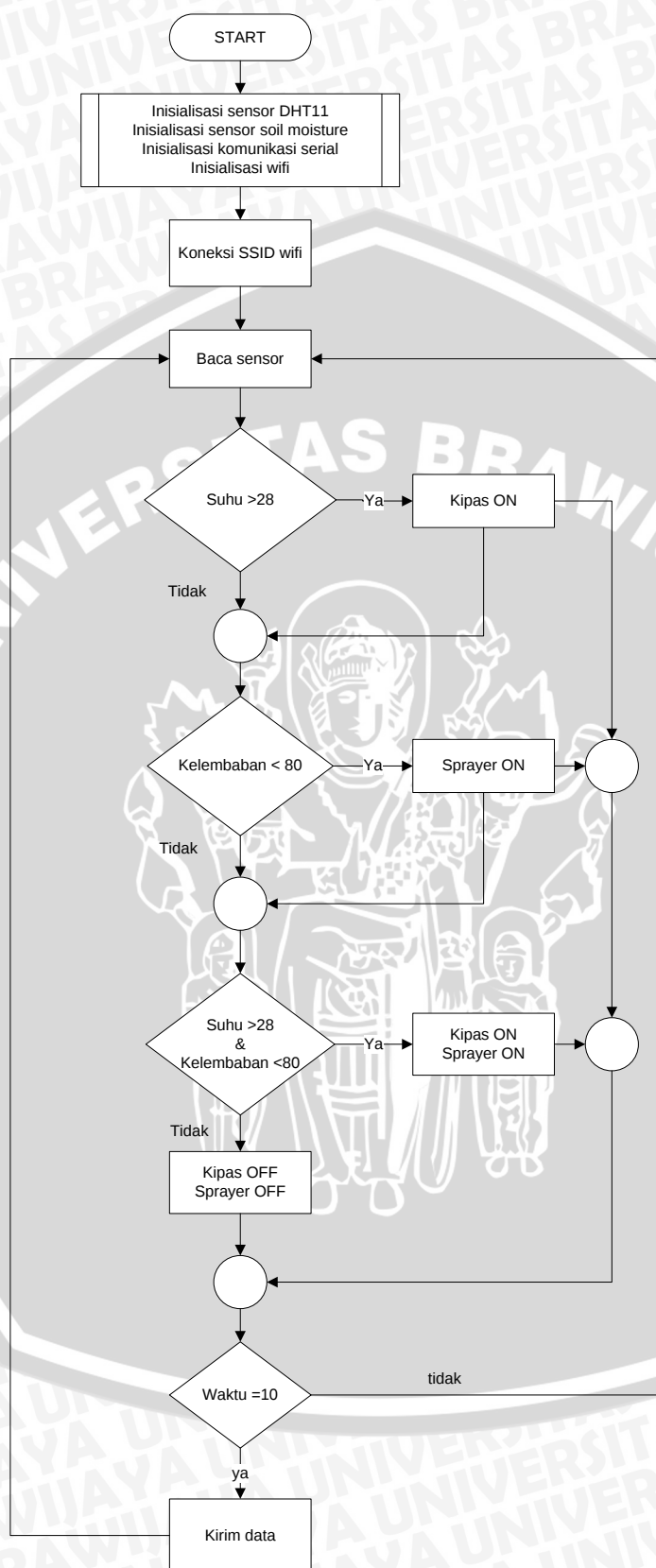
Perancangan sistem *software* untuk sistem ini terdiri dari pemrograman arduino yang digunakan untuk proses pengambilan dan pengolahan data sensor dan pemrograman pada *datasink*. Pemrograman ini menggunakan bahasa arduino berbasis bahasa C dengan *software* Arduino IDE untuk proses pemrograman dan memasukkan kode kedalam mikrokontrol arduino. Untuk proses penerimaan

data sensor yang dikirim dari *node* sensor ke *datalink* menggunakan pemrograman php dan database.

4.4.1 Perancangan Pemrograman Arduino

Perancangan pemrograman arduino dimulai dari proses pembacaan sensor suhu dan kelembaban tanah, selanjutnya data diperiksa apakah data sensor melebihi batas yang telah ditentukan sebelumnya untuk mengaktifkan aktuator. Data sensor akan dikirim melalui jaringan nirkabel menggunakan WiFi *shield* menuju *datasink*. Sebelum terhubung dengan *datasink*, *node* sensor harus terhubung dengan *access point* yang satu jaringan dengan *datasink*. Pengiriman data sensor menuju *server* menggunakan *method http post*, dengan cara memanggil file php yang berada pada *server* disertai data sensor. Diagram alir dari algoritma sistem dijelaskan pada Gambar 4.13.





Gambar 4. 13 Flowchart Node Sensor
Sumber: [perancangan]

Berdasarkan *flowchart* pada Gambar 4.13 dapat dibuat sebuah algoritma yang ditunjukkan pada Gambar 4.14.

Nama algoritma : Sistem pengendalian suhu dan kelembaban

Deklarasi

- Char SSID -> nama SSID *access point*.
- integer -> pin DHT11, pin sensorSoil, suhuRuang, kelembaban aktuator kipas, aktuator sprayer.
- Integer status -> status WiFi *shield*
- String Data
- Alamat IP server

Deskripsi

- Input : pin DHT11, pin sensorSoil.
- Proses :
 1. Inisialisasi sensor DHT11, inisialisai sensor SEN0114, inisialisasi komunikasi serial, dan inisialisasi WiFi
 2. Melakukan pengkoneksian sampai WiFi terkoneksi dengan SSID.
 3. Proses pembacaan sensor DHT11 dan SEN0114.
 4. Jika suhu yang terbaca > 28, maka kipas menyala.
 5. Jika kelembaban yang terbaca < 75, maka sprayer menyala.
 6. Jika suhu yang terbaca > 28 dan kelembaban < 75, maka kipas menyala dan sprayer menyala.
 7. Jika kedua parameter suhu dan kelembaban tidak memenuhi, maka kipas dan sprayer mati.
 8. Jika waktu interval memenuhi yaitu 10 detik, maka data dikirim ke server.

Output : *node* sensor dapat membaca nilai sensor, mengaktifkan dan menonaktifkan aktuator sesuai dengan pengkondisian, *node* sensor dapat mengirim data sensor ke server sesuai dengan waktu interval pengiriman dan data tersimpan pada database.

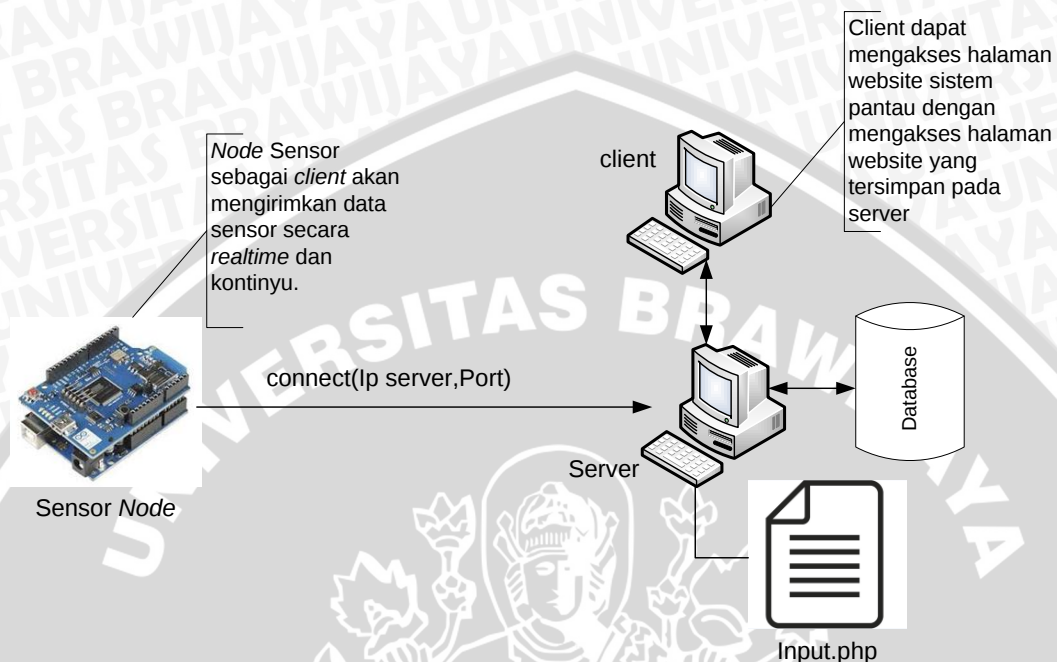
Gambar 4. 14 Pseudocode Algoritma Node Sensor

Sumber: [perancangan]

4.4.2 Perancangan Datasink

Perancangan *datasink* merupakan perancangan sistem pada sisi komputer yang bertindak sebagai *server* yang berfungsi sebagai penerima data yang dikirim oleh *node* sensor secara *realtime* dan berkelanjutan. Pada *datasink* inilah terdapat *service webserver* yang digunakan untuk melayani pelayanan sistem informasi berupa *website* sistem pantau yang menyajikan informasi mengenai kondisi lahan pantau. *Node* sensor akan memanggil file *input.php* yang berada pada sisi *server* disertai dengan data-data sensor dengan *method http post*. Data yang diterima akan disimpan ke dalam database dengan menggunakan *php* sebagai penghubung

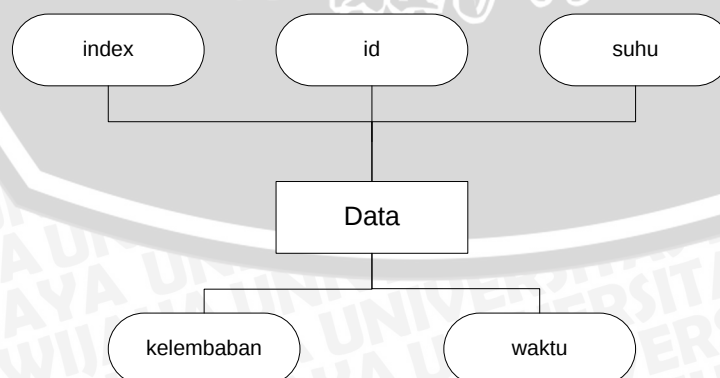
ke database. Selanjutnya komputer *client* dapat mengakses halaman website sistem pantau yang tersimpan pada komputer server. Skema proses yang terjadi pada *datasink* ditunjukkan pada Gambar 4.15.



Gambar 4. 15 Skema proses pada *datasink*
Sumber: [perancangan]

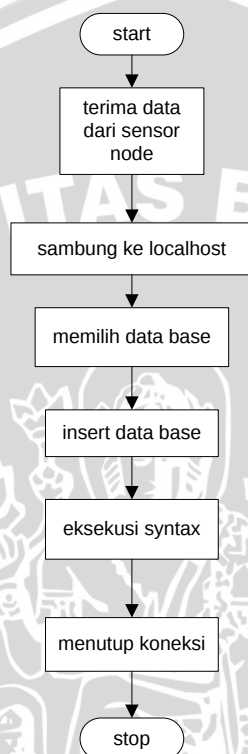
4.4.2.1 Perancangan Database

Pada perancangan database ini yang perlu dirancang pertama kali adalah perancangan *Entity Relationship Diagram* (ERD). Entitas pada database ini yaitu entitas data yang berisi tentang data-data suhu dan kelembaban yang disimpan di server. ERD dari basisdata arduino ditunjukkan pada Gambar 4.16.



Gambar 4. 16 ERD Data Sensor
Sumber: [perancangan]

Algoritma koneksi antara *datasink* dan *node* sensor berfungsi untuk mengupdate dan menerima data dari *node* sensor ke dalam database. *Flowchart* algoritma koneksi antara *datasink* dan *node* sensor ditunjukkan pada Gambar 4.17 dan penjelasan mengenai *pseudocode* koneksi database ditunjukkan pada Gambar 4.18.



Gambar 4. 17 Flowchart algoritma koneksi antara *datasink* dan *node* sensor
Sumber: [perancangan]

Nama algoritma : Koneksi database

Deklarasi

- Method get
- Koneksi basis data

Deskripsi

- Input : data dari *node* sensor
- Proses :
 1. Menerima data dari method get *node* sensor
 2. Menyambungkan ke database dengan server localhost
 3. Memilih database
 4. Membuat syntax insert data ke database
 5. Mengeksekusi syntax
 6. Menutup koneksi ke database
- Output : Data masuk ke dalam basis data

Gambar 4. 18 Pseudocode koneksi database
Sumber: [perancangan]

Dalam perancangan database ini dibutuhkan satu *tabel* untuk menyimpan nilai sensor yang telah dikirim. Pada tabel tersebut terdiri dari beberapa kolom dengan tipe data tertentu. Untuk penjelasan mengenai struktur tabel pada database sistem ditunjukkan pada Tabel 4.2.

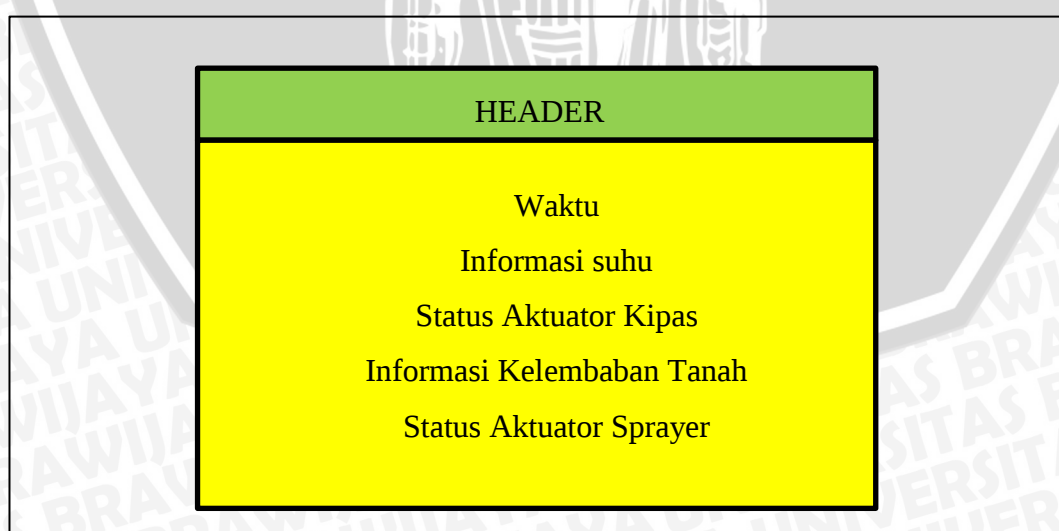
Tabel 4. 2 Struktur tabel database

NO	Nama Kolom	Tipe Data	Keterangan
1	index	Integer (11)	Primary
2	id	Integer (11)	
3	suhu	Integer (11)	
4	kelembaban	Integer (11)	
5	waktu	timestamp	on update CURRENT_TIMESTAMP

Sumber: [perancangan]

4.4.2.2 Perancangan Antarmuka

Perancangan tampilan antar muka website bertujuan untuk menampilkan data terbaru sensor suhu dan kelembaban yang tersimpan pada database. Untuk menampilkan data yang selalu terbaharui yang terdapat pada database perlu adanya fungsi *autorefresh*. Pemrograman PHP didesain sebagai bahasa pemrograman berbasis *web* yang bisa memproses dengan cepat serta mudah diaplikasikan ke berbagai macam *platform* OS dan hampir semua *browser* bisa mengakses *web* dengan PHP. Informasi yang ditampilkan berupa informasi waktu pembacaan sensor, nilai dari sensor suhu dan kelembaban serta status dari kedua aktuator. Layout tampilan antar muka *website* ditunjukkan pada Gambar 4.19.



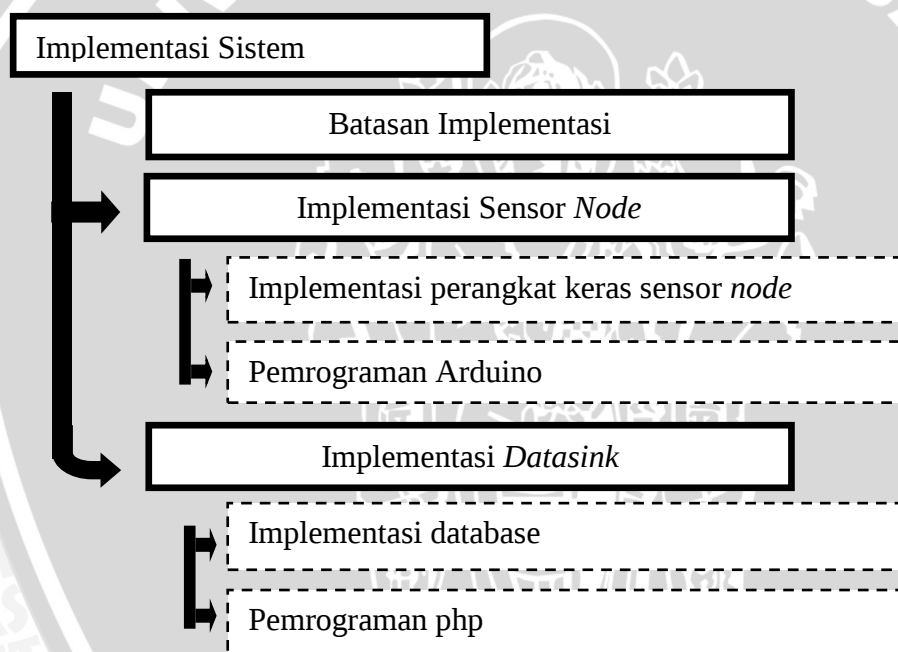
Gambar 4. 19 Desain website

Sumber: [perancangan]

BAB V

IMPLEMENTASI SISTEM

Pada bab implementasi akan dibahas mengenai langkah-langkah yang dilakukan dalam pembuatan sistem pengendalian suhu dan kelembaban berbasis *wireless embedded system*. Sesuai dengan rancangan pada Bab IV, sistem ini dibangun menggunakan perangkat arduino uno, WiFi *shield*, sensor DHT11, sensor SEN0114 dan relay. Pada tahap implementasi ini, langkah-langkah yang akan dilakukan penulis adalah instalasi perangkat keras, pembuatan program untuk arduino, pembuatan database dan pembuatan program php.



Gambar 5. 1 Diagram Pohon Implementasi
Sumber: [implementasi]

5.1 Batasan Implementasi

Beberapa batasan dalam mengimplementasikan Sistem pengendalian suhu dan kelembaban berbasis *wireless embedded system* sebagai berikut :

1. Konfigurasi *ip address* pada perangkat *node* sensor menggunakan metode DHCP.
2. Database *management system* yang digunakan adalah MySQL.
3. Sumber tegangan untuk perangkat *node* sensor berasal dari kabel USB yang terhubung dengan komputer
4. Untuk terhubung dengan jaringan pengaturan pada *access point* dibuat terbuka dan tidak menggunakan *password*

5.2 Implementasi Node Sensor

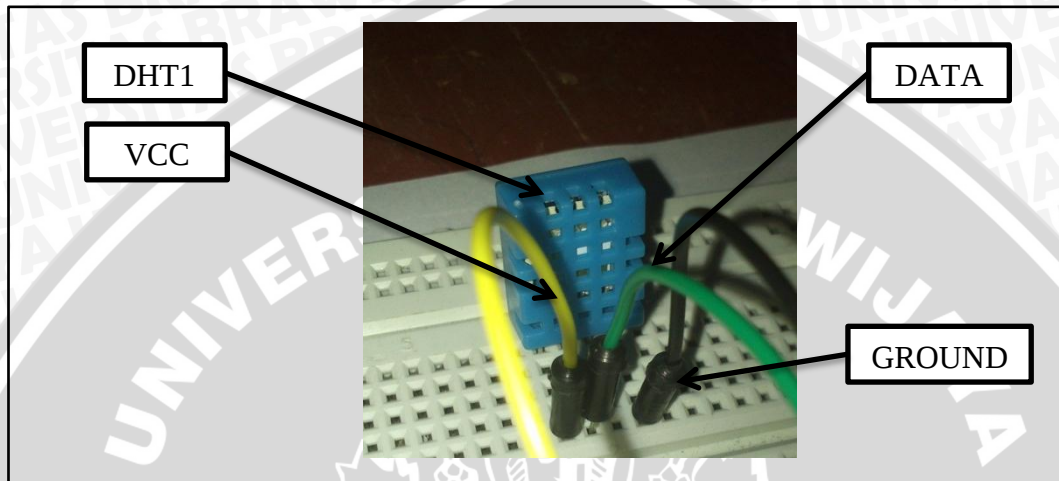
Pada implementasi *node* sensor dijelaskan mengenai langkah-langkah yang dilakukan dalam instalasi dan pemrograman perangkat *node* sensor. modul yang digunakan meliputi arduino uno, sensor suhu DHT11, sensor SEN0114, WiFi *shield*, dan relay. Untuk proses kompilasi dan memasukkan kode program ke dalam arduino menggunakan perangkat lunak arduino ide dengan menggunakan bahasa C.

5.2.1 Implementasi Perangkat Keras Node Sensor

Langkah awal yang dilakukan dalam membangun sistem pengendalian suhu dan kelembaban berbasis *wireless emebded system* adalah mempersiapkan perlengkapan alat yang digunakan yang meliputi:

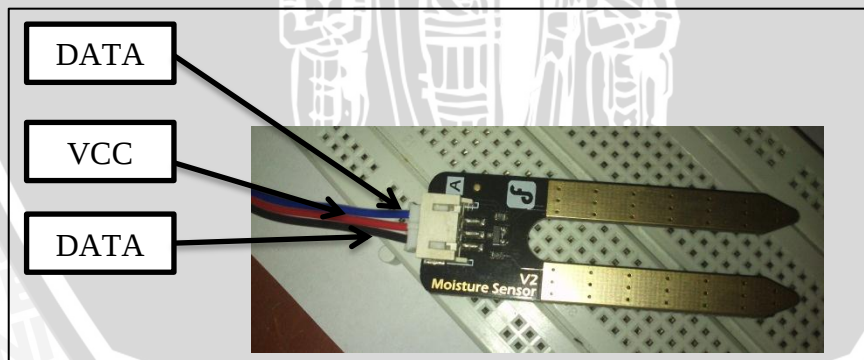
1. Arduino Uno R3
2. WiFi *shield*
3. Sensor suhu ruangan (DHT11) dan Sensor kelembaban tanah (SEN0114).
4. *Bread board*
5. Kabel
6. Kipas dan sprayer (pompa air)
7. 2 buah relay

Tahap selanjutnya adalah pemasangan WiFi *shield* pada arduino dengan cara menumpuknya tepat diatas arduino dan menghubungkan sensor DHT11 serta SEN0114. Pada sensor DHT11 terdapat 3 pin yang digunakan dengan pin data dihubungkan pada pin A0 WiFi *shield*, sedangkan pin *ground* dihubungkan pada *ground* dan pin vcc dihubungkan pada 5v seperti Gambar 5.2.



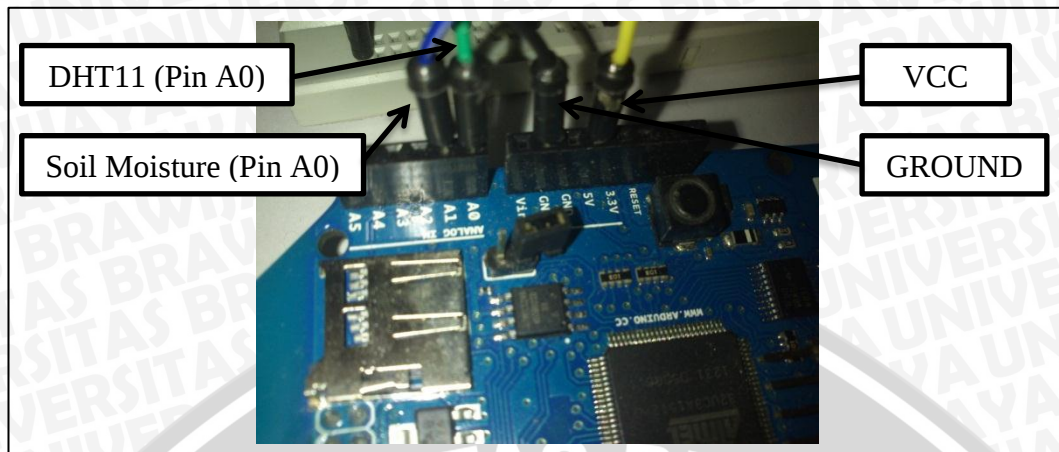
Gambar 5. 2 Pemasangan kabel pada DHT11
Sumber: [implementasi]

Pada sensor SEN0114 memiliki 3 pin dengan pin data dihubungkan pada pin A1 WiFi *shield*, sedangkan pin *ground* dihubungkan pada *ground* dan pin vcc dihubungkan pada 5v seperti Gambar 5.3.



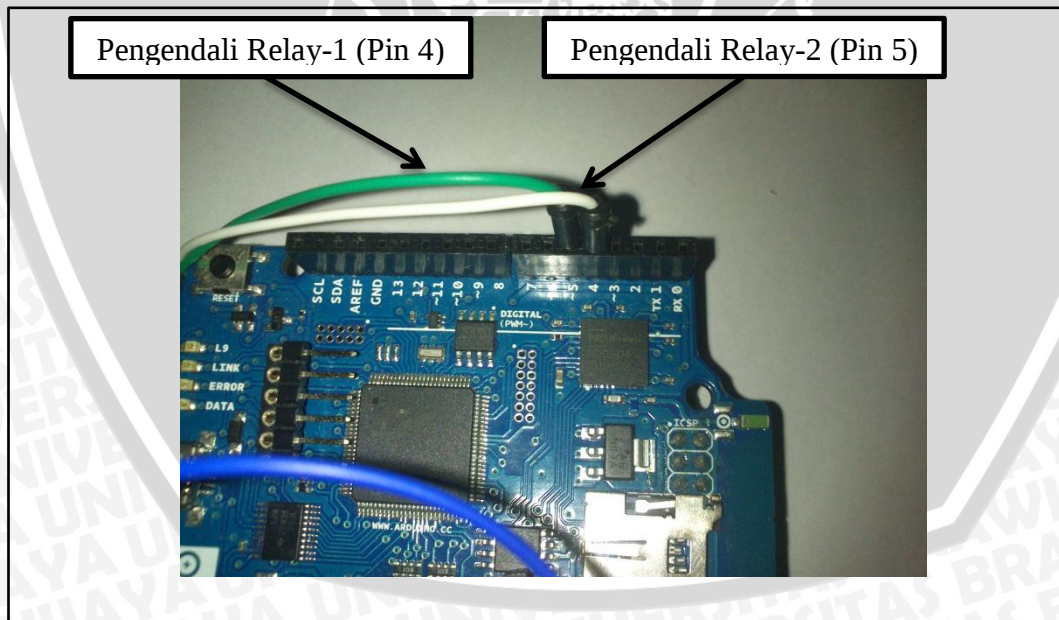
Gambar 5. 3 Konfigurasi sensor SEN0114
Sumber: [implementasi]

Penjelasan mengenai pemasangan kabel pada arduino yang menghubungkan sensor SEN0114 dan sensor suhu DHT11 ditunjukkan pada Gambar 5.4.

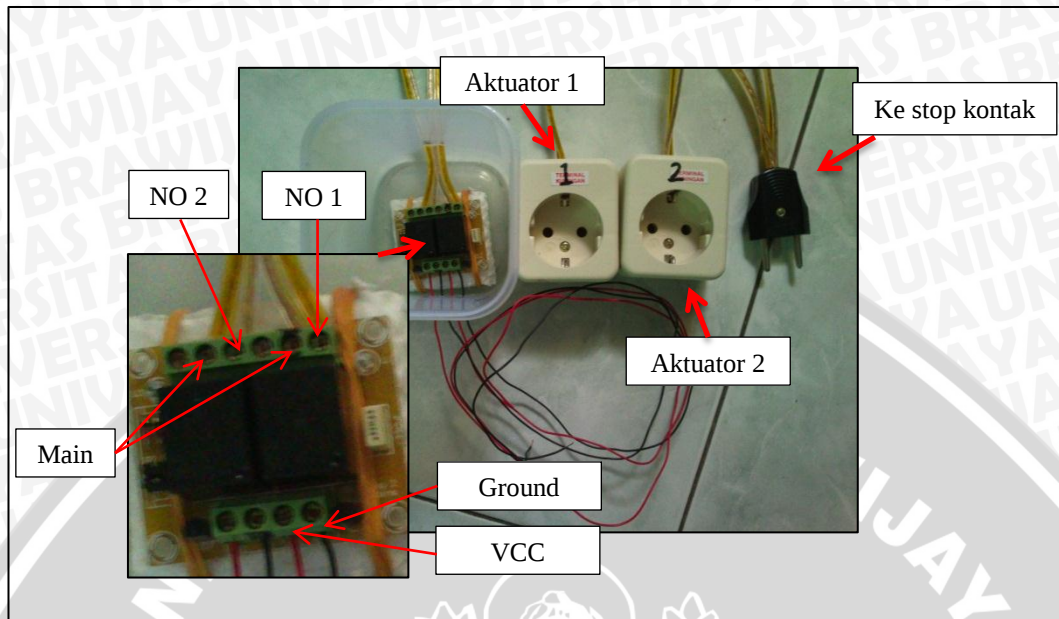


Gambar 5. 4 Konfigurasi pin sensor pada arduino
Sumber: [implementasi]

Tahap selanjutnya adalah menghubungkan relay dengan komponen yang telah dibangun sebelumnya (Arduino uno, WiFi shield, sensor DHT11, dan sensor SEN0114). Pada modul relay *ground* dihubungkan ke *ground*, *vcc* dihubungkan ke 5v, pin pengendali relay-1 dihubungkan ke pin D4 dan pin pengendali relay-2 dihubungkan ke pin D5 arduino seperti Gambar 5.5. Penjelasan mengenai pemasangan relay ditunjukkan pada Gambar 5.6.

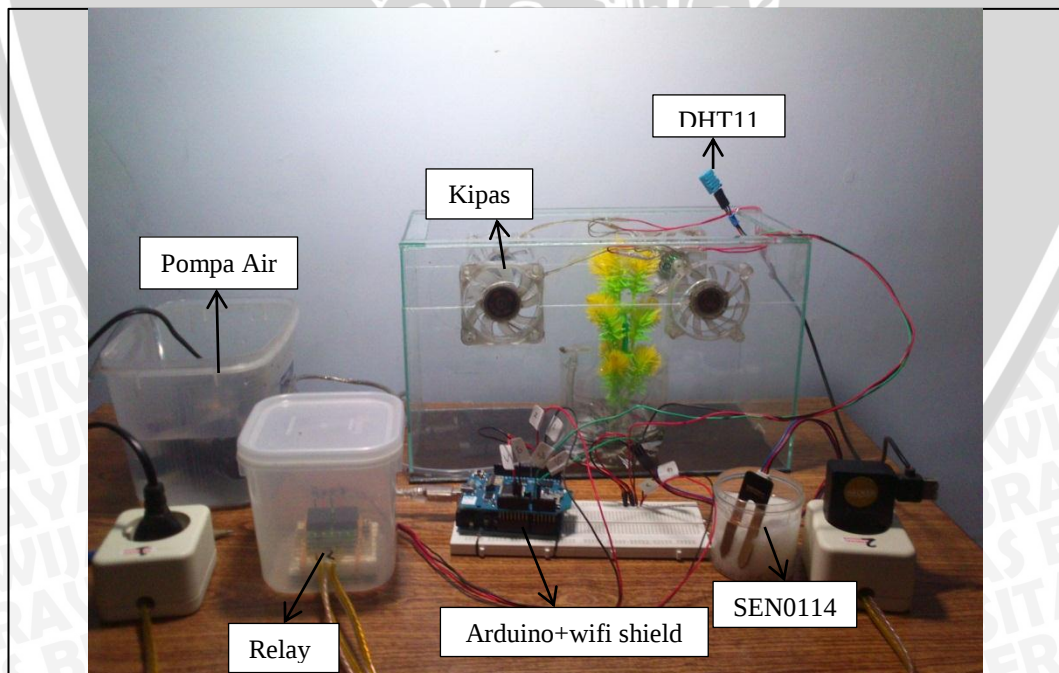


Gambar 5. 5 Konfigurasi pin aktuator pada arduino
Sumber: [implementasi]



Gambar 5. 6 Pemasangan relay pada arduino
Sumber: [implementasi]

Setelah semua perangkat pada *node* sensor telah terpasang semua, *node* sensor siap digunakan. Hasil imlementasi perangkat *node* sensor ditunjukkan pada Gambar 5.7.



Gambar 5. 7 Perangkat node sensor
Sumber: [implementasi]

5.2.2 Pemrograman Arduino

Setelah melakukan proses implementasi perangkat keras *node* sensor selesai dilakukan, proses selanjutnya adalah pembuatan berkas program arduino. Berdasarkan sub bab 4.4.1 perancangan pemrograman arduino diimplementasikan menggunakan bahasa pemrograman arduino berbasis bahasa C. aplikasi yang digunakan untuk melakukan pemrograman arduino adalah arduino IDE. Pada penelitian ini *node* sensor terhubung dengan *access point* yang bersifat terbuka dan tidak ada *password*.

Langkah awal yang dilakukan adalah mikrokontroller Arduino Uno disiapkan. Setelah itu, koneksi USB dihubungkan ke dalam *port* USB laptop, dengan tujuan untuk komunikasi serial dengan laptop dan memberikan daya pada Arduino Uno agar tetap bekerja. Berikut ini penulis menjabarkan perintah-perintah (fungsi) pada berkas program arduino tersebut, untuk *list* kode program secara keseluruhan dilampirkan pada Lampiran 1.

1. Fungsi setup

Fungsi setup merupakan bagian persiapan atau inisialisasi program, berisi kode program yang digunakan untuk membuat konfigurasi pada *node* sensor. Fungsi ini hanya dijalankan sekali sesaat setelah Arduino dihidupkan atau di-*reset*. Fungsi setup program ditunjukkan pada Gambar 5.8.

```

1. void setup()
2. {
3.   idSensor=1;
4.   pinMode(sprayer,OUTPUT);
5.   pinMode(kipas,OUTPUT);
6.   Serial.begin(9600);
7.   setTime(8,29,0,1,1,11);
8.   Alarm.timerRepeat(10, Repeats);
9.   while ( status != WL_CONNECTED) {
10.     Serial.print("Mencoba melakukan koneksi ke SSID: ");
11.     Serial.println(ssid);
12.     status = WiFi.begin(ssid);
13.     Serial.print("Lama koneksi: ");
14.     Serial.println(millis()/1000);
15.   }
16.   printWiFiStatus();
17. }
18.

```

Gambar 5. 8 Potongan kode fungsi void setup
Sumber: [implementasi]

Baris 4 dan 5 merupakan konfigurasi pin sprayer dan kipas sebagai pin *output*. Baris 6 merupakan konfigurasi komunikasi serial dengan *boudrate* yang digunakan untuk komunikasi serial. Baris 7 merupakan konfigurasi waktu *real time clock* (RTC), namun pada penelitian ini tidak menggunakan modul RTC eksternal sehingga ketika arduino mati atau *reset* fungsi waktu tidak terupdate. Baris 8 merupakan konfigurasi fungsi pengulangan dengan waktu yang telah diset 10 detik, artinya tiap 10 detik program akan memanggil fungsi *Repeats*. Baris 9-15 merupakan proses koneksi WiFi *shield* dengan *access point*, baris 14 merupakan kode program untuk menghitung lama konfigurasi koneksi dengan *access point*. Baris 16 merupakan kode program untuk memanggil fungsi *printWiFiStatus* apabila koneksi berhasil.

2. Fungsi loop

Fungsi ini berisi kode program yang akan dijalankan terus menerus selama arduino menyala atau tempat untuk program utama. Fungsi loop ditunjukkan pada Gambar 5.9.

```
1. void loop(){
2.   aktuator();
3.   Alarm.delay(1000);
4. }
```

Gambar 5. 9 Potongan kode fungsi void loop
Sumber: [implementasi]

3. Fungsi bacaSensor

Fungsi ini digunakan untuk membaca nilai sensor. Konfigurasi pin sensor yang digunakan ditempatkan pada fungsi ini. Fungsi *bacaSensor* ditunjukkan pada Gambar 5.10.

```
1. void bacaSensor()
2. {
3.   DHT.read11(dht_dpin);
4.   suhuRuang = DHT.temperature;
5.   lembabTanah = analogRead(soil_pin)*0.127;
6. }
```

Gambar 5. 10 Potongan kode fungsi bacaSensor
Sumber: [implementasi]

Baris 3 merupakan konfigurasi pin yang digunakan untuk sensor suhu. Baris 4 merupakan kode pengambilan nilai sensor yang kemudian disimpan

pada variabel suhuRuang. Baris 5 merupakan pengambilan nilai sensor dan pengkalibrasian agar nilai yang dihasilkan siap digunakan.

4. Fungsi SerialEvent

Fungsi ini digunakan untuk menampilkan informasi nilai-nilai sensor melalui komunikasi serial monitor. Fitur serial monitor dapat dilihat pada arduino IDE. Fungsi SerialEvent ditunjukkan pada Gambar 5.11.

```
1. void SerialEvent(){
2.   bacaSensor();
3.   Serial.print("Current humidity = ");
4.   Serial.print(lembabTanah);
5.   Serial.print(" % ");
6.   Serial.print("temperature = ");
7.   Serial.print(suhuRuang);
8.   Serial.print(" C ");}
```

Gambar 5. 11 Potongan kode fungsi SerialEvent

Sumber: [implementasi]

Baris 2 merupakan pemanggilan fungsi bacaSensor yang digunakan untuk memperoleh nilai terbaru dari sensor. Baris 3-8 merupakan kode program yang digunakan untuk menampilkan informasi nilai-nilai sensor melalui program serial monitor pada arduino IDE.

5. Fungsi aktuator

Fungsi ini digunakan untuk mengaktifkan dan menonaktifkan aktuator dan menampilkan informasi mengenai nilai sensor dan status aktuator melalui serial monitor. Pada fungsi ini arduino akan memberikan sinyal *high* atau *low* kepada relay ketika suatu kondisi terpenuhi. Fungsi aktuator ditunjukkan pada Gambar 5.12.

```
1. void aktuator()
2. {
3.   bacaSensor();
4.   if(suhuRuang >= 28 && lembabTanah >= 75)
5.   {
6.     SerialEvent();
7.     Serial.println(" >>kipas on");
8.     digitalWrite(kipas, HIGH);
9.     digitalWrite(sprayer, LOW);
10.  }
11.  else if (suhuRuang >= 28 && lembabTanah < 75)
12.  {
13.    SerialEvent();
14.    Serial.println(" >>spray on kipas on");
15.    digitalWrite(kipas, HIGH);
16.    digitalWrite(sprayer, HIGH);
17.  }
18.  else if(suhuRuang < 28 && lembabTanah < 75)
```

```

19. {
20.   SerialEvent();
21.   Serial.println(" >>spray on ");
22.   digitalWrite(kipas, LOW);
23.   digitalWrite(sprayer, HIGH);
24. }
25. else
26. {
27.   SerialEvent();
28.   Serial.println();
29.   digitalWrite(kipas, LOW);
30.   digitalWrite(sprayer, LOW);
31. }
32. }

```

Gambar 5. 12 Potongan kode fungsi aktuator
Sumber: [implementasi]

Baris 3 digunakan untuk memanggil fungsi bacaSensor untuk mengetahui nilai terbaru dari sensor. Baris 5-10 merupakan kode yang dijalankan ketika kondisi suhu ≥ 28 dan kelembaban ≥ 75 terpenuhi, arduino akan memberikan sinyal *high* pada pin kipas dan sinyal *low* pada pin sprayer. Baris 12-17 merupakan kode yang dijalankan ketika kondisi suhu ≥ 28 dan kelembaban < 75 terpenuhi, arduino akan memberikan sinyal *high* pada pin kipas dan sprayer. Baris 19-24 merupakan kode yang dijalankan ketika kondisi suhu < 28 dan kelembaban < 75 terpenuhi, arduino akan memberikan sinyal *low* pada pin kipas dan sinyal *high* pada pin sprayer. Baris 26-31 merupakan kode yang dijalankan apabila semua kondisi tidak terpenuhi atau kondisi dikatakan normal, sehingga kedua aktuator pada posisi mati.

6. Fungsi Repeats

Fungsi yang digunakan untuk menjalankan perintah tiap periode waktu (10 detik). Pada fungsi ini perintah yang dijalankan tiap 10 detik adalah fungsi kirim_data. Tiap 10 detik nilai-nilai sensor akan dikirimkan ke server, sehingga nantinya data yang tersimpan pada database memiliki interval 10 detik tiap data yang diterima. Fungsi Repeats ditunjukkan pada Gambar 5.13.

```

1. void Repeats() {
2.   kirim_data();
3. }

```

Gambar 5. 13 Potongan kode fungsi Repeats
Sumber: [implementasi]

7. Fungsi printWiFiStatus

Fungsi ini digunakan untuk memberikan informasi mengenai status WiFi. Fungsi ini dipanggil sekali selama proses koneksi dan pemanggilan dilakukan pada fungsi setup. Fungsi printWiFiStatus memberikan informasi mengenai status koneksi WiFi *shield* dengan *access point* dan alamat ip *node* sensor. Fungsi printWiFiStatus ditunjukkan pada Gambar 5.14.

```
1. void printWiFiStatus() {
2.     Serial.println("Status WiFi: ");
3.     Serial.println("Koneksi berhasil ");
4.     Serial.print("SSID: ");
5.     Serial.println(WiFi.SSID());
6.     IPAddress ip = WiFi.localIP();
7.     Serial.print("IP Address: ");
8.     Serial.println(ip);
9. }
```

Gambar 5. 14 Potongan kode fungsi paket_data
Sumber: [implementasi]

8. Fungsi paket_data

fungsi ini digunakan untuk membuat paket data yang akan dikirimkan dengan menggunakan *method http post*. Paket data yang dikirimkan terdiri dari id sensor, nilai sensor suhu dan kelembaban. Fungsi paket_data ditunjukkan pada Gambar 5.15.

```
1. void paket_data(){
2.     bacaSensor();
3.     data+="id=";
4.     data+=idSensor;
5.     data+="&kelembaban=";
6.     data+=lembabTanah;
7.     data+="&suhu=";
8.     data+=suhuRuang;
9. }
```

Gambar 5. 15 Potongan kode fungsi paket_data
Sumber: [implementasi]

Baris 2 digunakan untuk memanggil fungsi bacaSensor untuk mengetahui nilai terbaru dari sensor. Baris 3 dan 4 merupakan variabel yang berisi id sensor, karena pada penelitian ini *node* sensor hanya satu maka semua id sensor berisi nilai 1. Baris 5 dan 6 merupakan variabel yang berisi data kelembaban tanah. Baris 7 dan 8 merupakan variabel yang berisi data suhu ruangan. Ketiga data tersebut disimpan pada satu variabel sehingga data yang disimpan pada variabel

tersebut tergabung satu sama lain. Sebagai pemisah antara informasi id sensor, data suhu dan kelembaban digunakan pemisah berupa tanda “&”.

9. Fungsi kirim_data

Fungsi ini digunakan untuk mengirim paket data yang berisi informasi mengenai id sensor, data suhu dan kelembaban yang dikirim setiap 10 detik. Fungsi ini dipanggil oleh fungsi Repeats yang selalu berjalan tiap 10 detik. Fungsi kirim_data ditunjukkan pada Gambar 5.16.

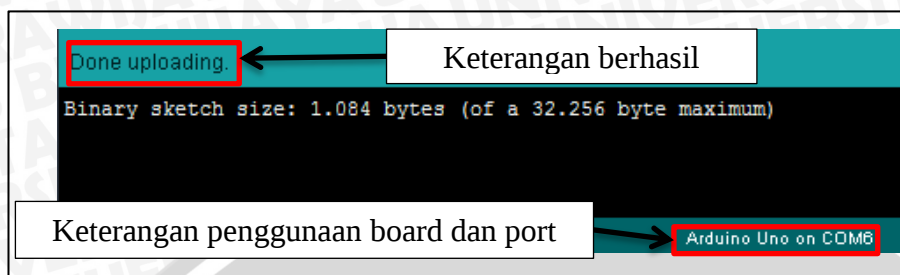
```
1. void kirim_data(){
2.     if (client.connect(server, 80)) {
3.         paket_data();
4.         SerialEvent();
5.         client.println("POST                /skripsi/insert_mysql.php
6. HTTP/1.1");
7.         client.println("Host: www.yourdomain.com");
8.         client.println("User-Agent: Arduino/1.0");
9.         client.println("Connection: close");
10.        client.println("Content-Type:        application/x-www-form-
11. urlencoded;");
12.        client.print("Content-Length: ");
13.        client.println(data.length());
14.        client.println();
15.        client.println(data);
16.        Serial.println(" >>Terkirim ");
17.        data="";
18.    }
19.    else {
20.        Serial.println("Koneksi ke server gagal!");
21.        client.stop();
22.    }
23. }
```

Gambar 5. 16 Potongan kode fungsi kirim_data
Sumber: [implementasi]

Baris 2 merupakan proses koneksi dengan *server* dengan menyertai alamat *ip server* dan *port* yang digunakan dalam pengiriman. Baris 3 merupakan pemanggilan fungsi *paket_data* yang berisi id sensor, data suhu dan kelembaban dengan membentuk format yang sesuai agar dapat dikirim dengan *method http post*. Baris 5-15 merupakan proses pengiriman data-data sensor menuju *server*.

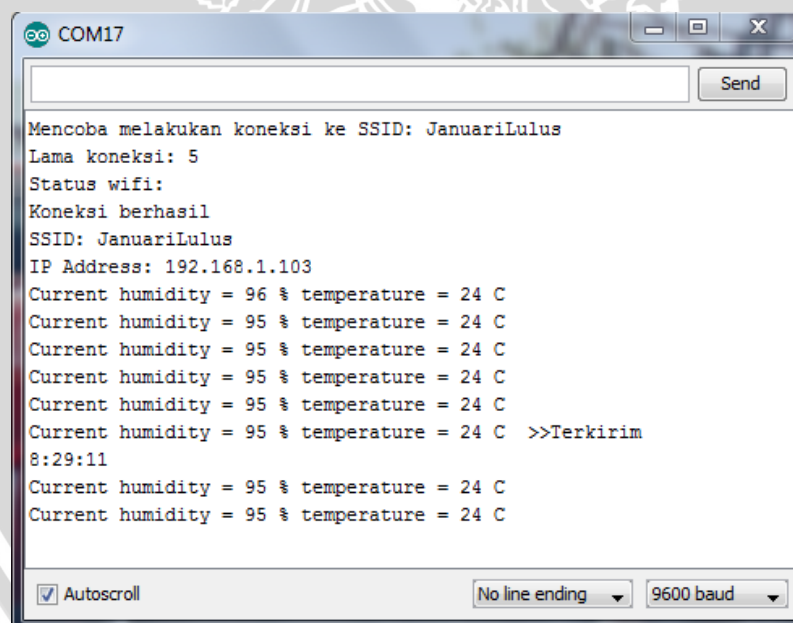
Setelah kode program arduino selesai ditulis, langkah selanjutnya adalah klik menu *verify* untuk memastikan kode yang diprogram benar, kemudian klik menu *upload* untuk memasukkan kode ke arduino. Sebelum memasukkan kode ke dalam arduino pastikan pengaturan tipe *board* dan *port* yang digunakan benar, dalam penelitian ini digunakan *board arduino uno* dan *port COM 17*. Pengaturan

tipe *board* ada di menu Tools -> board -> Arduino Uno sedangkan pengaturan *port* ada di menu Tools -> Serial port -> COM 17. Apabila program berhasil di upload maka akan keluar pemberitahuan seperti Gambar 5.17.



Gambar 5. 17 Pemberitahuan pada arduino IDE
Sumber: [implementasi]

Setelah program berhasil diupload, informasi data-data yang ada pada *node* sensor dapat dilihat dengan menggunakan serial monitor. Tampilan informasi-informasi yang ada pada *node* sensor dapat ditunjukkan pada Gambar 5.18.



Gambar 5. 18 Tampilan informasi node sensor melalui serial monitor
Sumber: [implementasi]

5.3 Implementasi *Datasink*

Implementasi *datasink* meliputi pemrograman php dan database. Database yang digunakan adalah Mysql. Pada implementasi data *sink* data yang diterima oleh server akan dimasukkan ke dalam database.

5.3.1 Implementasi Database

Berdasarkan struktur tabel database pada subbab 4.3.2 perancangan *datasink*, database diimplementasikan menggunakan bahasa pemrograman MySQL. Selain itu juga menggunakan web server XAMPP. Berikut ini implementasi database menggunakan bahasa pemrograman MySQL ditunjukkan pada Tabel 5.1.

Tabel 5. 1 Tabel database arduino

NAMA	data
KETERANGAN	tabel ini digunakan untuk menyimpan nilai sensor suhu dan kelembaban tanah
DDL	<pre>CREATE TABLE IF NOT EXISTS `data` (`index` int(11) NOT NULL AUTO_INCREMENT, `id` int(11) NOT NULL, `suhu` int(11) NOT NULL, `kelembaban` int(11) NOT NULL, `waktu` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP, PRIMARY KEY (`index`)) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=610 ;</pre>

Sumber: [implementasi]

Berdasarkan DDL database yang ditunjukkan pada Tabel 5.1 dihasilkan struktur tabel pada database seperti Gambar 5.19.

Server: localhost ▶ Database: skripsi ▶ Table: data

Browse

Structure

SQL

Search

Insert

Export

Import

Operations

Empty

Drop

	Field	Type	Collation	Attributes	Null	Default	Extra
☒	index	int(11)			No	None	auto_increment
☒	id	int(11)			No	None	
☒	suhu	int(11)			No	None	
☒	kelembaban	int(11)			No	None	
☒	waktu	timestamp		on update CURRENT_TIMESTAMP	No	CURRENT_TIMESTAMP	on update CURRENT_TIMESTAMP

Gambar 5. 19 Struktur tabel pada database
Sumber: [implementasi]

5.3.2 Pemrograman PHP

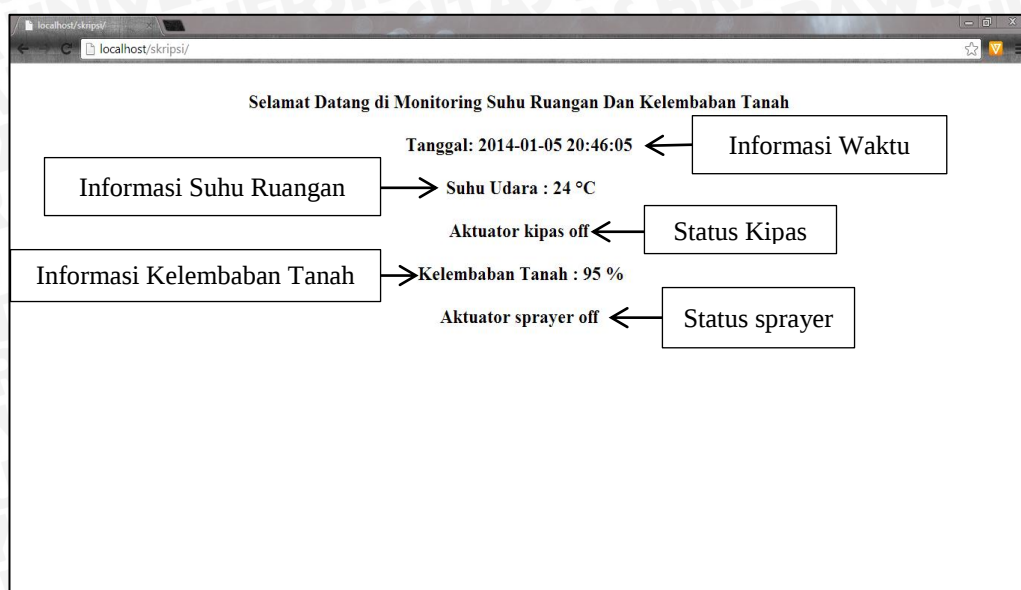
Pada pemrograman php yang dilakukan adalah membuat file berkas php yang digunakan untuk koneksi database dan memasukkan nilai sensor kedalam database. berdasarkan pada sub bab 4.4.2 Perancangan *Datasink*, inputSensor.php berfungsi sebagai penghubung data yang diperoleh dari *node* sensor ke dalam basisdata

Berikut ini *source code* berkas insert_mysql.php

```
1. <?php
2. $username = "root";
3. $password = "";
4. $database = "skripsi";
5. $tablename = "data";
6. $localhost = "localhost";
7. if($_POST){
8. if (mysql_connect($localhost, $username, $password)){
9.     @mysql_select_db($database) or die ("Unable to select
        database");
10.     $query = "INSERT INTO $tablename (id, suhu, kelembaban,
        waktu) VALUES
        ('$_POST[id]', '$_POST[suhu]', '$_POST[kelembaban]', now())";
11.     $result = mysql_query($query);
12. } else {echo('Unable to connect to database.')}
13. }?>
```

Gambar 5. 20 Source Code inputSensor.php
Sumber: [implementasi]

Baris 2-6 merupakan konfigurasi mengenai database yang digunakan, meliputi *username* dan *password* database, nama database, nama tabel, dan nama *host*. Baris 7 merupakan proses pengecekan data yang dilewatkan melalui *method http post*. Baris 8 merupakan proses koneksi database dan baris 9 merupakan *query* untuk memasukan data ke dalam database. Setelah data berhasil dikirim dari *node* sensor ke *server*, informasi mengenai keadaan lahan pantau dapat dilihat melalui website seperti Gambar 5.21.



Gambar 5. 21 Tampilan web sistem pantau
Sumber: [implementasi]



BAB VI

PENGUJIAN DAN ANALISIS

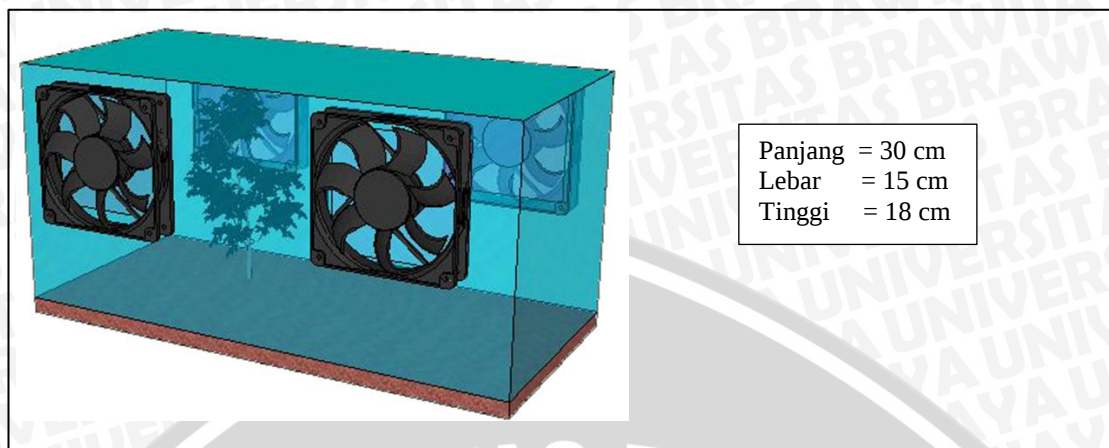
Pada bab ini akan dijelaskan bagaimana cara melakukan pengujian dan analisa terhadap sistem yang telah dibuat. Pengujian dilakukan dalam 2 macam yaitu pengujian unit dan pengujian sistem secara keseluruhan. Pengujian unit terdiri dari pengujian sensor suhu, pengujian sensor kelembabaan tanah, pengujian WiFi *shield*, dan pengujian aktuator. Pengujian dan analisis sistem perlu dilakukan untuk mengetahui apakah sistem yang telah dirancang dan diimplementasikan telah bekerja sesuai dengan tujuan.

6.1 Lingkungan pengujian

Lingkungan pengujian yang dilakukan pada sistem ini:

1. Dilakukan di lab dengan koneksi jaringan *wireless*.
2. *Access point* yang digunakan TP-link.
3. Node sensor berjumlah 1.
4. Server menggunakan laptop toshiba e205.
5. SSID access point bersifat terbuka dan tidak ada *password*.
6. Sistem disimulasikan pada model *green house*.
7. Untuk memanipulasi kondisi suhu dilakukan dengan memanaskan udara disekitar sensor DHT11 menggunakan *hairdryer* atau *blower*.
8. Diasumsikan suhu didalam akuarium/model pengujian tidak sama dengan suhu diruang akuarium.

Desain model simulasi dilakukan pada sebuah box dengan ventilasi kipas elektrik DC pada sisi depan dan belakang :



Gambar 6. 1 Desain model simulasi
Sumber: [Perancangan]

6.2 Pengujian Unit

Pengujian unit adalah pengujian sistem berdasarkan unit tertentu. Pengujian ini bertujuan untuk menganalisa apakah bagian unit dari sistem ini bekerja dengan baik atau tidak. Pengujian unit ini terdiri dari pengujian sensor suhu, pengujian sensor kelembabaan tanah, pengujian aktuator, dan pengujian WiFi shield.

6.2.1 Pengujian Sensor Suhu

Pengujian sensor suhu dilakukan dengan membandingkan hasil pengukuran sensor DHT11 dengan termometer ruangan digital. Pengujian dilakukan dengan mengukur suhu ruangan di beberapa ruangan. Pengambilan suhu dilakukan sebanyak 10 kali dengan interval 1 detik untuk setiap percobaannya kemudian data pengujian suhu dicari rata-ratanya untuk dibandingkan dengan hasil pengukuran termometer ruangan digital.

A. Alat yang digunakan

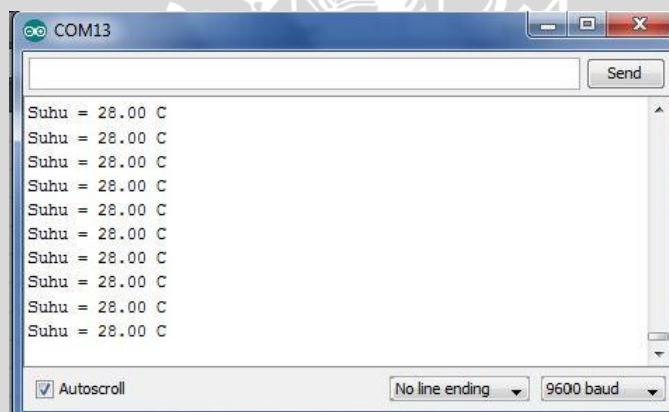
1. Komputer
2. Kabel USB
3. Kabel
4. Bread board
5. Arduino Uno R3
6. Sensor suhu DHT11 dan Termometer ruangan digital

B. Prosedur Pengujian

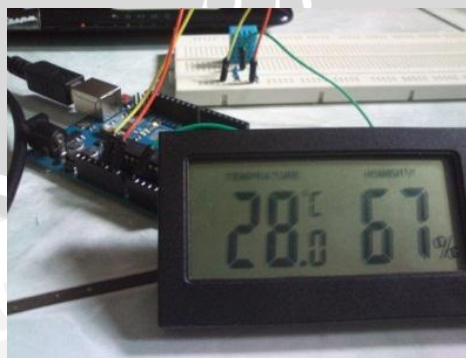
1. Hubungkan pin 1 pada DHT11 ke pin 5v arduino, pin 2 pada DHT11 ke pin A0 arduino dan pin 4 pada DHT11 ke pin *ground* arduino.
2. Hubungkan arduino dengan komputer dengan menggunakan kabel USB.
3. Buka aplikasi Arduino IDE 1.0.4 dan ketik kode program (Lampiran 2)
4. Konfigurasi pengaturan port dan *board* pada aplikasi Arduino IDE 1.0.4.
5. Klik menu *compile* kemudian *upload*, untuk mengetahui tampilan program klik tombol serial monitor.
6. Bandingkan hasil keluaran serial monitor dengan termometer pembanding.

C. Hasil dan Analisis

Gambar 6.2 menunjukkan suhu terukur dari sensor DHT11 dan Gambar 6.3 menunjukkan suhu terukur dari termometer pembanding.



Gambar 6. 2 Suhu terukur dari DHT11
Sumber: [Pengujian]



Gambar 6. 3 Suhu terukur dari termometer pembanding
Sumber: [Pengujian]

Berikut ini adalah hasil pengujian sensor suhu ditunjukkan pada Tabel 6.1

Tabel 6. 1 Pengujian Sensor Suhu

N o	DHT11											Termometer Digital	
	1	2	3	4	5	6	7	8	9	10	rata-rata		Error (°C)
1	27	27	27	27	27	27	27	27	27	27	27	27,2	0.2
2	28	28	28	28	28	28	28	28	28	28	28	28,5	0.5
3	28	28	28	28	28	28	28	28	28	28	28	29	1
4	28	27	28	28	28	28	28	28	28	28	27,9	28,5	0,6
5	26	27	27	26	26	27	27	26	26	26	26,4	26,2	0.2
6	26	26	26	26	27	26	26	26	26	26	26,1	26,1	0
7	27	27	27	27	27	27	27	27	27	27	27	26,8	0.2
8	27	27	27	27	27	27	27	27	27	27	27	27,2	0.2
9	28	28	28	28	28	28	28	28	28	28	28	27,6	0,4
10	28	28	28	28	28	28	28	28	28	28	28	28,8	0,8
											27,34	27,59	0,25

Sumber: [Pengujian]

Dari hasil pengujian sensor DHT11 yang dibandingkan dengan termometer ruangan digital pada Tabel 6.1 hasil tersebut dapat disimpulkan bahwa suhu yang terukur dari sensor DHT11 mendekati suhu terukur dari termometer ruangan digital, dengan rata-rata kesalahan $\pm 0,25$ °C.

6.1.2 Pengujian Sensor SEN0114

Tujuan pengujian sensor kelembaban tanah menggunakan sensor dfrobot SEN0114 adalah pengkalibrasian nilai yang terukur menggunakan sensor tersebut agar keluaran nilai sensor sesuai yang diinginkan, dalam hal ini nilai keluaran berupa presentase yang menunjukkan kelembaban tanah. Nilai sensor ini menjelaskan tentang kadar kelembaban, semakin besar nilai sensor berarti kelembaban semakin tinggi sedangkan semakin kecil nilai sensor maka kelembaban semakin berkurang. Berikut ini jangkauan nilai yang diharapkan yaitu:

- 0~300 : tanah kering / udara bebas
- 300~700 : tanah lembab
- 700~950 : didalam air

A. Alat yang digunakan

1. Komputer
2. Kabel USB
3. Kabel
4. Arduino Uno R3
5. Sensor kelembaban tanah dfrobot SEN0114
6. Wadah yang berisi kapas kering dan kapas basah yang dipenuhi air

B. Prosedur pengujian

1. Hubungkan pin 1 atau kabel biru pada SEN0114 ke pin A0 arduino, pin 2 atau kabel merah pada SEN0114 ke pin 5v arduino dan pin 3 atau kabel hitam pada SEN0114 ke pin *ground* arduino.
2. Hubungkan arduino dengan komputer dengan menggunakan kabel USB.
3. Buka aplikasi Arduino IDE 1.0.4 dan ketik kode program (Lampiran 3).
4. Konfigurasi pengaturan *port* dan *board* pada aplikasi Arduino IDE 1.0.4.
5. Klik menu *compile* kemudian *upload*, untuk mengetahui tampilan program klik tombol serial monitor.
6. Tancapkan sensor SEN0114 pada media yang diuji dan bandingkan hasil keluaran serial monitor untuk kedua media uji.

C. Hasil dan Analisis

Untuk mengetahui batas minimum dan maksimum pembacaan sensor SEN0114 dilakukan dua percobaan yaitu dengan mengukur kadar air nol persen (kapas kering) dan kadar air seratus persen (kapas jenuh). Berdasarkan hasil pengamatan terhadap kedua media uji, nilai sensor yang terbaca di serial monitor setiap 1 detik selama 10 detik ditunjukkan pada Tabel 6.2 berikut ini.

Tabel 6. 2 Pengujian Sensor SEN0114

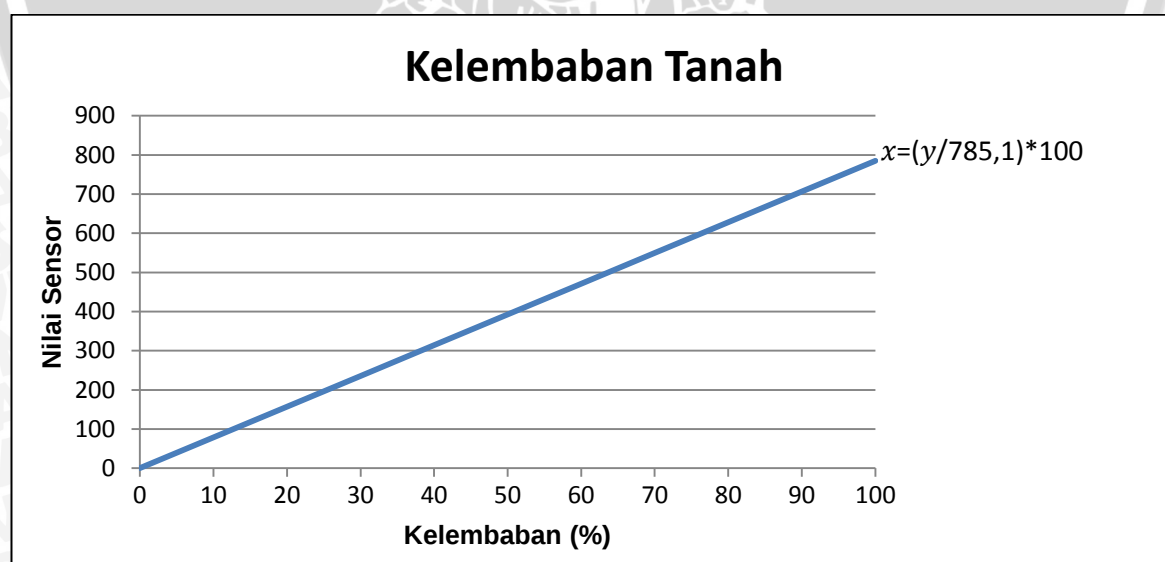
No	Kering	Basah
1	0	783
2	0	785
3	0	785
4	0	786
5	0	787
6	0	786
7	0	786
8	0	783

9	0	784
10	0	786
rata-rata	0	785,1

Sumber: [Pengujian]

Nilai yang terbaca di monitor diamati selama 10 detik sehingga didapat 10 nilai sensor. Pengulangan sebanyak 10 kali ini dilakukan agar nilai yang didapat memiliki presisi dan akurasi yang tinggi. Setelah didapat 10 nilai sensor, maka diambil 2 nilai rata-rata sebagai nilai acuan dalam pembuatan kurva linear. Kurva ini digunakan dalam penentuan nilai persentasi kelembaban yang sebenarnya. Hal ini dilakukan karena nilai yang terbaca pada sensor tidak mencerminkan nilai kelembaban yang sebenarnya, sehingga nilai sensor yang terbaca harus dikalibrasi terlebih dahulu melalui kurva kalibrasi agar nilai sensor yang terbaca dapat langsung mencerminkan persentasi kelembabannya.

Berdasarkan Tabel 6.2 diatas nilai sensor 0 merupakan nilai rata-rata dari 10 nilai sensor kapas kering. Persentase 0% merupakan asumsi bahwa pada kapas kering tidak terdapat kandungan air sama sekali sehingga dianggap persentase kelembabannya 0. Nilai sensor 785,1 merupakan nilai rata-rata dari 10 nilai sensor kapas jenuh. Persentase 100% merupakan asumsi bahwa kapas berada pada kondisi jenuh yang memiliki kandungan persentase kelembaban 100.



Gambar 6. 4 Grafik kurva kalibrasi kelembaban tanah
Sumber: [Pengujian]

Grafik yang ditunjukkan Gambar 6.4 merupakan hasil plot dari Tabel 6.2. Grafik tersebut digunakan sebagai kurva kalibrasi untuk menentukan persentase kelembaban dari nilai sensor yang didapat. Karena data yang di plot hanya 2, baik pada sumbu x maupun y, maka kurva yang didapat linear. Persamaan linear yang didapat digunakan untuk menentukan persentase kelembaban. Persentase kelembaban merupakan variabel x yang didapat dari persamaan linear pada grafik diatas.

Tabel 6. 3 Nilai kelembaban dalam rentang 0-100 %

No	Persentase (%)	Nilai Sensor
1	0	0
2	10	78,51
3	20	157,02
4	30	235,53
5	40	314,04
6	50	392,55
7	60	471,06
8	70	549,57
9	80	628,08
10	90	706,59
11	100	785,1

Sumber: [Pengujian]

6.1.3 Pengujian Relay

Jenis relay yang digunakan dalam penelitian ini adalah relay jenis SPDT dengan catu daya 5v untuk mengaktifkan *arpature* relay. Rangkaian relay ini digunakan sebagai saklar elektrik, apabila pada inputan terdapat tegangan 5v maka relay berfungsi sebagai saklar tertutup. Dan apabila input 0v maka relay berfungsi sebagai saklar tertutup. Tujuan dari pengujian relay adalah untuk mengetahui kinerja relay.

A. Alat yang digunakan

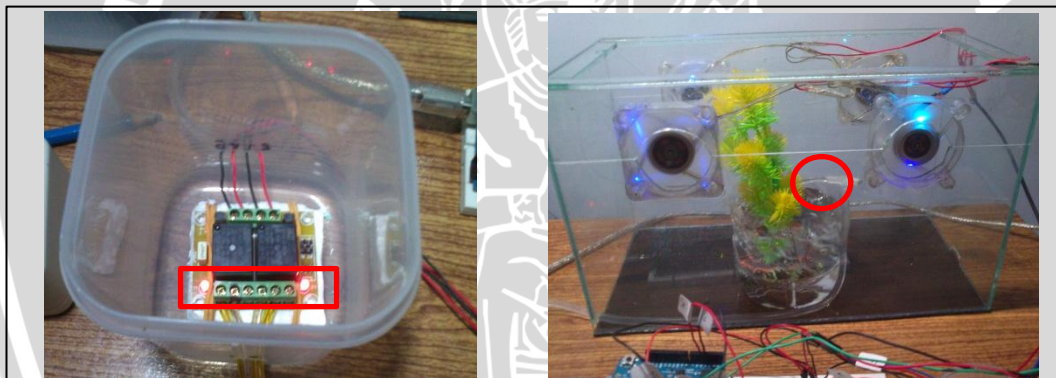
1. Komputer
2. Kabel USB
3. Kabel
4. 2 Relay SPDT
5. Arduino Uno R3
6. Kipas
7. Pompa air

B. Prosedur pengujian

1. Hubungkan kabel *ground* pada relay ke pin *ground* arduino, kabel 5v pada relay ke pin 5v arduino, kabel relay 1 ke pin D4 arduino, kabel relay 2 ke pin D5 arduino, pin NO1 ke aktuator 1, dan pin NO ke aktuator 2.
2. Hubungkan arduino dengan komputer dengan menggunakan kabel USB.
3. Buka aplikasi Arduino IDE 1.0.4 dan ketik kode program (Lampiran 4)
4. Konfigurasi pengaturan *port* dan *board* pada aplikasi Arduino IDE 1.0.4.
5. Klik menu *compile* kemudian *upload*, amati lampu indikator pada relay dan kedua aktuator yang menandakan bahwa relay aktif.

C. Hasil dan Analisa

Berikut ini adalah hasil pengujian relay ketika aktif ditunjukkan pada Gambar 6.5 dan relay ketika mati ditunjukkan pada Gambar 6.6.



Gambar 6. 5 Relay aktif
Sumber: [Pengujian]



Gambar 6. 6 Relay mati
Sumber: [Pengujian]

Pada saat kedua relay diberi tegangan 5v, led indikator pada kedua relay menyala merah yang menandakan bahwa relay aktif sehingga mengakibatkan kedua aktuator menyala. Aktuator menyala ditandai dengan berputarnya kipas, led pada kipas menyala biru dan pompa air mengeluarkan air. Sedangkan saat relay tidak diberi tegangan atau *low*, kondisi aktuator menjadi tidak aktif mengakibatkan kedua aktuator mati.

Dari hasil pengujian ini disimpulkan bahwa rangkaian relay akan aktif ketika relay diberi tegangan *high* sebesar 5v yang mengakibatkan saklar tertutup sehingga led indikator dan kedua aktuator menyala. Relay akan mati atau saklar menjadi terbuka jika rangkaian relay tidak diberi tegangan atau kondisi input *low*.

6.1.4 Pengujian WiFi Shield

Tujuan dari Pengujian unit WiFi Shield adalah pengujian untuk menganalisis apakah WiFi Shield dapat bekerja dengan baik. Selain pengujian tersebut adalah menguji tingkat kecepatan WiFi Shield untuk melakukan koneksi dengan Access Point dan berkomunikasi dengan *datasink*. Pada pengujian ini dilakukan sebanyak 10 kali percobaan.

A. Alat yang digunakan

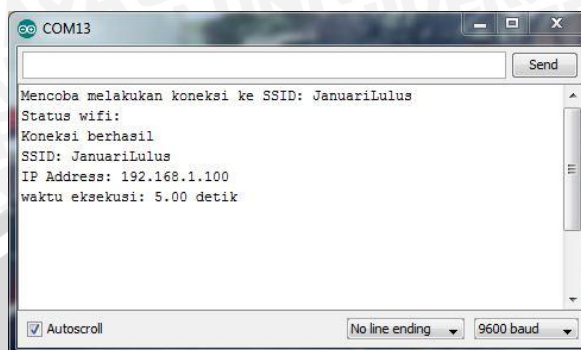
1. Komputer
2. Kabel USB
3. Kabel
4. WiFi Shield
5. Arduino Uno R3

B. Prosedur pengujian

1. Pasang WiFi shield ke arduino uno
2. Hubungkan arduino dengan komputer dengan menggunakan kabel USB.
3. Buka aplikasi Arduino IDE 1.0.4 dan ketik kode program (Lampiran 5).
4. Konfigurasi pengaturan *port* dan *board* pada aplikasi Arduino IDE 1.0.4.
5. Klik menu *compile* kemudian *upload*, untuk mengetahui tampilan program klik tombol serial monitor.

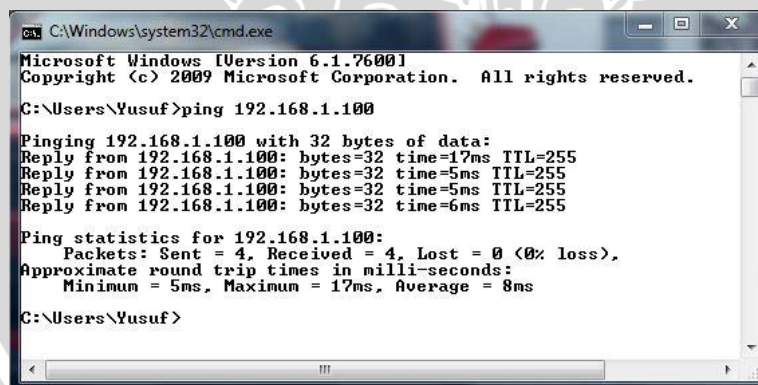
C. Hasil dan Analisis

Tampilan program pengujian konektifitas WiFi *shield* ditunjukkan pada Gambar 6.7.



Gambar 6. 7 Tampilan program pengujian WiFi shield
Sumber: [Pengujian]

Berdasarkan Gambar 6.7 tampilan program pengujian WiFi *shield* menunjukkan bahwa *node* sensor telah berhasil terkoneksi dengan *access point* dengan SSID bernama “JanuariLulus”, kemudian *node* sensor mendapatkan alamat ip 192.168.1.100 dengan waktu koneksi 5 detik. Waktu koneksi adalah waktu yang ditempuh oleh *node* sensor dari posisi mati atau tidak ada daya listrik sampai *node* sensor mendapatkan alamat ip.



Gambar 6. 8 Pengujian konektifitas datasink dengan WiFi shield
Sumber: [Pengujian]

Pengujian berikutnya dilakukan dengan menguji konektifitas *node* sensor dengan *datasink*. Berdasarkan Gambar 6.8 *datasink* dengan alamat ip 192.168.1.107 telah berhasil berhubungan dengan *node* sensor dengan alamat ip 192.168.1.100 melalui perintah “ping” yang disertai alamat ip *node* sensor. Dari hasil pengujian 4 buah permintaan ICMP *echo request* telah berhasil dijawab

dengan baik oleh *node* sensor dengan waktu yang dihasilkan relatif kecil. Hal ini menunjukkan bahwa *node* sensor sedang berada dalam satu jaringan bersama-sama dengan *datasink* atau komputer dan *node* sensor mampu menangani ICMP *echo request* yang diberikan kepadanya dengan nilai kesuksesan paket data terkirim 100%. Pada percobaan konektifitas *node* sensor dengan *access point* dilakukan sebanyak 10 kali percobaan dengan cara *node* sensor di *reset* atau *restart*. Dapat disimpulkan bahwa rata-rata waktu koneksi *node* sensor dengan *access point* sebesar 5 detik dengan nilai kesuksesan sebesar 100%. Pada pengujian konektifitas WiFi *shield* dilakukan percobaan sebanyak 10 kali, dari hasil pengujian tersebut semua percobaan berhasil terkoneksi dengan rata-rata waktu yang dibutuhkan sebesar 5 detik. Hasil pengujian WiFi *shield* ditunjukkan pada Tabel 6.4.

Tabel 6. 4 Hasil pengujian WiFi *shield*

No	Waktu Koneksi dengan Access Point	Status
1.	5 detik	Terhubung
2.	5 detik	Terhubung
3.	5 detik	Terhubung
4.	5 detik	Terhubung
5.	5 detik	Terhubung
6.	5 detik	Terhubung
7.	5 detik	Terhubung
8.	5 detik	Terhubung
9.	5 detik	Terhubung
10	5 detik	Terhubung

Sumber: [Pengujian]

6.2 Pengujian Sistem Secara Keseluruhan

Pengujian sistem secara keseluruhan merupakan gabungan dari pengujian unit yang saling berhubungan. Pengujian ini dilakukan untuk mengetahui integritas antar unit dalam satu sistem dengan tujuan sistem yang dibangun sesuai dengan perancangan sistem. Pengujian ini meliputi pengujian waktu yang dibutuhkan oleh WiFi *shield* untuk terkoneksi dengan jaringan nirkabel, pengujian aktuator terhadap nilai sensor, pengujian ketepatan data transfer dan kebenaran data terkirim.

6.2.1 Pengujian Konektifitas Jaringan Nirkabel pada *Node* sensor

A. Alat dan bahan yang digunakan

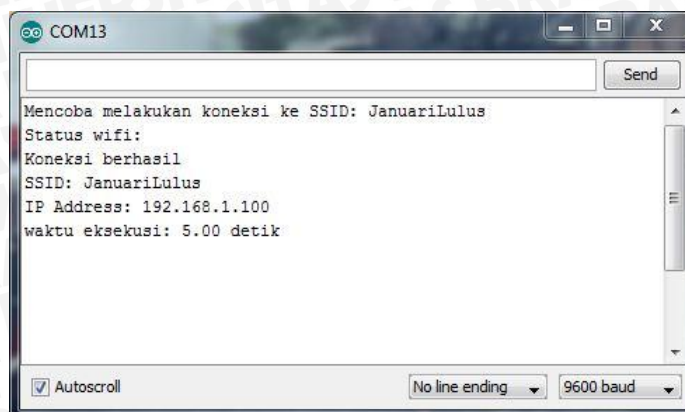
1. Komputer
2. Kabel USB
3. Kabel
4. Arduino Uno R3
5. *WiFi Shield*
6. Relay
7. *Bread board*
8. Sensor kelembaban tanah dfrobot SEN0114
9. Sensor suhu DHT11
10. Kipas dan pompa air akuarium
11. Wadah yang berisi kapas kering dan kapas basah yang dipenuhi air

B. Prosedur pengujian

1. Pasang perangkat keras *node* sensor sesuai dengan Subbab 4.3 Perancangan Perangkat *Node* sensor.
2. Hubungkan arduino dengan komputer dengan menggunakan kabel USB.
3. Buka aplikasi Arduino IDE 1.0.4 dan ketik kode program (Lampiran 1)
4. Konfigurasi pengaturan *port* dan *board* pada aplikasi Arduino IDE 1.0.4.
5. Klik menu *compile* kemudian *upload*, untuk mengetahui tampilan program klik tombol serial monitor.

C. Hasil dan Analisis

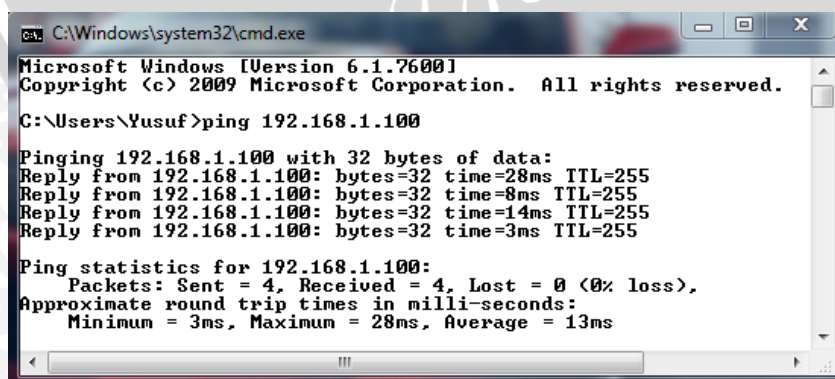
Pada pengujian ini *node* sensor terkoneksi dengan *access point* “JanuariLulus” dengan alamat ip diperoleh dari DHCP. Tampilan program arduino dapat dilihat menggunakan fitur serial monitor yang tersedia di aplikasi Arduino IDE. Tampilan program ditunjukkan pada Gambar 6.9.



Gambar 6. 9 Tampilan status WiFi shield
Sumber: [Pengujian]

Gambar 6.9 diatas memberikan informasi mengenai status WiFi *shield* pada *node* sensor. Informasi tersebut meliputi nama SSID yang terkoneksi yaitu “JanuariLulus” dengan alamat IP *node* sensor 192.168.1.100, waktu yang dibutuhkan untuk koneksi dengan SSID yaitu 5 detik. Waktu 5 detik ini dapat diartikan sebagai waktu yang digunakan oleh *node* sensor untuk menghidupkan perangkat dari kondisi mati sampai terkoneksi dengan SSID.

Setelah perangkat *node* sensor terkoneksi dengan jaringan dengan komputer *server*, langkah selanjutnya adalah menguji koneksi antara komputer dengan *node* sensor. Pengujian dengan menggunakan *command* “ping” melalui aplikasi cmd. Pada pengujian ini komputer *server* mendapatkkan alamat ip 192.168.1.112. Hasil pengujian pada Gambar 6.10 menunjukkan bahwa *node* sensor berhasil menjawab ICMP *echo request* dengan persentase *packet loss* sebesar 0% dan komputer *server* terhubung ke *node* sensor dengan baik.



Gambar 6. 10 Pengujian konektifitas *node* sensor dengan komputer *server*
Sumber: [Pengujian]

6.2.2 Pengujian Aktuator Terhadap Nilai Sensor

Pengujian ini bertujuan untuk melakukan pengecekan pada aktuator. Dimana tiap-tiap kondisi akan menghasilkan output yang berbeda sehingga aktuator akan bereaksi secara berbeda-beda juga. Pada pengujian ini terdapat 4 skenario uji yang ditunjukkan pada Tabel 6.5.

Tabel 6. 5 Jangkauan Nilai Sensor Untuk Masing-masing Kondisi

No.	Kondisi		Keterangan	Aktuator
	Suhu (°C)	Kelembaban (%)		
1.	>= 28	>= 75	Suhu Panas	Kipas On Sprayer Off
2.	>= 28	< 75	Suhu Panas, Kurang Air	Kipas On Sprayer On
3.	< 28	< 75	Kurang Air	Kipas Off Sprayer On
4.	< 28	> 75	Normal	Kipas Off Sprayer Off

Sumber: [Pengujian]

A. Alat dan bahan yang digunakan

1. Komputer
2. Kabel USB
3. Kabel
4. Arduino Uno R3
5. WiFi Shield
6. Relay
7. Bread board
8. Sensor kelembaban tanah dfrobot SEN0114
9. Sensor suhu DHT11
10. Kipas dan pompa air akuarium
11. Wadah yang berisi kapas kering dan kapas basah yang dipenuhi air
12. Blower / pemanas

B. Prosedur pengujian

1. Pasang perangkat keras *node* sensor sesuai dengan Subbab 4.3 Perancangan Perangkat *Node* sensor.
2. Hubungkan arduino dengan komputer dengan menggunakan kabel USB.
3. Buka aplikasi Arduino IDE 1.0.4 dan ketik kode program (Lampiran 1).

4. Konfigurasi pengaturan *port* dan *board* pada aplikasi Arduino IDE 1.0.4.
5. Klik menu *compile* kemudian *upload*, untuk mengetahui tampilan program klik tombol serial monitor.
6. Lakukan beberapa manipulasi kondisi sensor dengan memberikan perlakuan seperti memberikan suhu panas dengan menggunakan blower dan memindahkan sensor SEN0114 ke beberapa media yang memiliki tingkat kelembaban yang berbeda-beda.

C. Hasil dan Analisis

Dengan memberikan perlakuan terhadap sensor seperti yang dilakukan dalam prosedur pengujian, sensor mampu merespon dengan baik terhadap perubahan kondisi lingkungan. Pada pengujian ini sebelumnya telah ditentukan beberapa batasan untuk suhu dan kelembaban. Diasumsikan untuk batas tertinggi suhu diset dengan angka 28 dan batas terendah untuk kelembaban diset dengan angka 75. Dari hasil pengamatan terhadap respon yang diberikan aktuator terhadap nilai sensor diambil beberapa sampel hasil pengujian yang ditunjukkan pada Tabel 6.6.

Tabel 6. 6 Hasil Pengujian Aktuator

No.	Kondisi		Aktuator	Keterangan
	Suhu (°C)	Kelembaban (%)		
1.	25	62	Sprayer On	Valid
2.	25	64	Sprayer On	Valid
3.	25	65	Sprayer On	Valid
4.	26	62	Sprayer On	Valid
5.	27	65	Sprayer On	Valid
6.	28	64	Sprayer On & Kipas On	Valid
7.	29	64	Sprayer On & Kipas On	Valid
8.	29	66	Sprayer On & Kipas On	Valid
9.	30	63	Sprayer On & Kipas On	Valid
10.	30	67	Sprayer On & Kipas On	Valid
11.	28	86	Kipas On	Valid
12.	28	86	Kipas On	Valid
13.	28	86	Kipas On	Valid
14.	28	86	Kipas On	Valid
15.	28	87	Kipas On	Valid
16.	27	87	Sprayer Off & Kipas Off	Valid
17.	27	87	Sprayer Off & Kipas Off	Valid
18.	27	88	Sprayer Off & Kipas Off	Valid
19.	27	88	Sprayer Off & Kipas Off	Valid
20.	27	87	Sprayer Off & Kipas Off	Valid

Sumber: [Pengujian]

Pada hasil pengujian no 1-5 dilakukan pengujian dengan cara memindahkan sensor kelembaban ke media yang memiliki kelembaban lebih rendah. Didapatkan hasil bahwa nilai sensor kelembaban dibawah 75 dan suhu dibawah 28, pada perlakuan ini dapat dihasilkan aktuator aktif berupa sprayer menyala dan kipas mati. Pada hasil pengujian 6-10 dilakukan pengujian dengan cara memindahkan sensor kelembaban ke media yang memiliki kelembaban lebih rendah dan sensor suhu diberi perlakuan pemberian suhu panas dengan menggunakan blower. Didapatkan hasil bahwa nilai sensor kelembaban dibawah 75 dan suhu diatas 28, pada perlakuan ini dapat dihasilkan aktuator aktif berupa sprayer dan kipas menyala. Pada hasil pengujian 11-15 dilakukan pengujian dengan cara memberikan perlakuan pemberian suhu panas kepada sensor suhu dengan menggunakan blower. Didapatkan hasil bahwa nilai sensor suhu diatas 28, pada perlakuan ini dapat dihasilkan aktuator aktif berupa kipas menyala. Sedangkan pada hasil pengujian 16-20 sensor tidak diberi perlakuan dan kondisi suhu serta kelembaban dalam kondisi ideal sehingga kedua aktuator berada pada posisi mati.

Dari hasil pengujian diatas sensor suhu membaca nilai suhu masih dalam batas normal yaitu dibawah 27 dan sensor kelembaban tanah membaca nilai kelembaban kurang dari 75, sehingga aktuator yang aktif hanya sprayer dengan mengeluarkan air. Pengujian aktuator terhadap nilai sensor memberikan hasil yang diharapkan yaitu sensor mampu merespon dengan baik terhadap perubahan kondisi lingkungan disekitar sensor dan aktuator mampu berjalan dengan baik dengan memberikan beberapa tindakan terhadap perubahan nilai sensor.

6.2.3 Pengujian Ketepatan Data Transfer dan Kebenaran Data Terkirim

Pengujian ini bertujuan untuk menguji sisi fungsional dan validasi sistem. Dari segi fungsionalnya apakah sistem dalam proses pengiriman data sensor dari *node* sensor ke *server* berhasil atau tidak mengirimkan data. Sisi validasinya apakah nilai sensor yang diterima oleh *server* nilainya sama dengan nilai yang dikirim *node* sensor.

A. Alat dan bahan yang digunakan

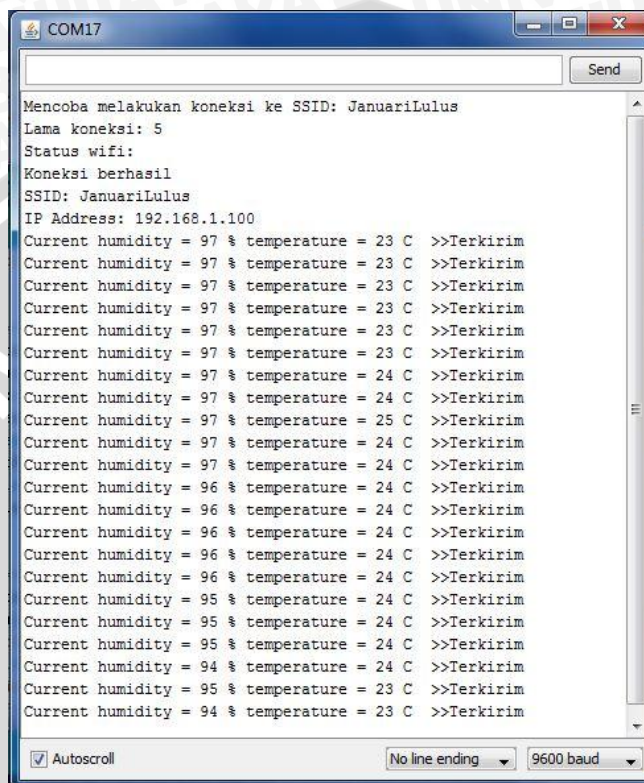
1. Komputer
2. Kabel USB
3. Kabel
4. Arduino Uno R3
5. *WiFi Shield*
6. Relay
7. *Bread board*
8. Sensor kelembaban tanah dfrobot SEN0114
9. Sensor suhu DHT11
10. Kipas dan pompa air akuarium
11. Wadah yang berisi kapas kering dan kapas basah yang dipenuhi air
12. Blower / pemanas

B. Prosedur pengujian

1. Pasang perangkat keras *node* sensor sesuai dengan Subbab 4.3 Perancangan Perangkat *Node* sensor.
2. Hubungkan arduino dengan komputer dengan menggunakan kabel USB.
3. Buka aplikasi Arduino IDE 1.0.4 dan ketik kode program (terlampir)
4. Konfigurasi pengaturan *port* dan *board* pada aplikasi Arduino IDE 1.0.4.
5. Aktifkan firur Apache dan Mysql pada aplikasi xampp.
6. Klik menu *compile* kemudian *upload*, untuk mengetahui tampilan program klik tombol serial monitor.
7. Lakukan beberapa manipulasi kondisi sensor dengan memberikan perlakuan seperti memberikan suhu panas dengan menggunakan blower dan memindahkan sensor SEN0114 ke beberapa media yang memiliki tingkat kelembaban yang berbeda-beda.
8. Amati data yang ditampilkan pada serial monitor dan database.

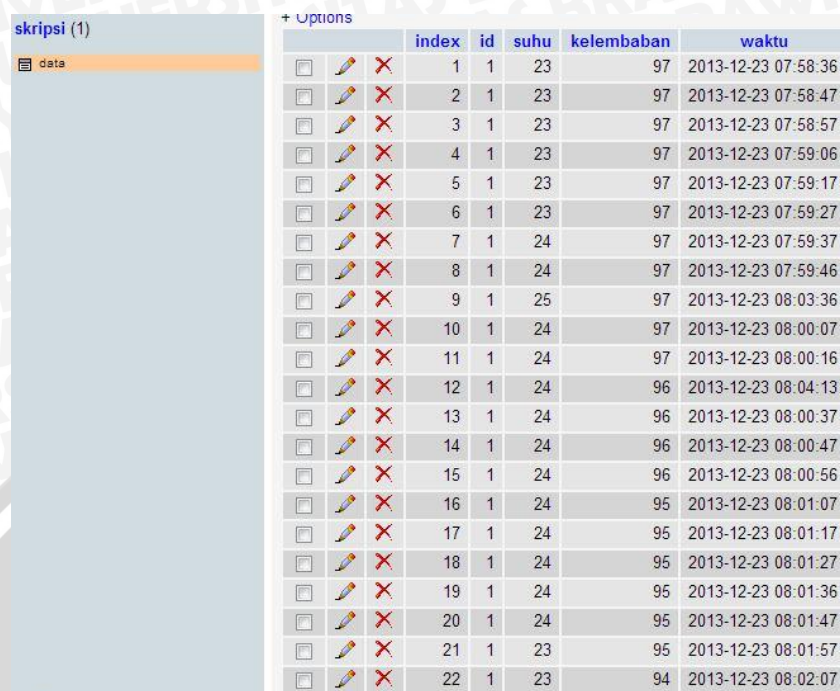
C. Hasil dan Analisis

Dengan perlakuan pada sensor suhu dan kelembaban mengakibatkan perubahan nilai pembacaan sensor. Nilai-nilai yang dibaca sensor dapat diamati dengan menggunakan serial monitor seperti Gambar 6.11.



**Gambar 6. 11 Perubahan nilai-nilai sensor pada serial monitor
Sumber: [Pengujian]**

Berdasarkan Gambar 6.11 pengujian dilakukan sebanyak 22 kali dan semua data berhasil terkirim dari *node* sensor, hal ini ditandai dengan notifikasi pada serial monitor berupa “>>Terkirim”. Untuk mengetahui data yang diterima oleh server dapat diamati pada database server. Gambar 6.12 menunjukkan data-data sensor yang diterima dan tersimpan pada database *server*.



	index	id	suhu	kelembaban	waktu
☐	1	1	23	97	2013-12-23 07:58:36
☐	2	1	23	97	2013-12-23 07:58:47
☐	3	1	23	97	2013-12-23 07:58:57
☐	4	1	23	97	2013-12-23 07:59:06
☐	5	1	23	97	2013-12-23 07:59:17
☐	6	1	23	97	2013-12-23 07:59:27
☐	7	1	24	97	2013-12-23 07:59:37
☐	8	1	24	97	2013-12-23 07:59:46
☐	9	1	25	97	2013-12-23 08:00:36
☐	10	1	24	97	2013-12-23 08:00:07
☐	11	1	24	97	2013-12-23 08:00:16
☐	12	1	24	96	2013-12-23 08:04:13
☐	13	1	24	96	2013-12-23 08:00:37
☐	14	1	24	96	2013-12-23 08:00:47
☐	15	1	24	96	2013-12-23 08:00:56
☐	16	1	24	95	2013-12-23 08:01:07
☐	17	1	24	95	2013-12-23 08:01:17
☐	18	1	24	95	2013-12-23 08:01:27
☐	19	1	24	95	2013-12-23 08:01:36
☐	20	1	24	95	2013-12-23 08:01:47
☐	21	1	23	95	2013-12-23 08:01:57
☐	22	1	23	94	2013-12-23 08:02:07

Gambar 6. 12 Data-data sensor pada database
Sumber: [Pengujian]

Pada Gambar 6.12 tabel data pada database skripsi memiliki 5 kolom yang terdiri dari kolom index, id, suhu, kelembaban dan waktu. Pada kolom *index* menjelaskan tentang nomor baris data pada tabel tersebut. Kolom id berisi tentang id *node* sensor, karena pada pengujian ini *node* sensor hanya berjumlah satu maka semua isi kolom pada tabel tersebut berisi nilai 1 semua. Kolom suhu dan kelembaban berisi tentang informasi data suhu dan kelembaban yang dibaca dan dikirim oleh *node* sensor. Kolom waktu berisi tentang informasi waktu data-data sensor diterima oleh *server*.

Berdasarkan hasil pengujian yang ditunjukkan pada Gambar 6.11 Perubahan nilai-nilai sensor pada serial monitor dan Gambar 6.12 Data-data sensor pada database, dapat dibandingkan tentang kedua hasil yang ditunjukkan pada Tabel 6.7.

Tabel 6. 7 Hasil pengujian integrasi

No	Node sensor		Server			Ket
	Suhu (°C)	Kelembaban (%)	Suhu	Kelembaban	Waktu	
1.	23	97	23	97	2013-12-23 07:58:36	Terkirim
2.	23	97	23	97	2013-12-23 07:58:47	Terkirim
3.	23	97	23	97	2013-12-23 07:58:57	Terkirim
4.	23	97	23	97	2013-12-23 07:59:06	Terkirim
5.	23	97	23	97	2013-12-23 07:59:17	Terkirim
6.	23	97	23	97	2013-12-23 07:59:27	Terkirim
7.	24	97	24	97	2013-12-23 07:59:37	Terkirim
8.	24	97	24	97	2013-12-23 07:59:46	Terkirim
9.	25	97	25	97	2013-12-23 07:59:57	Terkirim
10.	24	97	24	97	2013-12-23 08:00:07	Terkirim
11.	24	97	24	97	2013-12-23 08:00:16	Terkirim
12.	24	96	24	96	2013-12-23 08:00:27	Terkirim
13.	24	96	24	96	2013-12-23 08:00:37	Terkirim
14.	24	96	24	96	2013-12-23 08:00:47	Terkirim
15.	24	96	24	96	2013-12-23 08:00:56	Terkirim
16.	24	96	24	96	2013-12-23 08:01:07	Terkirim
17.	24	95	24	95	2013-12-23 08:01:17	Terkirim
18.	24	95	24	95	2013-12-23 08:01:27	Terkirim
19.	24	95	24	95	2013-12-23 08:01:36	Terkirim
20.	24	94	24	94	2013-12-23 08:01:47	Terkirim
21.	23	95	23	95	2013-12-23 08:01:57	Terkirim
22.	23	94	23	94	2013-12-23 08:02:07	Terkirim

Sumber: [Pengujian]

Dari hasil pengujian yang ditunjukkan pada Tabel 6. 7 Hasil pengujian integrasi, data yang dikirim oleh *node* sensor dapat diterima oleh *server* dan disimpan pada database. Dari sisi fungsionalitas sistem ini dapat berjalan dengan baik, sedangkan dari sisi validitas data sistem ini dapat mengirimkan data dengan tingkat validitas 100%.



BAB VI PENUTUP

6.1 KESIMPULAN

Berdasarkan implementasi dan hasil pengujian dari Sistem Pengendalian Suhu dan Kelembaban Berbasis *Wireless Embedded System*, maka didapatkan kesimpulan sebagai berikut :

1. Pengujian unit sensor suhu DHT 11 bekerja dengan baik untuk mengukur suhu ruangan dengan nilai kesalahan kurang dari 2 °C sesuai dengan *datasheet* DHT11.
2. Pengujian unit sensor kelembaban tanah SEN0114 bekerja dengan baik.
3. Pengujian unit WiFi *shield* dapat digunakan untuk terhubung dengan jaringan dengan waktu yang dibutuhkan untuk melakukan koneksi rata-rata 5 detik.
4. Aktuator dapat berjalan dengan baik dengan memberikan respon tergantung pada batas nilai sensor yang telah ditentukan.
5. Dalam proses pengiriman data-data sensor dari *node* sensor ke *server* dapat berjalan dengan baik
6. Data-data hasil pengukuran sensor dapat disimpan pada database.

6.2 SARAN

Saran yang dapat diberikan untuk pengembangan perangkat lunak ini antara lain :

1. Untuk pengembangan lebih lanjut, disarankan untuk mengolah data-data sensor kedalam sistem pakar
2. Untuk pengembangan lebih lanjut, disarankan untuk menambah sensor dan aktuator.
3. Untuk pengembangan lebih lanjut, disarankan untuk menguji sensor kelembaban tanah dengan alat ukur digital.
4. Untuk pengembangan lebih lanjut, disarankan menggunakan *node* sensor lebih dari satu.

DAFTAR PUSTAKA

- [ARD-13] “Arduino”, <http://www.arduino.cc> [Online; 1 September 2013].
- [CLA-07] Reese, John. 2007. “eBookwholesaler”, *e-Book*.
- [MCE-10] Ogata, Katsuhiko. 2010. “Modern Control Engineering”. New Jersey, Pearson Education, Inc.
- [DDU-07] Donny K. Sutantyo, Darmawan Utomo. 2007. *Implementasi Embedded Web Server Via Modem Berbasis Mikrokontroler*. Fakultas Teknik Jurusan Teknik Elektro, Universitas Kristen Satya Wacana, Salatiga.
- [DSD-13] “Datasheet DHT11”, <http://goo.gl/IRZV2> [Online; 1 September 2013].
- [DSW-13] “Arduino WiFi shield”, <http://goo.gl/IRZV2> [Online; 1 September 2013].
- [IAL-12] Artanto, Dian. 2012. “Interaksi Arduino dan LabView”, Penerbit PT Elex Media Komputindo, Jakarta.
- [ISW-08] Sibarani, Marimbun. 2008. *Implementasi Sistem Wireless Sensor Network Berbasis Internet Protocol (IP) Untuk Pemantauan Tingkat Polusi Udara*. Departemen Teknik Elektro, Universitas Indonesia, Depok.
- [MIK-11] “Budidaya tomat” <http://epetani.deptan.go.id/budidaya/budidaya-tomat-1652> [Online; 17 November 2013].
- [MOD-12] *Modul Praktikum Jaringan Komputer*. 2012. PTIIK - Universitas Brawijaya, Malang.
- [PTA-08] Kholifah, Anik Nur. 2008. *Sistem Pengendalian Ruang Tanaman Anggrek Bulan Berbasis Microcontroller*. Sistem Komputer, STIKOM, Surabaya.
- [RIS-12] “Waktu nyata”, http://id.wikipedia.org/wiki/Waktu_nyata [Online; 20 Agustus 2013].
- [RLY-13] “Relay”, <http://goo.gl/lhvDbr> [Online; 1 September 2013].
- [SDS-13] Eva Fatra LT, Desi dkk. “Pengenalan Sensor Kelembaban Tanah VN400 dan SEN0057 dan Aplikasinya pada Pengukuran

Kelembaban Tanah Kering dan Jenuh”. Departemen Teknik Sipil dan Lingkungan, Institut Pertanian Bogor, Bogor.

[SSM-13] “Datasheet Sensor SEN0114”, <http://goo.gl/VzzTe1> [Online; 17 November 2013].

[SYT-13] “Sistem Tertanam”, <http://goo.gl/ZaUDu7> [Online; 25 Agustus 2013].

[TMT-13] “ Menanam tomat tanpa pupuk dasar ”, <http://goo.gl/5ozpSD> [Online; 17 November 2013].

[WSN-10] Dargie, Waltenegus. 2010. “Fundamentals of wireless sensor networks : theory and practice”. Indianapolis Canada : Wiley Publishing, Inc Publication.

[WAG-05] Wagito, “Jaringan Komputer, Teori dan Implementasi Berbasis Linux” Penerbit Gaya Media, 2005



LAMPIRAN

Lampiran 1 : Source Code Node Sensor

```
#include <SPI.h>
#include <WiFi.h>
#include <dht.h>
#include <Time.h>
#include <TimeAlarms.h>
#define soil_pin A1
#define dht_dpin A0

String data;
int suhuRuang, lembabTanah, idSensor;
int kipas=5;// kipas
int sprayer=4;//sprayer
int status = WL_IDLE_STATUS;
char ssid[] = "JanuariLulus"; //
IPAddress server(192,168,1,100);
dht DHT;
WiFiClient client;

//===== Fungsi Setup akan berjalan sekali setiap
arduino menyala atau reset
void setup()
{
    idSensor=1;//mendeklarasikan sensor node 1
    pinMode(sprayer,OUTPUT);//pendefinisian pin sprayer sebagai output
    pinMode(kipas,OUTPUT);//pendefinisian pin kipas sebagai output
    pinMode(sprayer,OUTPUT);//pendefinisian pin sprayer sebagai
    output
    Serial.begin(9600);//set baudrate untuk komunikasi serial
    setTime(8,29,0,1,1,11);//set waktu
    Alarm.timerRepeat(10, Repeats);//mengeset waktu untuk memanggil
    fungsi repeat
    while ( status != WL_CONNECTED) {
        Serial.print("Mencoba melakukan koneksi ke SSID: ");
        Serial.println(ssid);
        status = WiFi.begin(ssid);//melakukan koneksi sesuai dengan
        nama SSID
        Serial.print("Lama koneksi: ");
        Serial.println(millis()/1000);
    }
    printWiFiStatus();
}

//===== fungsi yang dijalankan terus menerus selama arduino
menyala
void loop(){
    aktuator();
    Alarm.delay(1000);//mengeset delay waktu 1 detik
}

//===== fungsi yang digunakan untuk membaca nilai sensor
void bacaSensor()
```

```
{
  DHT.read11(dht_dpin); //pendefinisian pembacaan sensor DHT11
  dengan pin A0
  suhuRuang = DHT.temperature;//Variable suhu
  lembabTanah = analogRead(soil_pin)*0.127;//kalibrasi sensor
  SEN0114 (kelembaban tanah)
}

//===== fungsi yang digunakan untuk menampilkan
informasi melalui komunikasi serial
void SerialEvent()
{
  bacaSensor();
  Serial.print("Current humidity = ");
  Serial.print(lembabTanah);
  Serial.print(" % ");
  Serial.print("temperature = ");
  Serial.print(suhuRuang);
  Serial.print(" C ");
}

//===== fungsi yang digunakan untuk mpengkondisian
void aktuator()
{
  bacaSensor();
  if(suhuRuang >= 28 && lembabTanah >= 75)
  {
    SerialEvent();
    Serial.println(">>kipas on");
    digitalWrite(kipas, HIGH);
    digitalWrite(sprayer, LOW);
  }
  else if (suhuRuang >= 28 && lembabTanah < 75)
  {
    SerialEvent();
    Serial.println(">>spray on kipas on");
    digitalWrite(kipas, HIGH);
    digitalWrite(sprayer, HIGH);
  }
  else if(suhuRuang < 28 && lembabTanah < 75)
  {
    SerialEvent();
    Serial.println(">>spray on ");
    digitalWrite(kipas, LOW);
    digitalWrite(sprayer, HIGH);
  }
  else
  {
    SerialEvent();
    Serial.println();
    digitalWrite(kipas, LOW);
    digitalWrite(sprayer, LOW);
  }
}

//===== fungsi yang digunakan untuk menjalankan perintah
```

```

tiap periode waktu (10detik)
void Repeats(){
    kirim_data();
    digitalClockDisplay(); //memanggil fungsi tampilan waktu
}

//===== fungsi yg digunakan untuk mencetak
waktu
void digitalClockDisplay()
{
    Serial.print(hour());
    printDigits(minute());
    printDigits(second());
    Serial.println();
}

void printDigits(int digits)
{
    Serial.print(":");
    if(digits < 10)
        Serial.print('0');
    Serial.print(digits);
}

//===== fungsi yg digunakan untuk memberikan
informasi mengenai status WiFi
void printWiFiStatus() {
    Serial.println("Status WiFi: ");
    Serial.println("Koneksi berhasil ");
    Serial.print("SSID: ");
    Serial.println(WiFi.SSID());
    IPAddress ip = WiFi.localIP();
    Serial.print("IP Address: ");
    Serial.println(ip);
}

//===== fungsi yang digunakan untuk membentuk formatan
yg digunakan method post
void paket_data(){
    bacaSensor();
    data+="id="; //mengisi ke field
    data+=idSensor;//menyimpan data idSensor
    data+="&kelembaban=";//mengisi ke field
    data+=lembabTanah;//menyimpan data kelembaban tanah
    data+="&suhu=";//mengisi ke field
    data+=suhuRuang;//menyimpan data suhu
}

//===== fungsi pengiriman data ke server
void kirim_data(){

    if (client.connect(server, 80)) {
        paket_data();
        SerialEvent();
        client.println("POST /skripsi/insert_mysql.php HTTP/1.1");
        client.println("Host: www.yourdomain.com");
        client.println("User-Agent: Arduino/1.0");
    }
}

```

```

        client.println("Connection: close");
        client.println("Content-Type:          application/x-www-form-
urlencoded;");
        client.print("Content-Length: ");
        client.println(data.length());
        client.println();
        client.println(data);
        Serial.println(" >>Terkirim ");
        data="";
    }
    else {
        Serial.println("Koneksi ke server gagal!");
        client.stop();
    }
}

```

Lampiran 2 : Source code pengujian sensor suhu

```

#include <dht.h>
#define dht_dpın A0

dht DHT;

void setup() {
    Serial.begin(9600);
    DHT.read11(dht_dpın);
}

void loop () {
    Serial.print("Suhu = ");
    Serial.print(DHT.temperature);
    Serial.println(" C");
    delay(1000);
}

```

Lampiran 3 : Source code pengujian sensor SEN0114

```

int sensorPin = A1;
int sensorValue;
void setup() {
    Serial.begin(9600);
}
void loop() {
    sensorValue = analogRead(sensorPin);
    Serial.print("sensor = " );
    Serial.println(sensorValue);
    delay(1000);
}

```

Lampiran 4 : Source code pengujian relay

```
int kipas = 5;
int sprayer= 4;
void setup() {
  pinMode(sprayer, OUTPUT);
  pinMode(kipas, OUTPUT);
}
void loop() {
  digitalWrite(sprayer, HIGH);
  digitalWrite(kipas, HIGH);
  delay(3000);
  digitalWrite(sprayer, LOW);
  digitalWrite(kipas, LOW);
  delay(3000);
}
```

Lampiran 5 : Source code pengujian koneksi

```
#include <WiFi.h>

char ssid[] = "JanuariLulus";
int status = WL_IDLE_STATUS;
float waktu;
WiFiServer server(80);
void setup() {
  Serial.begin(9600);
  while ( status != WL_CONNECTED) {
    Serial.print("Mencoba melakukan koneksi ke SSID: ");
    Serial.println(ssid);
    status = WiFi.begin(ssid);
  }
  waktu=millis()/1000;

  printWiFiStatus();
}
void loop() {
}
void printWiFiStatus() {
  Serial.println("Status WiFi: ");
  Serial.println("Koneksi berhasil ");
  Serial.print("SSID: ");
  Serial.println(WiFi.SSID());
  IPAddress ip = WiFi.localIP();
  Serial.print("IP Address: ");
  Serial.println(ip);
  Serial.print("waktu eksekusi: ");
  Serial.print(waktu);
  Serial.println(" detik");
}
```