

SISTEM KENDALI JEMURAN ANTISIPASI HUJAN



Disusun Oleh :

Ahmad Lufi Alfianul ‘Ula **3.34.16.1.02**

Muhammad Faizul Fikri Ilmansyah 3.34.16.1.18

PROGRAM STUDI TEKNIK INFORMATIKA

JURUSAN TEKNIK ELEKTRO

POLITEKNIK NEGERI SEMARANG

2019

SISTEM KENDALI JEMURAN ANTISIPASI HUJAN



**Tugas akhir ini disusun untuk melengkapi sebagian persyaratan
menjadi Ahli Madya**

Disusun Oleh :

Ahmad Lufi Alfianul ‘Ula **3.34.16.1.02**

Muhammad Faizul Fikri Ilmansyah 3.34.16.1.18

PROGRAM STUDI TEKNIK INFORMATIKA

JURUSAN TEKNIK ELEKTRO

POLITEKNIK NEGERI SEMARANG

2019

PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini, saya :

Nama/ NIM/ Kelas : Ahmad Lufi Alfianul 'Ula / 3.34.16.1.02/ IK-3B

Nama/ NIM/ Kelas : Muhammad Faizul Fikri Ilmansyah / 3.34.16.1.18 / IK-3B

Jurusan : Teknik Elektro

Program Studi : Teknik Informatika

Saya menyatakan dengan sesungguhnya bahwa tugas akhir dengan judul **SISTEM KENDALI JEMURAN ANTISIPASI HUJAN** yang dibuat untuk melengkapi sebagian persyaratan menjadi Ahli Madya pada Program Studi Teknik Informatika Jurusan Teknik Elektro Politeknik Negeri Semarang, sejauh yang saya ketahui bukan merupakan tiruan atau duplikasi dari tugas akhir yang sudah dipublikasikan dan atau pernah dipakai untuk mendapatkan gelar Ahli Madya di lingkungan Politeknik Negeri Semarang maupun di perguruan tinggi atau instansi manapun, kecuali bagian yang sumber informasinya dicantumkan sebagaimana mestinya.

Semarang, 15 Agustus 2019

Mahasiswa,

Mahasiswa,



Ahmad Lufi Alfianul 'Ula
NIM 3.34.16.1.02



Muhammad Faizul Fikri Ilmansyah
NIM 3.34.16.1.18

HALAMAN PERSETUJUAN

Tugas akhir dengan judul “**SISTEM KENDALI JEMURAN ANTISIPASI HUJAN**” dibuat untuk melengkapi sebagian persyaratan menjadi Ahli Madya pada Program Studi Teknik Informatika, Jurusan Teknik Elektro Politeknik Negeri Semarang dan disetujui untuk diajukan dalam sidang ujian tugas akhir.

Pembimbing I,

Semarang, 15 Agustus 2019

Pembimbing II,



Drs. Parsumo Rahardjo, M.Kom.
NIP. 196008221988031001



Wahyu Sulistiyo, S.T., M.Kom.
NIP. 197704012005011001

Mengetahui,
Ketua Program Studi Teknik Informatika



Sukamto, S.Kom., M.T.
NIP. 197101172003121001

HALAMAN PENGESAHAN

Tugas akhir/skripsi dengan judul “**SISTEM KENDALI JEMURAN ANTISIPASI HUJAN**” Telah dipertahankan dalam ujian wawancara dan diterima sebagai syarat untuk menjadi Ahli Madya pada Program Studi Teknik Informatika Jurusan Teknik Elektro Politeknik Negeri Semarang pada tanggal 15 Agustus 2019

Penguji I,



Tri Raharjo Yudantoro, S.Kom, M.Kom.

NIP. 196810252000121001

Tim Penguji

Penguji II,



Slamet Handoko, S.Kom, M.Kom.

NIP. 197501302001121001

Penguji III,



Sukanto, S.Kom., M.T.

NIP. 197101172003121001

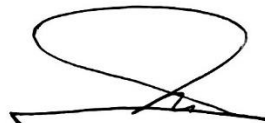
Ketua Penguji,



Drs. Parsumo Rahardjo, M.Kom.

NIP. 196008221988031001

Sekretaris,



Khamami, S.Ag., M.M.

NIP. 197206102000031001

Mengesahkan,

Ketua Jurusan Teknik Elektro

Dr. Anni Subarjono, S.T., M.T.
NIP. 197210271999031002

KATA PENGANTAR

Syukur Alhamdulillah penulis ucapkan kepada Allah SWT yang telah melimpahkan rahmat dan hidayah-Nya kepada penulis sehingga Tugas Akhir (TA) ini dapat terselesaikan dengan baik. TA ini dimaksudkan untuk memenuhi syarat kelulusan Program Diploma III Jurusan Teknik Elektro Program Studi Teknik Informatika Politeknik Negeri Semarang.

Selama menyelesaikan TA ini penulis banyak mendapatkan bantuan dan dukungan dari berbagai pihak. Oleh karena itu penulis ingin menyampaikan ucapan terima kasih kepada :

1. Orang tua dan keluarga yang telah memberikan do'a, motivasi, bantuan, dan dukungan.
2. Bapak Ir. Supriyadi, M.T selaku Direktur Politeknik Negeri Semarang.
3. Bapak Dr. Amin Suharjono, S.T, M.T. selaku Ketua Jurusan Teknik Elektro.
4. Bapak Sukanto, S.Kom., M.T. selaku Ketua Program Studi Teknik Informatika.
5. Bapak Budi Suyanto, S.T., M.Eng. selaku Dosen Wali.
6. Bapak Drs. Parsumo Rahardjo, M.Kom., selaku pembimbing 1.
7. Bapak Wahyu Sulistiyo, S.T., M.Kom., selaku Pembimbing 2.
8. Teman-teman Program Studi Teknik Informatika angkatan tahun 2016 yang telah menempuh pendidikan bersama-sama selama 3 tahun.
9. Seluruh pihak yang telah membantu penulis dalam menyelesaikan TA.

Penulis menyadari bahwa dalam pembuatan TA ini masih jauh dari kata sempurna, oleh karena itu kritik dan saran yang membangun dari seluruh pihak sangat penulis harapkan. Semoga TA ini bermanfaat bagi banyak pihak.

Abstrak

Ahmad Lufi Alfianul 'Ula dan Muhammad Faizul Fikri Ilmansyah,"Sistem Kendali Jemuran Antisipasi", Tugas Akhir DIII Program Studi Teknik Informatika, Jurusan Teknik Elektro Politeknik Negeri Semarang, di bawah bimbingan Drs. Parsumo Rahardjo, M.Kom. dan Wahyu Sulistiyo, S.T., M.Kom., Agustus 2019, 61 halaman.

Permasalahan yang sering terjadi di *laundry* adalah kondisi cuaca sekarang yang tidak bisa diprediksi terkadang hujan terkadang cerah. Apabila terjadi hujan jasa *laundry* memiliki kendala yaitu ketika proses penjemuran terjadi hujan maka pakaian harus segera dimasukkan ke dalam rumah agar tidak basah. Ketika pakaian yang terjemur basah lagi maka akan memakan waktu kembali untuk mengeringkannya dan tentunya sangat merugikan jasa *laundry*. Maka dari itu tujuan pembuatan Tugas Akhir ini adalah untuk membangun Sistem Kendali Jemuran Antisipasi Hujan. Sistem ini dibuat dengan metode *Waterfall* dimana setiap langkah penelitian dilakukan secara berurutan mulai dari *analysis*, *design*, *coding*, *testing*, dan *maintenance*. *Testing* dilakukan dengan metode *black box* yang menguji fungsi masing – masing fitur dan pengujian kepuasan pengguna menggunakan kuisioner. Aplikasi Sistem Kendali Jemuran Antisipasi Hujan sudah diuji di 5 sistem operasi android dan semuanya berjalan dengan lancar.

Kata kunci : Sistem Kendali Jemuran, Smart Jemuran,

DAFTAR ISI

JUDUL	i
SURAT PERNYATAAN.....	ii
HALAMAN PERSETUJUAN	iii
HALAMAN PENGESAHAN.....	iv
KATA PENGANTAR	v
ABSTRAK	vi
DAFTAR ISI	vii
DAFTAR GAMBAR	x
DAFTAR TABEL.....	xii
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.1.1 Rumusan Masalah.....	2
1.1.2 Manfaat	3
1.1.3 Tujuan	3
1.2 Batasan Masalah	3
1.3 Sistematika Penulisan	3
BAB II TINJAUAN PUSTAKA	5
2.1 Penelitian Terkait.....	5
2.2 NodeMCU ESP8266.....	7
2.3 Sensor.....	10
2.4 Sensor Hujan	10
2.5 Sensor Cahaya	11
2.6 Sensor DHT11	12
2.7 Motor Servo.....	14
2.8 IDE Arduino	15
2.9 Bahasa C Arduino.....	17

2.9.1	Struktur	17
2.9.2	Syntax	18
2.9.3	Variabel	18
2.9.4	Operator Matematika.....	19
2.9.5	Operator Pembanding.....	19
2.9.6	Struktur Pengaturan.....	20
2.9.7	Digital	20
2.9.8	Analog	21
2.10	VueJS	21
2.11	Android SDK.....	21
BAB III KEGIATAN PELAKSAAN		22
3.1	Metode Pengembangan Sistem.....	22
3.2	Analisis Kebutuhan	23
3.2.1	Kebutuhan Hardware.....	23
3.2.2	Kebutuhan Software	24
3.3	Perancangan Sistem	24
3.3.1	Block Diagram	24
3.3.2	UML	25
3.3.3	Use Case Diagram.....	25
3.3.4	Activity Diagram.....	26
3.3.4.1	Activity Diagram Memantau Jemuran	26
3.3.4.2	Activity Diagram Kendali Atap Jemuran Otomatis.....	27
3.3.4.3	Activity Diagram Kendali Atap Jemuran Manual	29
3.3.5	Flowchart	29
3.3.6	Wiring Diagram	31
3.4	Antarmuka Aplikasi	32
3.4.1	Tampilan Halaman Home.....	32
3.4.2	Tampilan Kendali Atap	33

3.4.3	Tampilan Halaman About.....	34
3.5	Perancangan Alat	34
3.5.1	Tampilan Bagian Depan	35
3.5.2	Tampilan Bagian Belakang.....	35
3.5.3	Tampilan Bagian Samping Kiri	36
3.5.4	Tampilan Bagian Samping Kanan.....	37
3.6	Penulisan Kode Program.....	37
3.6.1	Penulisan Kode Program Sensor Hujan.....	37
3.6.2	Penulisan Kode Program Sensor Suhu dan Kelembaban DHT11	38
3.6.3	Penulisan Kode Program Sensor Cahaya LDR.....	39
3.6.4	Penulisan Kode Program Motor Servo.....	40
3.7	Perancangan Pengujian Sistem.....	41
3.7.1	Perancangan Uji Alat.....	41
3.6.1.1	Rancangan Uji Hardware Sistem Kendali Jemuran Antisipasi Hujan.....	41
3.6.1.2	Rancangan Uji Aplikasi Sistem Kendali Jemuran Antisipasi Hujan.....	42
3.7.2	Perancangan Pengujian User	42
BAB IV ANALISIS DAN PEMBAHASAN		45
4.1	Analisis Sistem	45
4.1.1	Analisis Alat.....	45
4.1.2	Analisis Aplikasi Android	46
4.2	Pembahasan Sistem.....	47
4.2.1	Implementasi Hardware.....	47
4.2.2	Dimensi Hardware.....	48
4.2.3	Pengujian Fungsionalitas Hardware.....	49
4.2.4	Implementasi Aplikasi.....	58
4.2.5	Pengujian Fungsionalitas Aplikasi	58
4.2.6	Hasil Pengujian Tingkat Kepuasan Pengguna	63

BAB V PENUTUP	65
5.1 Kesimpulan.....	65
5.2 Saran.....	65
DAFTAR PUSTAKA	
LAMPIRAN	

DAFTAR GAMBAR

Gambar 2.1 NodeMCU ESP8266.....	7
Gambar 2.2 GPIO NodeMCU ESP8266 v3	9
Gambar 2.3 Sensor Hujan	11
Gambar 2.4 Sensor Cahaya (LDR)	12
Gambar 2.5 Sensor DHT11	13
Gambar 2.6 Motor Servo SG90.....	15
Gambar 2.7 Tampilan Arduino IDE	16
Gambar 2.8 Toolbar Arduino IDE.....	17
Gambar 3.1 Tahapan Metode Waterfall Menurut Roger S. Pressman	22
Gambar 3.2 Blok Diagram Sistem Kendali Jemuran Antisipasi Hujan.....	25
Gambar 3.3 Use Case Diagram Sistem Kendali Jemuran Antisipasi Hujan.....	26
Gambar 3.4 Activity Diagram Memantau Jemuran.....	27
Gambar 3.5 Activity Diagram Kendali Atap Jemuran Otomatis.....	28
Gambar 3.6 Activity Diagram Kendali Atap Jemuran Manual	29
Gambar 3.7 Diagram Alir Sistem Kendali Jemuran Antisipasi Hujan	30
Gambar 3.8 Wiring Diagram Sistem Kendali Jemuran Antisipasi Hujan	31
Gambar 3.9 Tampilan Halaman Home	32
Gambar 3.10 Tampilan Halaman Kendali Atap Mode Otomatis dan Manual	33
Gambar 3.11 Tampilan Halaman About	34
Gambar 3.12 Tampilan Bagian Depan.....	35
Gambar 3.13 Tampilan Bagian Belakang	36
Gambar 3.14 Tampilan Bagian Kiri	36
Gambar 3.15 Tampilan Bagian Kanan.....	37
Gambar 4.1 Maket Sistem Kendali Jemuran Antisipasi Hujan.....	49
Gambar 4.2 Pengujian Sensor Hujan dengan Tiga Tetesan Kondisi Atap Tertutup .	52
Gambar 4.3 Hasil Pengujian Pertama Sensor LDR Atap Terbuka	54
Gambar 4.4 Pengujian Sensor DHT11 Atap Terbuka	55

Gambar 4.5 Hasil Pengujian Mode Manual dengan Status Kendali Atap adalah Satu	57
Gambar 4.6 Hasil Pengujian Mode Manual Atap Jemuran Tertutup.....	57
Gambar 4.7 Hasil Pengujian Mode Manual dengan Status Kendali Atap adalah Nol58	
Gambar 4.8 Hasil Pengujian Mode Manual Atap Jemuran Terbuka.....	53
Gambar 4.9 Hasil Pengujian Mode Otomatis di Database.....	53
Gambar 4.10 Pengujian Aplikasi Halaman Home Apabila Cuaca Hujan	60
Gambar 4.11 Pengujian Aplikasi Halaman Home Apabila Cuaca Cerah	60
Gambar 4.12 Pengujian Aplikasi Halaman Kendali Tombol Buka Mode Manual ...	61
Gambar 4.13 Pengujian Aplikasi Halaman Kendali Tombol Tutup Mode Manual ..	62
Gambar 4.14 Pengujian Aplikasi Halaman Kendali Mode Otomatis	62
Gambar 4.15 Pengujian Aplikasi Halaman Informasi	63

DAFTAR TABEL

Tabel 2.1 Tabel Analisis Perbandingan Sistem Jemuran Antisipasi Hujan	6
Tabel 2.2 Karakteristik Sensor DHT11.....	14
Tabel 3.1 Daftar Pertanyaan	43
Tabel 3.2 Indikator Penilaian	43
Tabel 3.3 Indikator Kepuasan Pengguna Aplikasi	43
Tabel 4.1 Hasil Pengujian Fungsionalitas Hardware.....	49
Tabel 4.2 Hasil Pengujian Sensor Hujan	51
Tabel 4.3 Hasil Pengujian Sensor LDR	52
Tabel 4.4 Hasil Pengujian Sensor DHT11	54
Tabel 4.5 Implementasi Smart Clothesline	58
Tabel 4.6 Tabel Hasil Pengujian Fungsionalitas Aplikasi	58
Tabel 4.7 Hasil Pengujian Tingkat Kepuasan Pengguna	63

BAB I PENDAHULUAN

1.1 Latar Belakang

Kota Semarang merupakan daerah perkotaan yang mempunyai beragam aktivitas karena sebagai pusat pemerintahan bagi Provinsi Jawa Tengah. Terutama aktivitas ekonomi yang cukup baik membuat penduduk dari daerah pinggiran datang ke Kota Semarang untuk bekerja maupun tinggal. Salah satu contoh kawasan sub urban Semarang yang sedang berkembang adalah kawasan Tembalang. Jarak Tembalang dengan pusat kota cukup jauh, kurang lebih sekitar 15 kilometer. Namun, beberapa tahun terakhir kawasan Tembalang semakin ramai dan padat, bisa dibuktikan dengan melihat banyak bermunculan mini-market baru, menjamurnya kafe dan rumah makan, toko, ruko, kos-kosan, *laundry*, fotokopi, warnet, dan perumahan. Permukiman kawasan Tembalang sukses dalam mengembangkan wilayah dan menarik penduduk dari pusat kota (Prasetya dan Sunaryo, 2013). Hal tersebut juga didukung dengan berdirinya 4 perguruan tinggi yang berdiri di kawasan Kecamatan Tembalang yaitu Universitas Diponegoro, Universitas Pandanaran, Politeknik Negeri Semarang dan Politeknik Kesehatan Semarang (Hapsari, 2013).

Salah satu usaha yang meningkat adalah usaha jasa *laundry*. Jumlah usaha *laundry* yang ada di Tembalang lebih dari 50 outlet (Yuniarti Listia, 2017). Jasa *laundry* merupakan usaha yang menawarkan jasa cuci sebagai berikut : pakaian, sepatu, tas, dan karpet. Dengan adanya jasa *laundry* semakin memudahkan dalam menyelesaikan pekerjaan yang lain, hanya dengan menyerahkan pakaian yang kotor dan menunggu selesai di *laundry*. Biasanya hal tersebut sering dilakukan mahasiswa, karyawan, bahkan ibu rumah tangga yang tidak memiliki waktu untuk mencuci pakaian. Energi mereka sudah digunakan untuk aktivitas mereka yang lebih padat, sehingga lebih memilih menyerahkannya ke jasa *laundry*.

Sementara itu, jasa *laundry* sangat bergantung pada kondisi cuaca. Kondisi cuaca setiap harinya tidak bisa diprediksi terkadang hujan terkadang cerah. Apabila terjadi hujan jasa *laundry* memiliki kendala yaitu ketika proses penjemuran terjadi hujan maka

pakaian harus segera dimasukkan ke dalam rumah agar tidak basah. Ketika pakaian yang terjemur basah lagi maka akan memakan waktu kembali untuk mengeringkannya dan tentunya sangat merugikan jasa *laundry*. Kondisi seperti itu sering terjadi di Berkah *Laundry*, salah satu jasa *laundry* di daerah Bulusan, Tembalang. Selain itu, Berkah *Laundry* masih belum memiliki karyawan, oleh karena itu membutuhkan waktu yang lebih lama untuk memasukkan pakaian ke dalam rumah.

Beberapa penelitian yang telah dilakukan terkait pembuatan sistem kendali jemuranantisipasi hujan telah dilakukan oleh banyak ahli. Rancang bangun jemuran otomatis berbasis web dengan kendali Raspberry Pi yang dibuat oleh Faisal, Aliyadi dan Angga ini mampu mengendalikan jemuran dengan menggunakan Motor DC melalui web browser (Faisal, Aliyadi dan Angga, 2018). Selanjutnya, penelitian rancang bangun prototype alat penjemur pakaian otomatis berbasis arduino uno yang mampu menggerakkan alat jemuran pakaian ke dalam secara otomatis apabila terjadi hujan (Darusman, Dahlan dan Hilyana, 2018). Lalu, Kobandaha, Mosey dan Suoth membuat sistem kontrol atap otomatis tempat penjemuran yang mampu menggerakkan atap jemuran secara otomatis dengan deteksi sensor hujan, sensor dht22, dan sensor cahaya (Kobandaha, Mosey dan Suoth, 2018). Berdasarkan penelitian sebelumnya, fitur yang tersedia terbatas pada kendali atap jemuran secara manual melalui web. Fitur yang belum tersedia meliputi kendali atap jemuran secara otomatis ataupun manual melalui aplikasi android serta pemantauan kondisi sekitar jemuran melalui aplikasi android.

Tugas Akhir (TA) ini menjelaskan tentang pembuatan Sistem Kendali Jemuran Antisipasi Hujan yang memungkinkan pemilik *laundry* dapat mengendalikan jemuran. Fitur yang disediakan meliputi monitoring kondisi jemuran serta kendali atap jemuran secara otomatis ataupun manual melalui android sehingga pemilik *laundry* bisa meninggalkan jemuran tanpa harus menunggu.

1.1.1 Rumusan Masalah

Berdasarkan masalah yang telah diuraikan pada latar belakang, maka dirumuskan bahwa pengusaha *laundry* kesulitan dalam mengendalikan jemuran pada saat musim penghujan. Oleh sebab itu, dirancanglah Sistem Kendali Jemuran Antisipasi Hujan.

1.1.2 Manfaat

Hasil dari Tugas Akhir ini diharapkan dapat memberikan manfaat, diantaranya adalah sebagai berikut :

- a. Membantu pemilik *laundry* dalam mengontrol kondisi jemuran.
- b. Membantu pemilik *laundry* dalam melindungi jemuran pakaian apabila terjadi hujan.

1.1.3 Tujuan

Tujuan dari pembuatan Tugas Akhir ini adalah untuk memecahkan permasalahan para pelaku usaha *laundry* dalam menangani proses penjemuran ketika hujan. Sistem Kendali Jemuran Antisipasi dibangun secara otomatis yang terintegrasi dengan *smartphone* berbasis android yang digunakan sebagai pengendali kondisi jemuran.

1.2 Batasan Masalah

Dalam pembuatan Tugas Akhir ini, masalah yang akan dibahas terbatas pada :

- a. Sistem yang dibangun menggunakan bahasa *C* untuk hardwarenya dan bahasa *VueJS* untuk Androidnya.
- b. Sistem jemuran dibuat hanya untuk *laundry*.
- c. Menggunakan NodeMCU ESP8266 sebagai mikrokontroler.
- d. Menggunakan modul sensor hujan, sensor LDR, dan sensor DHT11.
- e. Menggunakan motor servo SG90 sebagai penggerak atap jemuran.

1.3 Sistematika Penulisan

Sistematika penulisan dari laporan tugas akhir ini adalah :

BAB I : Pendahuluan

Berisi tentang latar belakang, perumusan masalah, batasan masalah, manfaat pembuatan sistem, tujuan pembuatan sistem, dan sistematika penulisan laporan.

BAB II : Tinjauan Pustaka

Membahas materi – materi yang berkaitan dengan permasalahan tugas akhir.

BAB III: Kegiatan Pelaksanaan

Membahas mengenai bahan dan perancangan cara kerja sistem dan metodologi.

BAB IV : Analisis dan Pembahasan

Berisi tentang pembuatan sistem, pengoperasian sistem, pengujian sistem dan hasil pengujian sistem.

BAB V : Penutup

Berisi kesimpulan dan saran dibuatnya laporan TA

DAFTAR PUSTAKA**LAMPIRAN**

BAB II TINJAUAN PUSTAKA

2.1 Studi Sebelumnya

Pada bagian ini dipaparkan beberapa penelitian terkait TA sistem kendali jemuran antisipasi hujan yang telah dibuat oleh peneliti. Selain itu, terdapat perbandingan antara TA sistem kendali jemuran antisipasi hujan dengan sistem lain yang sudah ada. Beberapa penelitian terkait TA sistem kendali jemuran antisipasi hujan diantaranya alat pengendali atap jemuran otomatis dengan sensor cahaya dan sensor air berbasis mikrokontroler ATmega16 yang dibuat oleh Mufidan dan Abas. Sistem ini menggunakan ATmega16 sebagai pusat pengendali jemuran otomatis. Sensor yang diperlukan pada alat pengendali jemuran otomatis adalah sensor cahaya dan sensor air. Sistem ini masih terbatas pada hardware saja, belum adanya teknologi IoT (Mufida dan Abas, 2017).

Candra dan Very membuat sistem jemuran otomatis menggunakan metode logika fuzzy. Mikrokontroler yang digunakan adalah Arduino UNO R3. Sistem ini masih belum menggunakan teknologi IoT hanya memanfaatkan sensor air dan sensor cahaya sebagai inputan data dari lingkungan luar jika terjadi perubahan (Candra dan Very, 2018).

Guk – guk membuat sistem jemuran otomatis menggunakan sensor LDR, sensor hujan dan sensor kelembaban untuk deteksi. Mikrokontroler yang digunakan adalah Arduino UNO R3. Sistem ini masih terbatas hanya pada pengukuran tegangan saja (Guk-guk, 2015).

Faisal, Aliyadi dan Angga membuat sistem jemuran otomatis berbasis mikrokontroler yang dapat diakses melalui web. Mikrokontroler yang digunakan adalah Raspberry PI. Sistem ini masih terbatas belum adanya sensor sebagai deteksi, hanya menggunakan motor DC yang dapat dikendalikan melalui web browser (Faisal, Aliyadi dan Angga, 2018).

Yuwono dan Alam membuat rancang bangun sistem jemuran otomatis berbasis arduino uno. Komponen yang digunakan adalah sensor LDR, sensor hujan FC-37, sensor kelembaban digital dan motor DC. Sistem ini masih terbatas menggunakan LCD untuk menampilkan *monitoringnya*, belum adanya *monitoring* jarak jauh (Yuwono dan Alam, 2018).

Dari beberapa penelitian diatas yang telah dilakukan oleh ahli, belum terdapat sebuah sistem yang mempunyai fitur untuk kendali atap jemuran. Oleh karena itu, TA sistem kendali jemuran antisipasi hujan ini akan mewujudkan sebuah sistem dengan fitur yang telah disebutkan sebelumnya. Selain itu, database yang digunakan adalah MySQL. Kendali atap jemuran serta *monitoring* datanya akan dilakukan pada aplikasi android *Smart Clothesline*.

Tabel 2.1 Tabel Analisis Perbandingan Sistem Jemuran Antisipasi Hujan

No	Sistem	Fitur / Teknologi		
		1	2	3
1	Alat Pengendali Atap Jemuran Otomatis Dengan Sensor Cahaya Dan Sensor Air Berbasis Mikrokontroler ATmega16.	✓	-	-
2	Desain <i>System Smart Clothesline</i> Berbasis Arduino Dengan Metode Logika Fuzzy.	✓	-	-
3	Jemuran Otomatis Dengan Menggunakan Sensor Ldr, Sensor Hujan Dan Sensor Kelembaban.	✓	-	-
4	Rancang Bangun Sistem Jemuran dengan Kendali Raspberry PI (kendali melalui web)	-	✓	-
5	Rancang Bangun Sistem Jemuran Otomatis Berbasis Arduino Uno (tidak memerlukan internet).	✓	-	-
6	Sistem Kendali Jemuran Antisipasi Hujan.	✓	✓	✓

Keterangan :

- 1) Kendali atap otomatis
- 2) Kendali atap manual
- 3) Android

Berdasarkan Tabel 2.1 dapat dilihat bahwa dari beberapa sistem yang dibuat oleh peneliti dan berkaitan dengan TA ini belum ada yang membuat sistem kendali atap manual dan kendali atap otomatis menjadi satu sekaligus. Selain itu, belum ada sistem jemuran yang dibuat menggunakan aplikasi android. Sistem jemuran kendali jemuran antisipasi hujan yang dibuat memiliki fitur standar seperti kendali atap manual dan kendali atap otomatis yang dapat diakses melalui aplikasi android. Kelebihan yang ada dari sistem kendali jemuran antisipasi hujan adalah data yang diterima dan dikirim oleh aplikasi maupun sistem bersifat *real time*.

2.2 NodeMCU ESP8266

NodeMcu merupakan sebuah *opensource platform* IoT dan pengembangan *Kit* yang menggunakan bahasa pemrograman Lua untuk membantu programmer dalam membuat prototype produk IoT atau bisa dengan memakai sketch dengan Arduino IDE. Pengembangan *Kit* ini didasarkan pada modul ESP8266, yang mengintegrasikan GPIO, PWM (Pulse Width Modulation), IIC, 1-Wire dan ADC (Analog to Digital Converter) semua dalam satu board. Dalam pembuatan TA, NodeMCU digunakan sebagai mikrokontroler yang akan menerima perintah dan mengirimkan data ke database.

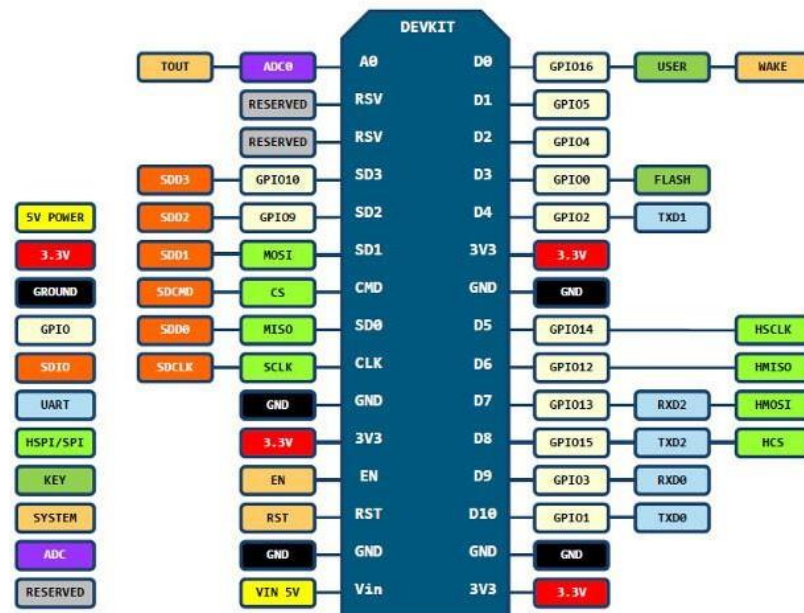


Gambar 2.1 NodeMCU ESP8266

Gambar 2.1 merupakan bentuk fisik dari NodeMCU. Board pada NodeMCU berukuran sangat kecil dengan panjang 4.83 cm dan lebar 2.54 cm, serta dengan berat 7 gram.

Spesifikasi yang dimiliki oleh NodeMCU sebagai berikut :

- a. Board ini berbasis ESP8266 serial WiFi SoC (Single on Chip) dengan onboard USB to TTL. Wireless yang digunakan adalah IEEE 802.11b/g/n.
- b. tantalum capacitor 100 micro farad dan 10 micro farad.
- c. 3.3v LDO regulator.
- d. Blue led sebagai indikator.
- e. CP2102 usb to UART bridge.
- f. Tombol reset, port usb, dan tombol flash.
- g. Terdapat 9 GPIO yang di dalamnya ada 3 pin PWM, 1 x ADC Channel, dan pin RX TX
- h. 3 pin ground.
- i. S3 dan S2 sebagai pin GPIO.
- j. S1 MOSI (Master Output Slave Input) yaitu jalur data dari master dan masuk ke dalam slave, sc cmd/sc.
- k. S0 MISO (Master Input Slave Input) yaitu jalur data keluar dari slave dan masuk ke dalam master.
- l. SK yang merupakan SCLK dari master ke slave yang berfungsi sebagai clock.
- m. Pin Vin sebagai masukan tegangan.
- n. Built in 32-bit MCU.



Gambar 2.2 GPIO NodeMCU ESP8266 v3

Gambar 2.2 merupakan GPIO dari NodeMCU ESP8266 v3. GPIO pada NodeMCU adalah sebagai berikut :

- a. RST : berfungsi mereset modul
- b. ADC: Analog Digital Converter. Rentang tegangan masukan 0-1v, dengan skup nilai digital 0-1024.
- c. EN: Chip Enable, Active High
- d. IO16 :GPIO16, dapat digunakan untuk membangunkan chipset dari mode deep sleep.
- e. IO14 : GPIO14; HSPI_CLK
- f. IO12 : GPIO12: HSPI_MISO
- g. IO13: GPIO13; HSPI_MOSI; UART0_CTS
- h. VCC: Catu daya 3.3V (VDD)
- i. CS0 :Chip selection
- j. MISO : Slave output, Main input
- k. IO9 : GPIO9
- l. IO10 GBIO10
- m. MOSI: Main output slave input

- n. SCLK: Clock
- o. GND: Ground
- p. IO15: GPIO15; MTDO; HSPICS; UART0_RTS
- q. IO2 : GPIO2;UART1_TXD
- r. IO0 : GPIO0
- s. IO4 : GPIO4
- t. IO5 : GPIO5
- u. RXD : UART0_RXD; GPIO3
- v. TXD : UART0_TXD; GPIO1

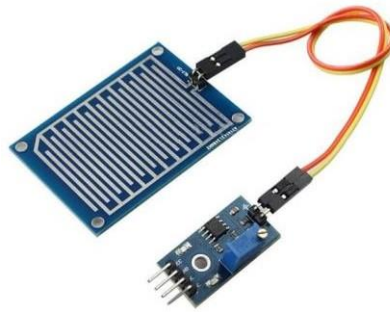
2.3 Sensor

Menurut fraden (2003:64), Sensor berasal dari kata *Sense* (merasakan atau mengindera), adalah mengidefinisikan sensor sebagai Piranti yang menerima sebuah stimulus dan meresponnya dengan sebuah sinyal listrik. Lebih jauh *fraden* mendefinisikan stimulus, atau rangsangan, sebagai kuantitas, sifat atau kondisi tertentu yang dapat dirasakan dan diubah menjadi sinyal listrik. Tujuan dari sebuah sensor adalah merespon sejenis masukan dan mengubah masukan tersebut menjadi sinyal listrik. Keluaran output dari sensor dapat berupa arus atau beda potensial. Setiap sensor pada prinsipnya adalah mengubah energi (energy converter). Sensor adalah jenis transduser yang digunakan untuk mengubah besaran mekanis, magnetis, panas, sinar dan kimia menjadi tegangan dan arus listrik. Sensor sering digunakan untuk pendeteksian pada saat melakukan pengukuran dan pengendalian.

Sensor adalah alat yang digunakan untuk mendeteksi dan mengetahui magnitude tertentu. Sensor merupakan jenis transduser yang digunakan untuk mengubah variasi mekanis, magnetis, panas, sinar dan kimia menjadi tegangan dan arus listrik. Sensor memegang peranan penting dalam mengendalikan proses pabrikasi modern (Petruszella, 2001: 157).

2.4 Sensor Hujan

Sensor hujan adalah sebuah alat yang dapat mendeteksi adanya cuaca hujan. Sensor hujan dalam sistem kendali jemuran antisipasi hujan digunakan untuk mendeteksi cuaca hujan dan tidak hujan. Sensor hujan dapat digunakan sebagai switch, saat adanya tetesan air hujan yang jatuh melewati raining board yang terdapat pada sensor. Selain itu, sensor hujan juga dapat digunakan untuk mengukur intensitas curah hujan (Sulastri, 2016; Katyal et al., 2016; Ünsal et al., 2016). Output analog sensor hujan digunakan untuk melakukan pendeteksian hujan, dengan kondisi nilai output sensor tinggi pada saat tidak mendeteksi hujan, sedangkan pada saat sensor mendeteksi hujan, nilai output sensor rendah. Gambar 2.3, memperlihatkan modul raindrop sensor (Katyal et al., 2016).



Gambar 2.3 Sensor Hujan

2.5 Sensor Cahaya (LDR)

Sensor cahaya adalah alat yang digunakan dalam bidang elektronika yang berfungsi untuk mengubah intensitas cahaya menjadi arus listrik. Sensor cahaya LDR (Light Dependent Resistor) merupakan suatu jenis resistor yang peka terhadap cahaya. Nilai resistansi LDR akan berubah-ubah sesuai dengan intensitas cahaya yang diterima. Jika LDR tidak terkena cahaya maka nilai tahanan akan menjadi besar (sekitar $10\text{M}\Omega$) dan jika terkena cahaya nilai tahanan akan menjadi kecil (sekitar $1\text{k}\Omega$). (Novianty, Lubis, & Tony, 2012 : 1). Sensor cahaya dalam sistem kendali jemuran antisipasi hujan digunakan untuk kondisi cuaca pagi atau malam hari.

Cara kerja dari sensor ini adalah mengubah energi dari foton menjadi elektron, umumnya satu foton dapat membangkitkan satu elektron. Sensor ini mempunyai kegunaan yang sangat luas salah satunya yaitu sebagai pendeteksi cahaya pada jemuran otomatis. Beberapa komponen yang biasanya digunakan dalam rangkaian sensor cahaya adalah LDR (Light Dependent Resistor), Photodiode, dan Photo Transistor.



Gambar 2.4 Sensor Cahaya (LDR)

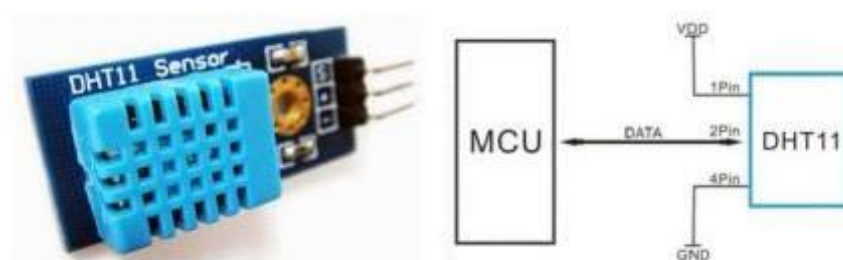
Gambar 2.4 merupakan bentuk fisik dari sensor cahaya LDR. Sensor cahaya ini merupakan suatu resistor yang memiliki nilai hambatan bergantung pada jumlah cahaya yang jatuh pada permukaan sensor tersebut. LDR dapat dibuat dari semikonduktor beresistensi tinggi yang tidak dilindungi dari cahaya. Jika cahaya yang mengenainya memiliki frekuensi yang cukup tinggi, foton yang diserap oleh semikonduktor akan menyebabkan elektron memiliki energi yang cukup untuk meloncat ke pita konduksi. Elektron bebas yang dihasilkan dan pasangan lubangnya akan mengalirkan listrik, sehingga menurunkan resistansinya.

Komponen yang menggunakan sensor cahaya berikutnya adalah Photo Transistor, secara sederhana adalah sebuah transistor bipolar yang memakai kontak (junction) base-collector yang menjadi permukaan agar dapat menerima cahaya sehingga dapat digunakan menjadi konduktivitas transistor. Secara lebih detail Photo Transistor merupakan sebuah benda padat pendeteksi cahaya yang memiliki gain internal. Hal ini yang membuat foto transistor memiliki sensitivitas yang lebih tinggi dibandingkan photodiode / foto diode, dalam ukuran yang sama. Alat ini dapat menghasilkan sinyal analog maupun sinyal digital. Photo Transistor sejenis dengan transistor pada umumnya, bedanya pada Photo Transistor dipasang sebuah lensa pemfokus sinar pada kaki basis untuk memfokuskan sinar jatuh pada pertemuan PN.

2.6 Sensor Suhu dan Kelembaban DHT11

DHT11 adalah sensor Suhu dan Kelembaban (air temperature sensor), memiliki keluaran sinyal digital yang dikalibrasi dengan sensor suhu dan kelembaban yang kompleks. Teknologi ini memastikan keandalan tinggi dan sangat baik stabilitasnya dalam jangka panjang. Mikrokontroler terhubung pada kinerja tinggi sebesar 8 bit. Sensor ini termasuk elemen resistif dan perangkat pengukur suhu NTC. Memiliki kualitas yang sangat baik, respon cepat, kemampuan anti-gangguan dan keuntungan biaya tinggi kinerja (Budiharto:2012). Sensor dht11 dalam sistem kendali jemuranantisipasi hujan digunakan untuk mendeteksi cuaca mendung dengan membaca suhu dan kelembaban sekitar lokasi.

Setiap sensor DHT11 memiliki fitur kalibrasi sangat akurat dari kelembaban ruang kalibrasi. Koefisien kalibrasi yang disimpan dalam memori program OTP, sensor internal mendeteksi sinyal dalam proses, kita harus menyebutnya koefisien kalibrasi. Sistem antarmuka tunggal-kabel serial terintegrasi untuk menjadi cepat dan mudah. Kecil ukuran, daya rendah, sinyal transmisi jarak hingga 20 meter, sehingga berbagai aplikasi dan bahkan aplikasi yang paling menuntut. Produk ini 4-pin pin baris paket tunggal. Koneksi nyaman, paket khusus dapat diberikan sesuai dengan kebutuhan pengguna. Spesifikasi dari sensor DHT11 yaitu Supply Voltage: +5 V, Temperature range : 0-50 °C error of ± 2 °C, Humidity : 20-90% RH $\pm 5\%$ RH error, dengan sesifikasi digital interfacing system. Output yang dihasilkan merupakan baris data digital 40 bits, yang terdiri dari 16 bit data temperature, 16 bit data humidity dan 8 bit data parity. Gambar 2.5 merupakan bentuk fisik dari sensor suhu dan kelembaban DHT11.



Gambar 2.5 Sensor Suhu dan Kelembaban DHT11

Tabel 2.2 Karakteristik Sensor DHT11

Model	DHT11
Power supply	3-5.5V DC
Output signal	digital signal via single-bus
Measuring range	humidity 20-90% RH \pm 5% RH error temperature 0-50 °C error of \pm 2 °C
Accuracy	humidity \pm 4%RH (Max \pm 5%RH); temperature \pm 2.0Celsius
Resolution or sensitivity	humidity 1%RH; temperature 0.1Celsius
Repeatability	humidity \pm 1%RH; temperature \pm 1Celsius
Humidity hysteresis	\pm 1% RH
Long-term Stability	\pm 0.5% RH/year
Sensing period	Average: 2s
Interchangeability	fully interchangeable
Dimensions size	12*15.5*5.5mm

Dari penjelasan Tabel 2.2 bahwa struktur yang merupakan cara kerja dari sensor kelembaban udara/Humidity DHT11 memiliki empat buah kaki yaitu: pada bagian kaki(VCC), dihubungkan ke bagian Vss yang bernilai sebesar 5V,pada board minimum sistem dan untuk bagian kaki GND dihubungkan ke ground (GND) pada minimum sistem, sedangkan pada bagian kaki data yang merupakan keluaran (Output) dari hasil pengolahan data analog dari sensor DHT11 yang dihubungkan ke bagian analog input mikrokontroler dan yang tak ketinggalan terdapat satu kaki tambahan yaitu kaki NC (Not Connected), yang tidak dihubungkan ke pin manapun.

2.7 Motor Servo

Motor servo adalah sebuah motor dengan sistem umpan balik tertutup di mana posisi dari motor akan diinformasikan kembali ke rangkaian kontrol yang ada di dalam motor servo. Motor servo merupakan salah satu jenis motor DC. Motor servo banyak

digunakan pada peranti R/C (Remote Control) seperti mobil, pesawat, helikopter, dan kapal, penggerak pada kamera serta sebagai aktuator robot. (Fahmi, 2011). Gambar 2.6 merupakan motor servo SG90.



Gambar 2.6 Motor Servo SG90

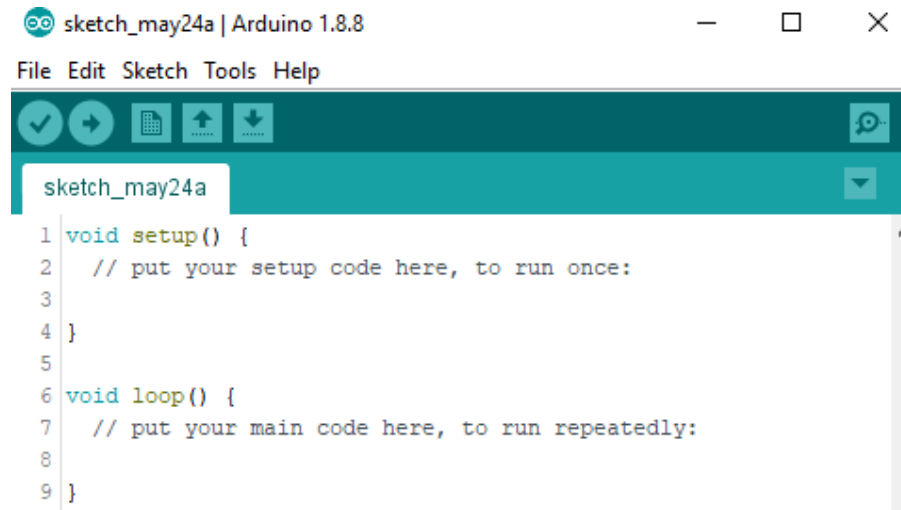
Motor Servo yang digunakan pada sistem kendali jemuran antisipasi hujan adalah motor servo SG90. Motor servo sistem kendali jemuran antisipasi hujan digunakan untuk membuka dan menutupnya atap jemuran secara otomatis maupun manual. Motor servo ini akan bekerja secara baik jika pada bagian pin kontrolnya diberikan sinyal PWM dengan frekuensi 50Hz dengan periode sebesar 20 ms. Pemberian besar pulsa dari mikrokontroler menentukan besar sudut yang harus dilakukan oleh motor servo. Pengaturan sudut motor servo diperlukan untuk mengetahui gerakan dari motor servo dan pulsa yang harus diberikan ke motor servo dalam pergerakan ke kanan atau ke kiri.

2.8 IDE Arduino

IDE Arduino adalah software yang sangat canggih ditulis dengan menggunakan java. IDE Arduino terdiri dari :

- Editor program, sebuah window yang memungkinkan pengguna menulis dan mengedit program dalam bahasa Processing.
- Compiler, sebuah modul yang mengubah kode program (bahasa processing menjadi kode biner. Bagaimanapun sebuah mikrokontroler tidak akan bisa memahami bahasa prosessing. Yang bisa dipahami oleh mikrokontroler adalah kode biner. Itulah sebabnya compiler diperlukan dalam hal ini.

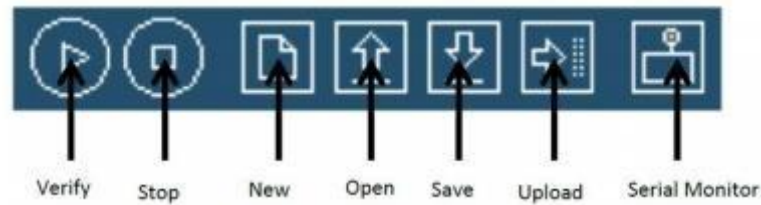
- Uploader, sebuah modul yang memuat kode biner dari computer ke dalam memory dalam papan Arduino.



Gambar 2.7 Tampilan Arduino IDE

Pada Gambar 2.7 merupakan tampilan software Arduino IDE, sedangkan pada Gambar 2.8 dapat melihat toolbar IDE yang memberikan akses instan ke fungsi-fungsi yang penting :

- Dengan tombol Verify, dapat mengkompilasi program yang saat ini di editor.
- Tombol New menciptakan program baru dengan mengosongkan isi dari jendela editor saat ini. Sebelum hal itu terjadi, IDE memberikan kesempatan untuk menyimpan semua perubahan belum disimpan.
- Dengan Open dapat membuka program yang ada dari sistem file.
- Tombol Save menyimpan program saat ini. Ketika mengklik tombol Upload, IDE mengkompilasi saat ini program dan upload ke papan Arduino yang telah pilih di IDE menu Tools > Serial port.
- Arduino dapat berkomunikasi dengan komputer melalui koneksi serial. Mengklik tombol serial monitor membuka jendela serial monitor yang memungkinkan dapat melihat yang dikirimkan oleh Arduino dan juga untuk mengirim data kembali.
- Tombol stop menghentikan serial monitor



Gambar 2.8 Toolbar Arduino IDE

Meskipun menggunakan IDE sangat mudah, mungkin dapat mengalami masalah. Dalam kasus tersebut, dapat lihat menu Help. Menu Help menunjukkan banyak sumber daya yang berguna di website Arduino yang menyediakan solusi cepat tidak hanya untuk semua masalah khas tetapi juga untuk referensi materi dan tutorial.

29 Bahasa C Arduino

Arduino menggunakan pemrograman dengan bahasa C. Berikut ini adalah sedikit penjelasan singkat mengenai karakter bahasa C arduino.

2.9.1 Struktur

Setiap program arduino (biasa disebut *sketch*) mempunyai dua buah fungsi yang harus ada :

1) *Void setup(){}*

Semua kode didalam kurung kurawal akan dijalankan hanya satu kali ketika program arduino dijalankan untuk pertama kalinya.

2) *Void loop (){}*

Fungsi ini akan dijalankan setelah *setup* (fungsi *void setup*) selesai. Setelah dijalankan satu kali fungsi ini akan dijalankan lagi, dan lagi secara terus menerus sampai catu daya (*power*) dilepaskan.

2.9.2 Syntax

Berikut ini adalah elemen bahasa C yang dibutuhkan untuk format penulisan.

1) `//` (Komentar satu baris)

Kadang diperlukan untuk memberikan catatan pada diri sendiri apa arti dari kode-kode yang dituliskan. Cukup menuliskan dua garis miring dan apapun yang diketikkan dibelakangnya akan berwarna hijau dan diabaikan oleh program.

2) `/* */` (komentar banyak baris)

Jika punya banyak catatan, maka hal itu dapat dituliskan pada beberapa baris sebagai komentar. Semua hal yang terlatak diantara dua symbol tersebut akan diabaikan oleh program.

3) `{ }` (kurung kurawal)

Digunakan untuk mendefinisikan kapan blok program mulai berakhir (digunakan juga pada fungsi dan pengulangan).

4) `;` (titik koma)

Setiap baris kode harus diakhiri dengan titik koma (jika titik koma yang hilang maka program tidak bisa dijalankan).

2.9.3 Variabel

Sebuah program secara garis besar dapat didefinisikan sebagai instruksi untuk memindahkan angka dengan cara yang cerdas. *Variable* inilah yang digunakan untuk memindahkannya.

1) *int (integer)*

Digunakan untuk menyimpan angka dalam 2 *byte* (16 bit). Tidak mempunyai angka desimal dan menyimpan nilai dari -32,768 dan 32,767.

2) *long (long)*

Digunakan ketika *integer* tidak mencukupi lagi. Memakai 4 *byte* (32 byte) dari memori (RAM) dan mempunyai rentang dari -2,147,483,683 dan 2,147, 483,647.

3) *Boolean (Boolean)*

Variable sederhana yang digunakan untuk menyimpan nilai *TRUE* atau *FALSE*. Sangat berguna karena hanya menggunakan 1 bit dari RAM.

4) *Float (float)*

Digunakan untuk angka *decimal (floating point)*. Memakai 4 *byte* (32 bit) dari RAM dan mempunyai rentang dari -3.4028235E+38 dan 3.4028235E+38.

5) *Char (character)*

Menyimpan 1 karakter menggunakan kode ASCII (misalnya „A“= 65). Hanya memakai 1 *byte* (8 bit) dari RAM.

2.9.4 Operator Matematika

Operator yang digunakan untuk memanipulasi angka (bekerja seperti matematika yang sederhana).

1) =

membuat sesuatu menjadi sama dengan nilai yang lain (misalnya: $x=10*2$, x sekarang sama dengan 20).

2) %

menghasilkan sisa dari hasil pembagian suatu angka dengan angka yang lain (misalnya : $12\% 10$, ini akan menghasilkan angka 2)

3) + Penjumlahan

4) – Pengurangan

5) * Perkalian

6) / Pembagian

2.9.5 Operator Pembandingan

Digunakan untuk membandingkan nilai logika.

1) ==

Sama dengan (misalnya: $12==10$ adalah *FALSE* atau $12==12$ adalah *TRUE*)

2) !=

Tidak sama dengan (misalnya $12!=10$ adalah *TRUE* atau $12!=12$ adalah *FALSE*).

3) <

Lebih kecil dari (misalnya: $12 < 10$ adalah *FALSE* atau $12 < 12$ adalah *FALSE* atau $12 < 14$ adalah *TRUE*).

4) >

Lebih besar dari (misalnya: $12 > 10$ adalah *TRUE* atau $12 < 12$ adalah *FALSE* atau $12 < 14$ adalah *FALSE*)

2.9.6 Struktur Pengaturan

Program sangat tergantung pada pengaturan apa yang akan dijalankan berikutnya, berikut ini adalah elemen dasar pengaturan.

1) *if..else*, dengan format seperti berikut ini:

```
If (kondisi) { }
Else if (kondisi) { }
Else { }
```

Dengan struktur seperti diatas program akan menjalankan kode yang ada didalam kurung kurawal jika kondisi *TRUE*, dan jika *FALSE* maka akan diperiksa apakah kondisi pada *else if* dan jika kondisinya *FALSE* maka kode pada *else* yang akan dijalankan.

2) *for*, dengan format seperti berikut ini:

```
For (int I = 0; I < #pengulangan; i++) { }
```

Digunakan bila ingin ,melakukan pengulangan kode didalam kurung kurawal beberapa kali, ganti #pengulangan dengan jumlah pengulangan yang diinginkan. Melakukan penghitungan ke atas dengan *i++* atau kebawah *i--*.

2.9.7 Digital

1) *pinMode (pin,mode)*

Digunakan untuk menetapkan mode dari suatu pin, pin adalah nomor pin yang akan digunakan dari 0-9 (pin analog 0-5 adalah 14-19). Mode yang bisa digunakan adalah Input atau output.

2) *DigitalWrite (pin,value)*

Ketika sebuah pin ditetapkan sebagai output, pin tersebut dapat dijadikan *high* (ditarik menjadi 5 volt) atau *low* (diturunkan menjadi *ground*).

2.9.8 Analog

1) *AnalogWrite (pin, value)*

Beberapa pin pada arduino mendukung PWM yaitu 3,5,6,9,10,11. Ini dapat merubah pin hidup (*on*) atau mati (*off*) dengan sangat cepat sehingga membuatnya dapat berfungsi layaknya keluaran analog. Value (nilai) pada format kode tersebut adalah antara 0 (0% *duty cycle* ~ 0V) dan 255 (1000% *duty cycle* ~ 5V)

2) *AnalogRead(pin)*

Ketika pin ditetapkan sebagai input, pengguna dapat membaca tegangan keluarannya. Keluarannya berupa angka antara 0 (untuk 0 volt) dan 1024 (untuk 5 volt)

2.10 VueJS

VueJs merupakan *progressive javascript framework* yang dapat membantu *developer* dalam membuat sebuah website atau aplikasi. VueJs hanya difokuskan untuk membangun tampilan atau hanya bekerja pada view layer. Pada pembuatan TA sistem kendali jemuran antisipasi hujan, VueJS digunakan sebagai media untuk mengedalikan atap jemuran.

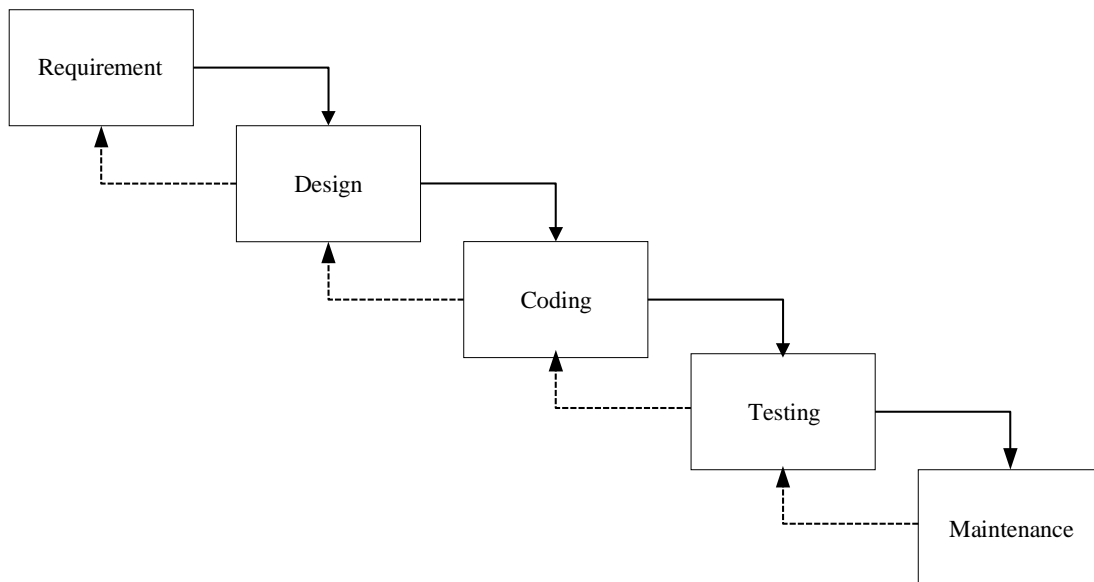
2.11 Android SDK

Android SDK adalah tools API (*Application Programming Interface*) yang diperlukan untuk memulai pengembangan aplikasi pada platform Android menggunakan bahasa pemrograman *Java* (Safaat:2011). Android merupakan subset perangkat lunak untuk ponsel yang meliputi, sistem operasi, middleware dan aplikasi yang dirilis oleh google. Saat ini disediakan android SDK (*Software Development Kit*) sebagai alat bantu dan API untuk mulai mengembangkan aplikasi platform android menggunakan bahasa pemrograman *java*. Sebagai platform aplikasi, android memberi anda kesempatan untuk membuat aplikasi yang kita butuhkan yang buka merupakan aplikasi bawaan *handphone*. Pada pembuata TA sistem kendali jemuran antisipasi hujan Android SDK digunakan *mengcompile* aplikasi android yang telah dibuat dengan VueJS.

BAB III KEGIATAN PELAKSANAAN

3.1 Metode Pengembangan Sistem

Pengembangan Sistem Kendali Jemuran Antisipasi Hujan menggunakan model SDLC (*Software Development Life Cycle*). Model SDLC yang dipakai dalam pengembangan sistem ini yaitu model Waterfall. Metode waterfall menyediakan pendekatan alur hidup perangkat lunak secara urut yang dimulai dari dimulai dengan spesifikasi kebutuhan pengguna lalu berlanjut melalui tahapan-tahapan Analisis yang berguna untuk mendapatkan informasi yang dibutuhkan, Perancangan untuk menentukan perangkat keras dan lunak manakah yang akan digunakan, Pemrograman baik pada *hardware* maupun *software*, dan diakhiri dengan Pengujian untuk menentukan apakah ada kesalahan pada sistem dan mengecek seluruh fungsionalitas sistem (Roger S. Pressman, 2010).



Gambar 3.1 Tahapan Metode Waterfall Menurut Roger S. Pressman

Gambar 3.1 merupakan tahap – tahap metode waterfall. Dalam pengembangannya metode waterfall memiliki beberapa tahapan yang berurut yaitu: *requirement*, *design*, *coding*, dan *testing*. Tahapan tahapan dari metode waterfall adalah sebagai berikut :

1) *Requirement Analysis*

Tahap ini pengembang sistem diperlukan komunikasi yang bertujuan untuk memahami perangkat lunak yang diharapkan oleh pengguna dan batasan perangkat lunak tersebut. Informasi ini biasanya dapat diperoleh melalui wawancara, diskusi atau survei langsung. Informasi dianalisis untuk mendapatkan data yang dibutuhkan oleh pengguna.

2) *Design*

Spesifikasi kebutuhan dari tahap sebelumnya akan dipelajari dalam fase ini dan desain sistem disiapkan. Desain Sistem membantu dalam menentukan perangkat keras (hardware) dan sistem persyaratan dan juga membantu dalam mendefinisikan arsitektur sistem secara keseluruhan.

3) *Coding*

Pemrograman diperlukan dalam tahap pembuatan Sistem Kendali Jemuran Antisipasi Hujan yang berupa pembuatan alat maupun aplikasi dari perancangan sistem sebelumnya.

4) *Testing*

Pada tahap ini, sistem akan diuji fungsionalitasnya untuk mengecek apakah ada kesalahan atau tidak.

5) *Maintenance*

Tahap terakhir yaitu *maintenance*. Tahap ini digunakan untuk pemeliharaan atau pembaharuan terhadap sistem jika ada penambahan fitur –fitur serta mengecek sistem apabila terjadi error pada sistem yang dikembangkan.

3.2 Analisis Kebutuhan

Pada bagian ini akan dibahas mengenai analisis kebutuhan berupa kebutuhan perangkat keras (*hardware*) dan kebutuhan perangkat lunak (*software*) yang diperlukan dalam pembuatan TA sistem kendali jemuran antisipasi hujan.

3.2.1 Kebutuhan *Hardware*

Hardware yang dibutuhkan untuk membangun Sistem Kendali Jemuran Antisipasi Hujan adalah sebagai berikut :

- 1) NodeMCU ESP8266 yang berfungsi sebagai mikrokontroler.
- 2) Sensor LDR yang berfungsi untuk mendeteksi kondisi terang dan gelap.
- 3) Sensor Suhu dan Kelembaban DHT11 yang berfungsi untuk mendeteksi suhu dan kelembaban jemuran.
- 4) Modul Sensor Hujan yang berfungsi untuk mendeteksi air pada saat turun hujan.
- 5) Motor Servo SG90 yang berfungsi untuk menggerakkan atap jemuran.
- 6) Kabel Jumper yang berfungsi untuk menghubungkan antar komponen.
- 7) Breadboard yang berfungsi untuk tempat komponen – komponen.
- 8) Kabel Power yang berfungsi untuk catu daya.

3.2.2 Kebutuhan *Software*

Software yang dibutuhkan untuk membangun Sistem Kendali Jemuran Antisipasi Hujan adalah sebagai berikut :

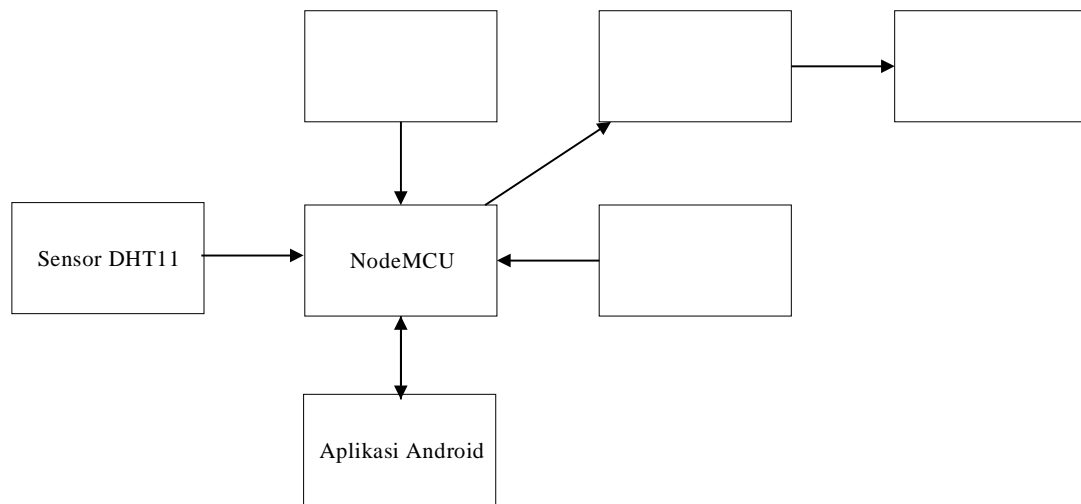
- 1) Sistem Operasi (Windows 10)
- 2) Visual Studio Code
- 3) Android SDK
- 4) Arduino IDE

3.3 Perancangan Sistem

Pada bagian ini membahas perancangan / desain sistem meliputi gambaran umum sistem, antarmuka aplikasi, dan UML yang digunakan dalam TA Sistem Kendali Jemuran Antisipasi Hujan.

3.3.1 Block Diagram

Block Diagram digunakan untuk mengetahui bagaimana Sistem Kendali Jemuran Antisipasi Hujan dalam bentuk blok-blok diagram seperti pada Gambar 3.2.



Gambar 3.2 Blok Diagram Sistem Kendali Jemuran Antisipasi Hujan

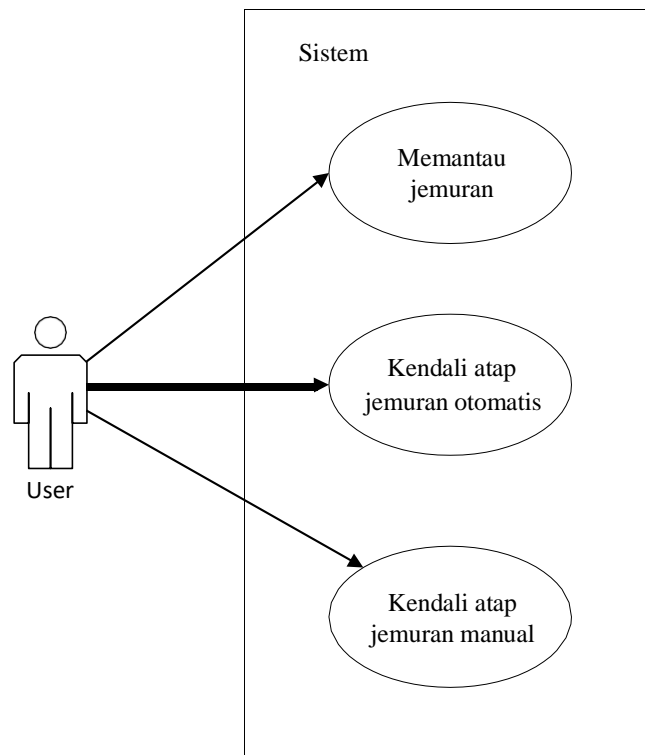
Pada Gambar 3.2 dapat diketahui bahwa NodeMCU menerima data dari beberapa sensor seperti sensor ldr, sensor dht11, dan sensor hujan. Kemudian data yang diterima oleh NodeMCU dikirim ke aplikasi android. Selain itu, NodeMCU mengontrol motor servo untuk menggerakkan atap jemuran.

3.3.2 UML

Model UML (*Unified Modelling Language*) yang digunakan dalam pembuatan Sistem Kendali Jemuran Antisipasi Hujan adalah *Use Case Diagram* dan *Activity Diagram*.

3.3.3 Use Case Diagram

Use Case Diagram merupakan sebuah teknik yang digunakan dalam pengembangan sebuah *software* untuk menjelaskan interaksi yang terjadi antara aktor—inisiator dari interaksi sistem itu sendiri dengan sistem yang ada. *Use Case Diagram* Sistem Kendali Jemuran Antisipasi Hujan terdapat pada Gambar 3.3.



Gambar 3.3 Use Case Diagram Sistem Kendali Jemuran Antisipasi Hujan

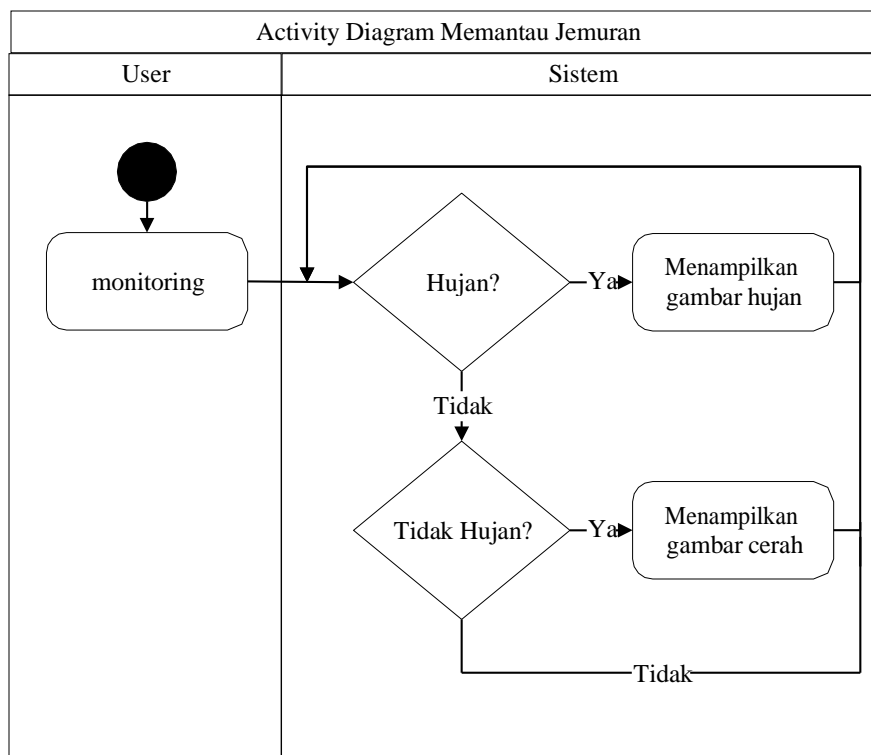
Berdasarkan Gambar 3.3 user dapat melakukan akses aplikasi untuk memantau jemuran serta mengendalikan atap jemuran secara otomatis dan mengendalikan atap jemuran secara manual.

3.3.4 Activity Diagram

Activity diagram menggambarkan logika prosedural, proses bisnis, jalur kerja, dan urutan aktivitas dalam suatu proses.

3.3.4.1 Activity Diagram Memantau Jemuran

Activity Diagram Memantau Jemuran ditampilkan pada Gambar 3.4.

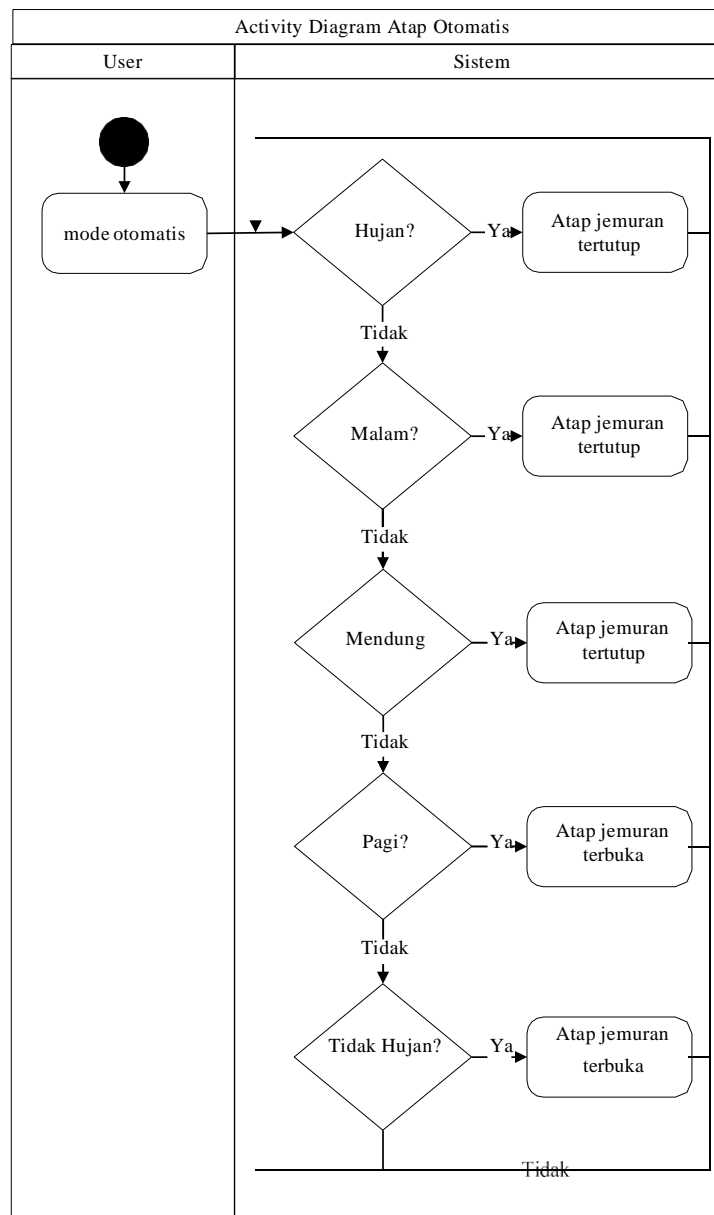


Gambar 3.4 Activity Diagram Memantau Jemuran

Berdasarkan Gambar 3.4 apabila sensor hujan mendeteksi adanya cuaca hujan, maka akan menampilkan gambar hujan. Apabila sensor hujan mendeteksi adanya cuaca tidak hujan, maka akan menampilkan gambar cerah.

3.3.4.2 Activity Diagram Kendali Atap Jemuran Otomatis

Activity Diagram Kendali Atap Jemuran Otomatis ditampilkan pada Gambar 3.5.



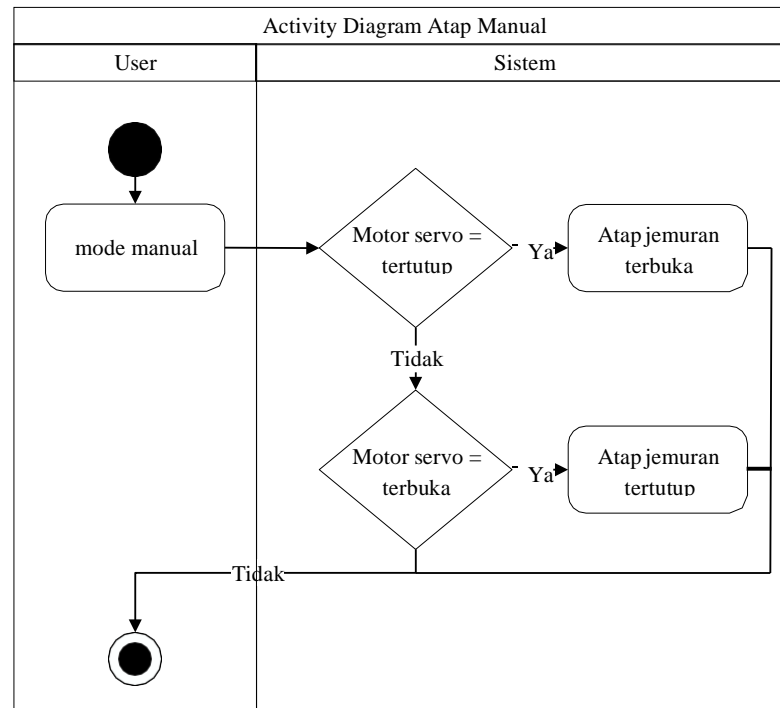
Gambar 3.5 Activity Diagram Kendali Atap Jemuran Otomatis

Berdasarkan Gambar 3.5 apabila mode otomatis, maka mengandalkan sensor hujan, sensor ldr, dan sensor dht11. Apabila sensor hujan mendeteksi adanya hujan, maka atap jemuran menjadi tertutup. Apabila sensor ldr mendeteksi cuaca malam hari, maka atap jemuran akan tertutup. Apabila sensor suhu dan kelembaban mendeteksi adanya mendung, maka atap jemuran akan tertutup. Apabila sensor hujan tidak mendeteksi

adanya hujan, maka atap jemuran menjadi terbuka. Apabila sensor ldr mendeteksi cuaca cerah, maka atap jemuran akan tertutup.

3.3.4.3 Activity Diagram Kendali Atap Jemuran Manual

Activity Diagram Kendali Atap Jemuran Manual ditampilkan pada Gambar 3.6

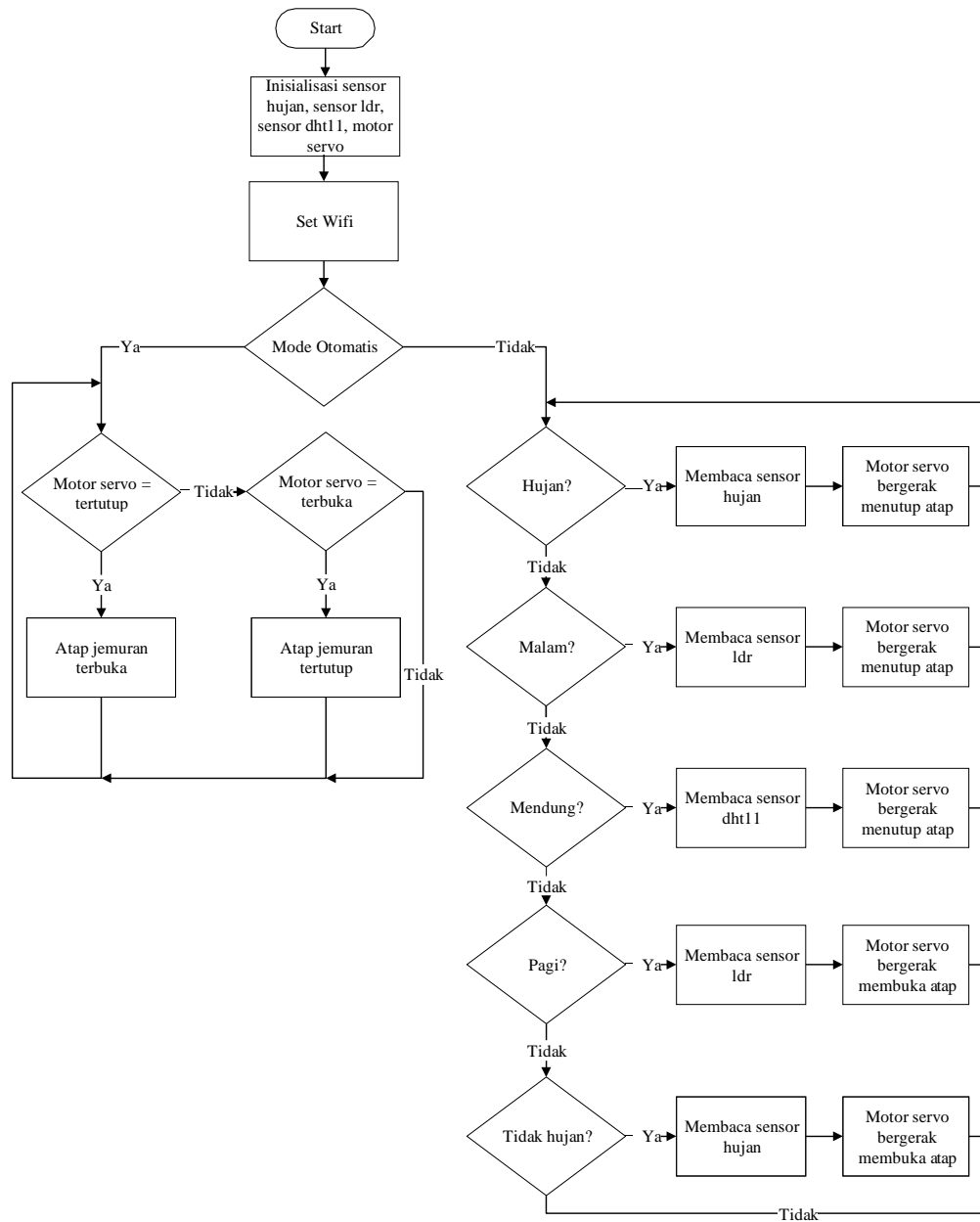


Gambar 3.6 Activity Diagram Kendali Atap Jemuran Manual

Berdasarkan Gambar 3.6 apabila mode manual, maka mengandalkan motor servo untuk membuka dan menutup atap jemuran. Apabila motor servo tertutup, maka atap jemuran menjadi terbuka. Sebaliknya, apabila motor servo terbuka, maka atap jemuran menjadi tertutup.

3.3.5 Flowchart

Flowchart menggambarkan prosedur atau rangkaian proses pada Sistem Kendali Jemuran Antisipasi Hujan. Berikut ini adalah diagram alir Sistem Kendali Jemuran Antisipasi Hujan yang ditampilkan pada Gambar 3.7.



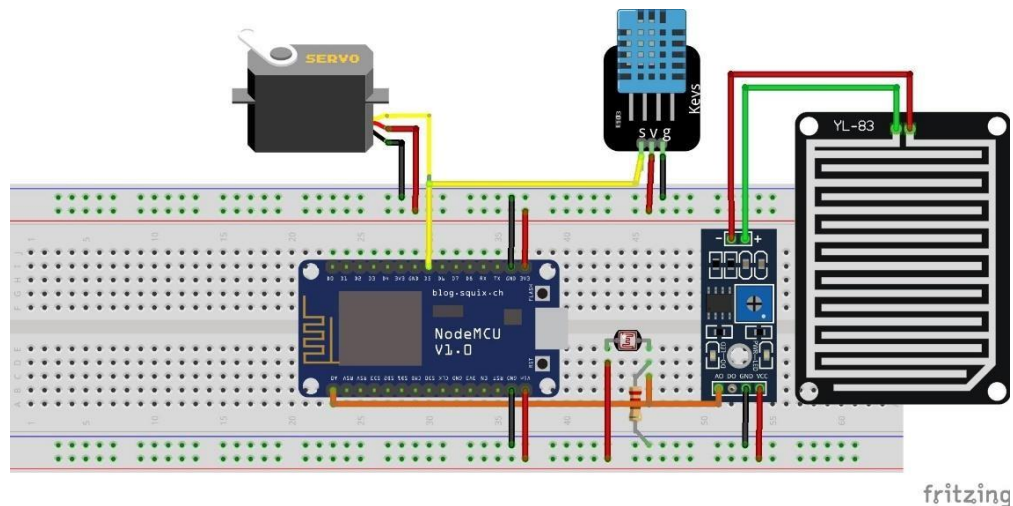
Gambar 3.7 Diagram Alir Sistem Kendali Jemuran Antisipasi Hujan

Pada Gambar 3.7 menunjukkan diagram alir sistem kendali jemuran antisipasi hujan. Apabila mode otomatis aktif, maka sistem jemuran mengandalkan kendali motor servo. Apabila motor servo tertutup, maka atap jemuran menjadi terbuka. Sebaliknya, apabila motor servo terbuka, maka atap jemuran menjadi tertutup. Apabila mode otomatis tidak

aktif, maka sistem kendali jemuran antisipasi hujan mengandalkan sensor hujan fc-37, sensor cahaya ldr, dan sensor suhu dan kelembaban dht11 untuk mendeteksi cuaca sekitar. Ketika sensor hujan fc-37 mendeteksi adanya cuaca hujan, maka akan memberi perintah pada nodemcu untuk menghidupkan motor servo yang menggerakkan atap jemuran menjadi tertutup. Ketika sensor cahaya ldr mendeteksi adanya cuaca malam hari, maka akan memberi perintah pada nodemcu untuk menghidupkan motor servo yang menggerakkan atap jemuran menjadi tertutup. Ketika sensor suhu dan kelembaban dht11 mendeteksi adanya cuaca mendung, maka akan memberi perintah pada nodemcu untuk menghidupkan motor servo yang menggerakkan atap jemuran menjadi tertutup. Ketika sensor cahaya ldr mendeteksi adanya cuaca pagi hari, maka akan memberi perintah pada nodemcu untuk menghidupkan motor servo yang menggerakkan atap jemuran menjadi terbuka. Ketika sensor hujan fc-37 mendeteksi adanya cuaca tidak hujan, maka akan memberi perintah pada nodemcu untuk menghidupkan motor servo yang menggerakkan atap jemuran menjadi terbuka.

3.3.6 Wiring Diagram

Wiring diagram sistem kendali jemuran antisipasi hujan akan ditampilkan pada Gambar 3.8.



Gambar 3.8 Wiring Diagram Sistem Kendali Jemuran Antisipasi Hujan

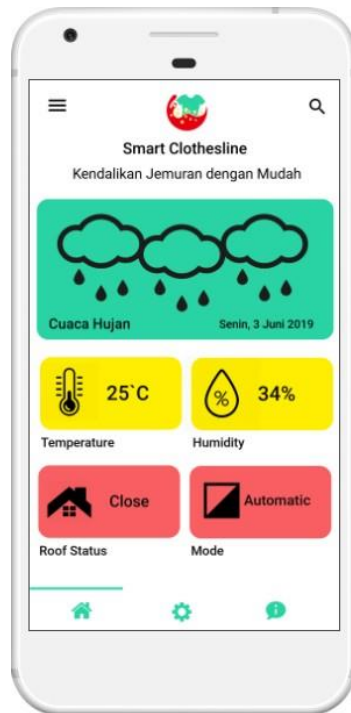
Pada Gambar 3.8 terdapat sebuah motor servo yang dihubungkan ke NodeMCU. Motor servo akan dikontrol pergerakannya melalui aplikasi. Aplikasi akan mengirimkan perintah yang akan dibaca oleh NodeMCU, kemudian NodeMCU akan mengatur pergerakan motor servo sesuai dengan perintah. Selain itu terdapat sensor hujan fc-37, sensor ldr, dan sensor dht11 yang terhubung ke NodeMCU. Ketiga sensor tersebut digunakan untuk mendeteksi pergerakan motor servo secara otomatis.

3.4 Antarmuka Aplikasi

Pada bagian ini akan dipaparkan tentang tampilan GUI (*Graphical User Interface*) dari sistem kendali jemuran antisipasi hujan.

3.4.1 Tampilan Halaman *Home*

Tampilan home adalah tampilan awal saat aplikasi pertama kali dibuka seperti pada Gambar 3.9.

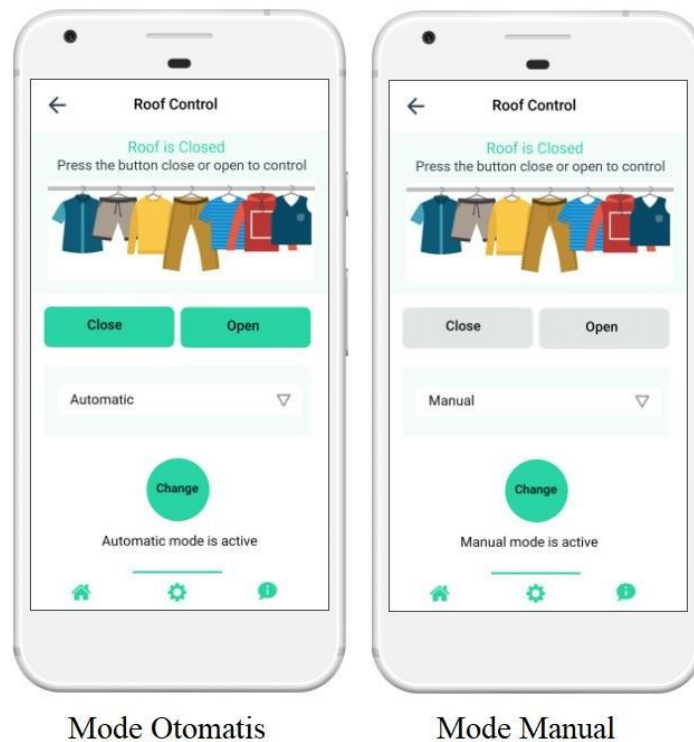


Gambar 3.9 Tampilan Halaman Home

Halaman seperti Gambar 3.9 akan tampil pertama kali saat aplikasi dijalankan. Pada halaman awal terdapat tiga tab menu yaitu menu home, menu kendali, dan menu informasi aplikasi. User dapat melihat kondisi jemuran apakah sedang terjadi hujan, mendung, ataupun panas. Lalu, halaman ini menampilkan kondisi nilai dari sensor suhu dan kelembaban. Kemudian terdapat status atap jemuran yang sedang tertutup atau terbuka. Terdapat juga informasi mode yang digunakan untuk mengendalikan jemuran secara otomatis ataupun manual.

3.4.2 Tampilan Halaman Kendali Atap

Tampilan halaman kendali atap terdapat pada tab menu kendali. Tampilan halaman kendali atap akan tampak seperti pada Gambar 3.10.



Gambar 3.10 Tampilan Halaman Kendali Atap Mode Otomatis dan Manual

Pada Gambar 3.10 user dapat mengendalikan atap jemuran secara otomatis ataupun manual. Apabila ingin mengendalikan atap jemuran secara otomatis, maka *user* harus memilih mode otomatis kemudian menekan tombol *change*. Apabila ingin

mengendalikan atap jemuran secara manual, maka *user* harus memilih mode manual, kemudian menekan tombol *change*. Apabila sudah berada dalam mode manual, untuk mengendalikan atap jemuran supaya tertutup, maka harus menekan tombol *close*. Untuk mengendalikan atap jemuran supaya terbuka, maka harus menekan tombol *open*.

3.4.3 Tampilan Halaman *About*

Tampilan halaman *about* terdapat pada tab menu informasi. Tampilan halaman *about* akan tampak seperti pada Gambar 3.11.



Gambar 3.11 Tampilan Halaman *About*

Pada Gambar 3.11 terdapat informasi aplikasi. Menampilkan nama aplikasi beserta deskripsi dari aplikasi. Informasi aplikasi ada pada tab menu informasi.

3.5 Perancangan Alat

Pada bagian ini akan ditampilkan rancangan sistem kendali jemuran antisipasi hujan yang meliputi bagian depan, samping kanan, samping kiri dan belakang. Pada rancangan sistem ini, akan ditampilkan perangkat-perangkat yang digunakan seperti

sensor hujan, sensor cahaya ldr, dan kotak panel yang berisi NodeMCU dan perangkat lainnya.

3.5.1 Tampilan Bagian Depan

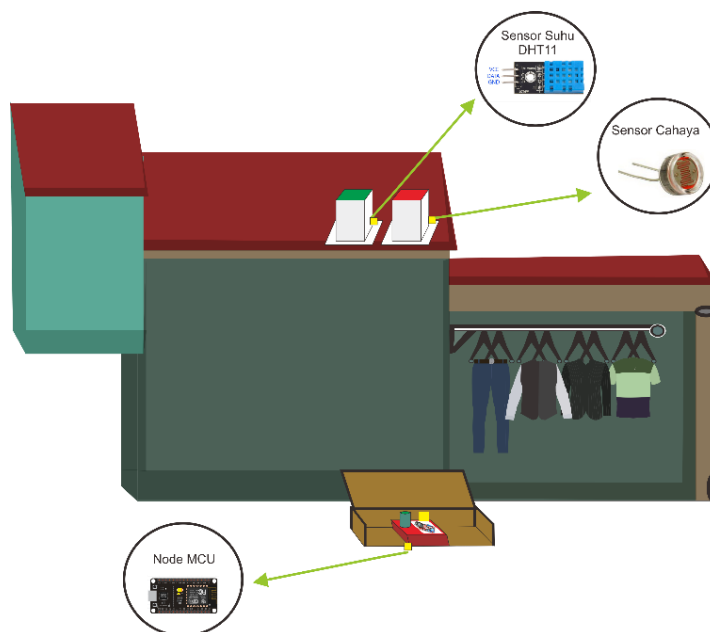
Tampilan bagian depan dari rancangan sistem kendali jemuran antisipasi hujan akan ditampilkan pada Gambar 3.12.



Gambar 3.12 Tampilan Bagian Depan

3.5.2 Tampilan Bagian Belakang

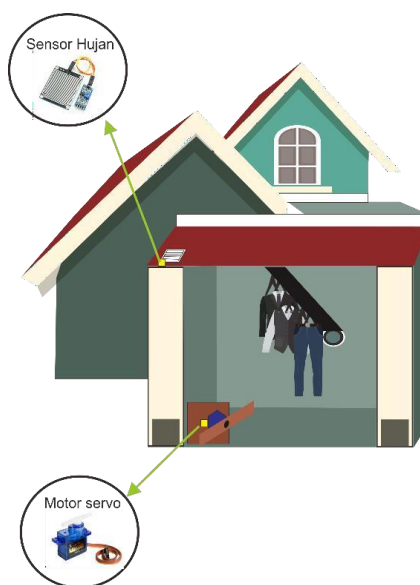
Tampilan bagian belakang dari rancangan sistem kendali jemuran antisipasi hujan akan ditampilkan pada Gambar 3.13.



Gambar 3.13 Tampilan Bagian Belakang

3.5.3 Tampilan Bagian Samping Kiri

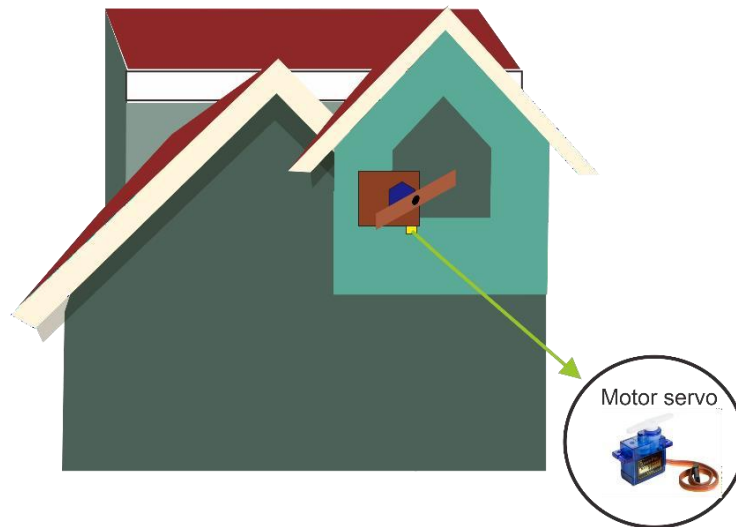
Tampilan bagian samping kiri dari rancangan sistem kendali jemuran antisipasi hujan akan ditampilkan pada Gambar 3.14.



Gambar 3.14 Tampilan Bagian Kiri

3.5.4 Tampilan Bagian Samping Kanan

Tampilan bagian samping kanan dari rancangan sistem kendali jemuran antisipasi hujan akan ditampilkan pada Gambar 3.15.



Gambar 3.15 Tampilan Bagian Kanan

3.6 Penulisan Kode Program

Dalam tahap penulisan kode program, Sistem Kendali Jemuran Antisipasi Hujan menggunakan *software* Arduino IDE dan menggunakan bahasa pemrograman C.

3.6.1 Penulisan Kode Program Sensor Hujan

Penulisan kode program pada sensor hujan bertujuan untuk mendeteksi cuaca hujan dan tidak hujan. Berikut kode program sensor hujan :

```
int HujanDigital = D6;

void setup () {
  Serial.begin(115200);
}
```

```

void loop() {
  int hujan = digitalRead(HujanDigital);
  if(hujan == 1) {
    Serial.println("Tidak Hujan");
  } else {
    Serial.println("Hujan");
  }
}

```

Kode program sensor hujan akan membaca data dari sensor hujan. Apabila sensor hujan membaca data yang nilainya satu, maka layar monitor akan menampilkan cuaca tidak hujan. Apabila sensor hujan membaca data yang nilainya nol, maka layar monitor akan menampilkan cuaca hujan.

3.6.2 Penulisan Kode Program Sensor Suhu dan Kelembaban DHT11

Penulisan kode program pada sensor sensor suhu dan kelembaban DHT11 bertujuan untuk mendeteksi cuaca mendung. Berikut kode program sensor suhu dan kelembaban DHT11 :

```

#include <DHT.h>
DHT dht(2, DHT11);

void setup(){
  Serial.begin(115200);
  dht.begin();
}

void loop(){
  float kelembaban = dht.readHumidity();
  float suhu = dht.readTemperature();
  Serial.print("kelembaban: ");
  Serial.print(kelembaban);
  Serial.print(" ");
  Serial.print("suhu: ");
  Serial.println(suhu);
  if (suhu <= 25) {

```

```

    Serial.print("Mendung");
  }
}

```

Kode program sensor suhu dan kelembaban DHT11 akan membaca data dari sensor suhu dan kelembaban DHT11. Apabila sensor suhu dan kelembaban DHT11 membaca data suhu yang nilainya kurang dari 25 derajat celcius, maka layar monitor akan menampilkan cuaca mendung.

3.6.3 Penulisan Kode Program Sensor Cahaya LDR

Penulisan kode program pada sensor cahaya LDR bertujuan untuk mendeteksi cuaca cerah (siang hari) dan gelap (malam hari). Berikut kode program sensor cahaya LDR :

```

byte ldr= A0;
int nilai;

void setup(){
  Serial.begin(115200);
}

void loop(){
  nilai= analogRead(ldr);
  Serial.print("Nilai LDR: ");
  Serial.println(nilai);
  if(nilai < 500){
    Serial.print("Gelap");
  }
  else{
    Serial.print("Cerah");
  }
}

```

Kode program sensor cahaya LDR akan membaca data dari cahaya LDR. Apabila sensor cahaya LDR membaca data yang nilainya kurang dari 500 lux, maka layar

monitor akan menampilkan gelap. Apabila sensor cahaya LDR membaca data yang nilainya lebih dari 500 lux, maka layar monitor akan menampilkan cerah.

3.6.4 Penulisan Kode Program Motor Servo

Penulisan kode program pada motor servo bertujuan untuk menggerakkan atap jemuran menjadi terbuka dan tertutup sesuai dengan perintah pada program. Berikut kode program motor servo:

```
#include <Servo.h>
Servo motorServo;

void setup()
{
    motorServo.attach(10);
}

void loop()
{
    motorServo.write(0);
    delay(1000);
    motorServo.write(90);
    delay(1000);
    motorServo.write(135);
    delay(1000);
    motorServo.write(180);
    delay(1000);
}
```

Kode program motor servo akan membaca nilai sudut putaran sesuai perintah pada program. Posisi motor servo dimulai dari 0 derajat. Kemudian setelah delay 1 detik motor servo akan bergerak ke sudut 90 derajat. Kemudian setelah delay 1 detik motor servo akan bergerak ke sudut 135 derajat. Kemudian setelah delay 1 detik motor servo akan bergerak ke sudut 180 derajat

3.7 Perancangan Pengujian Sistem

3.7.1 Perancangan Uji Alat

3.7.1.1 Rancangan Uji *Hardware* Sistem Kendali Jemuran Antisipasi Hujan

- 1) Sensor Hujan FC-37
 - a. Tujuan : sebagai pendeteksi hujan di sekitar alat sehingga atap jemuran dapat membuka dan menutup secara otomatis.
 - b. Cara menguji : hubungkan sensor hujan dengan NodeMCU dan *upload* kode program yang berisi kondisi apabila sensor sesuai dengan perintah yang ditetapkan, maka motor servo akan bergerak membuka atau menutup atap jemuran.
- 2) Sensor Cahaya LDR
 - a. Tujuan : sebagai pendeteksi cuaca terang dan gelap di sekitar alat sehingga atap jemuran dapat membuka dan menutup secara otomatis.
 - b. Cara menguji : hubungkan sensor cahaya dengan NodeMCU dan *upload* kode program yang berisi kondisi apabila sensor sesuai dengan perintah yang ditetapkan, maka motor servo akan bergerak membuka atau menutup atap jemuran.
- 3) Sensor Suhu dan Kelembaban DHT11
 - a. Tujuan : sebagai pendeteksi suhu dan kelembaban di sekitar alat sehingga atap jemuran dapat membuka dan menutup secara otomatis.
 - b. Cara menguji : hubungkan sensor suhu dan kelembaban dengan NodeMCU dan *upload* kode program yang berisi kondisi apabila sensor sesuai dengan perintah yang ditetapkan, maka motor servo akan bergerak membuka atau menutup atap jemuran.
- 4) Motor Servo SG90
 - a. Tujuan : sebagai penggerak atap jemuran yang dapat dikendalikan untuk membuka atau menutup melalui sensor atau android.
 - b. Cara menguji : hubungkan motor servo dengan vcc, ground, dan pin *output*

pada NodeMCU. Lalu *upload* kode program yang berisi kondisi yang memungkinkan bagi motor servo untuk membuka atau menutup.

3.7.1.2 Rancangan Uji Aplikasi Sistem Kendali Jemuran Antisipasi Hujan

- 1) Pengujian halaman home
 - a. Tujuan : mengetahui informasi jemuran sesuai dengan kondisi sensor.
 - b. Cara menguji : membuka tab menu *home*.
- 2) Pengujian halaman kendali atap
 - a. Tujuan : mengontrol atap jemuran dengan mode manual atau otomatis.
 - b. Cara menguji : memilih pilihan mode manual atau otomatis pada tab menu kendali atap di aplikasi. Apabila memilih mode manual maka tombol buka dan tutup aktif. Tekan tombol buka untuk membuka atap jemuran. Tekan tombol tutup untuk menutup atap jemuran. Apabila memilih mode otomatis, maka tombol buka dan tutup tidak aktif.
- 3) Pengujian halaman informasi
 - a. Tujuan : mengetahui informasi dari deskripsi aplikasi
 - b. Cara menguji : membuka tab menu informasi.

3.7.2 Perancangan Pengujian User

Pada tahapan ini dilakukan proses pengujian sistem yang sudah jadi kepada user. Dalam hal ini penulis menggunakan sampel 10 orang pemilik *laundry* untuk melakukan pengujian sistem. Dalam pengujian sistem ini didapatkan data dengan cara memberikan beberapa poin pertanyaan seputar sistem ini serta mengharapkan kritik dan saran demi perkembangan selanjutnya.

Dalam pengujian alat Sistem Kendali Jemuran Antisipasi Hujan ini diberikan beberapa pertanyaan untuk mendapatkan hasil perhitungan yang digunakan sebagai acuan untuk mengetahui tingkat kepuasan pengguna dalam penggunaannya. Untuk daftar

pertanyaan dapat dilihat pada Tabel 3.1, sedangkan untuk penilaian pengukuran tingkat kepuasan pengguna didasarkan pada Tabel 3.2 dan Tabel 3.3.

Tabel 3.1 Daftar Pertanyaan

No	Kriteria Pertanyaan
1	Setujukah kemanfaatan fungsi Sistem Kendali Jemuran Antisipasi Hujan?
2	Setujukah kemanfaatan fungsi aplikasi Smart Clothesline sebagai antarmuka Sistem Kendali Jemuran Antisipasi Hujan?
3	Setujukah kemudahan dalam menggunakan aplikasi Smart Clothesline?
4	Setujukah keberhasilan kontrol atap jemuran dalam Sistem Kendali Jemuran Antisipasi Hujan baik secara manual maupun otomatis?
5	Setujukah keakuratan informasi yang ada di aplikasi Smart Clothesline?

Tabel 3.2 Indikator Penilaian

Predikat	Nilai
Sangat setuju	5
Setuju	4
Cukup setuju	3
Kurang setuju	2
Tidak setuju	1

Tabel 3.3 Indikator Kepuasan Pengguna Aplikasi

Kategori	Presentase
Sangat Memuaskan	81% - 100%
Memuaskan	61% - 80%
Cukup Memuaskan	41% - 60%
Tidak Memuaskan	21% - 40%
Sangat Tidak Memuaskan	0-20 %

Untuk menghitung tingkat kepuasan pengguna menggunakan rumus sebagai berikut:

$$\text{Presentase kepuasan pengguna (\%)} = \frac{\text{Total kepuasan pengguna}}{\text{Total kepuasan maksimum pengguna}} \times 100\%$$

BAB IV ANALISIS DAN PEMBAHASAN

4.1 Analisis Sistem

Pada bagian ini dibahas mengenai analisis sistem kendali jemuran antisipasi hujan sebagai sistem otomasi atap menggunakan aplikasi android.

4.1.1 Analisis Alat

Sistem Kendali Jemuran Antisipasi Hujan dibuat dengan komponen utama yaitu NodeMCU sebagai mikrokontroller, sensor hujan FC-37, sensor cahaya LDR, sensor suhu dan kelembaban DHT11, dan motor servo SG90. Sensor hujan FC-37 digunakan untuk mendeteksi cuaca hujan. Sensor cahaya LDR digunakan untuk mendeteksi kondisi cuaca pagi dan malam hari. Sedangkan sensor suhu dan kelembaban DHT11 digunakan untuk mendeteksi cuaca mendung. Ketiga sensor tersebut akan memberi perintah pada nodemcu untuk menggerakkan motor servo sesuai dengan perintah pada program. Hasil pembacaan dari ketiga sensor tersebut akan dikirimkan ke aplikasi android sebagai informasi kepada pemilik laundry.

Motor servo SG90 digunakan untuk membuka dan menutupnya atap jemuran. Motor servo memiliki dua mode dalam pengaplikasiannya. Mode pertama adalah mode manual. Mode manual akan aktif ketika pemilik laundry tidak mengaktifkan mode otomatis di menu kendali pada aplikasi android. Sehingga tombol kendali atap pada aplikasi android menjadi aktif dan dapat digunakan untuk mengendalikan atap jemuran. Pada tombol kendali atap terdapat dua tombol yaitu tombol buka dan tutup. Apabila ditekan tombol buka, maka aplikasi android akan mengirimkan status kendali atap yang nilainya satu ke database kemudian atap jemuran akan terbuka. Sebaliknya, jika ditekan tombol tutup, maka aplikasi android akan mengirimkan status kendali atap yang nilainya nol ke database kemudian atap jemuran akan tertutup.

Pada mode mode otomatis, pemilik laundry harus memilih mode otomatis pada menu kendali hingga terdapat mode otomatis telah aktif di aplikasi android. Pada mode ini sensor hujan, sensor cahaya LDR, sensor suhu dan kelembaban DHT11 akan aktif.

Sensor hujan akan membaca apakah kondisi cuaca sedang hujan atau tidak hujan. Apabila kondisi hujan, maka sensor hujan bernilai nol. Nilai nol akan dikirim NodeMCU ke database. Kemudian sensor hujan akan mengirim perintah ke NodeMCU untuk menghidupkan motor servo yang menggerakkan atap jemuran menjadi tertutup. Sebaliknya, jika sensor hujan tidak mendeteksi adanya hujan, maka NodeMCU akan mengirimkan nilai satu pada database sehingga NodeMCU akan mengubah pergerakan motor servo menjadi 180 derajat sehingga atap jemuran menjadi terbuka.

Sensor cahaya LDR akan membaca apakah kondisi cuaca pagi atau malam hari. Apabila kondisi pagi, maka sensor cahaya bernilai lebih dari 500. Nilai lebih dari 500 akan dikirim NodeMCU ke database. Kemudian sensor cahaya LDR akan mengirim perintah ke NodeMCU untuk menghidupkan motor servo yang menggerakkan atap jemuran menjadi terbuka. Sebaliknya, jika sensor cahaya LDR kondisi cuaca malam hari, maka NodeMCU akan mengirimkan nilai kurang dari 500 pada database sehingga NodeMCU akan mengubah pergerakan motor servo menjadi 180 derajat sehingga atap jemuran menjadi tertutup.

Sensor suhu DHT11 akan membaca apakah kondisi cuaca mendung. Apabila kondisi cuaca mendung, maka sensor suhu DHT11 bernilai kurang dari 25 derajat. Nilai kurang dari 25 derajat akan dikirim NodeMCU ke database. Kemudian sensor suhu DHT11 akan mengirim perintah ke NodeMCU untuk menghidupkan motor servo yang menggerakkan atap jemuran menjadi tertutup. Sebaliknya, jika sensor DHT11 mendeteksi tidak adanya mendung, maka NodeMCU akan mengirimkan nilai lebih dari 25 derajat pada database sehingga NodeMCU akan mengubah pergerakan motor servo menjadi 180 derajat sehingga atap jemuran menjadi terbuka.

4.1.2 Analisis Aplikasi Android

Aplikasi android yang digunakan untuk sistem kendali jemuran antisipasi hujan adalah *Smart Clothesline*. Aplikasi *Smart Clothesline* memiliki tiga menu yaitu menu beranda, menu kendali, dan menu informasi. Menu beranda berisi data monitor dari sensor hujan FC-37, sensor cahaya LDR, sensor suhu dan kelembaban DHT11, status atap

jemuran, dan mode atap jemuran. Apabila sensor hujan FC-37 pada database bernilai nol maka pada monitor aplikasi android menampilkan gambar hujan dan status cuaca hujan. Sebaliknya, apabila sensor hujan FC-37 pada database bernilai satu maka pada monitor aplikasi android menampilkan gambar cerah dan status cuaca cerah.

Pada monitor suhu, aplikasi android menampilkan gambar suhu dan nilai suhu dalam derajat. Pada monitor kelembaban, aplikasi android menampilkan gambar kelembaban dan nilai kelembaban dalam persen. Pada status atap jemuran, apabila motor servo bergerak membuka atap jemuran, maka monitor aplikasi android menampilkan status atap terbuka. Sebaliknya apabila motor servo bergerak menutup atap jemuran, maka monitor aplikasi android menampilkan status atap menutup. Pada mode, apabila mode kendali atap jemuran, bernilai satu maka monitor aplikasi android menampilkan status mode menjadi manual. Sebaliknya, apabila mode kendali atap jemuran, bernilai nol maka monitor aplikasi android menampilkan status mode menjadi otomatis.

Menu kendali pada aplikasi *Smart Clothesline* berfungsi untuk mengendalikan atap jemuran secara manual dan otomatis. Apabila menggunakan mode manual, pilih pilihan manual kemudian tekan tombol ubah, maka atap jemuran modenya berubah menjadi manual. Sebaliknya, Apabila menggunakan mode otomatis, pilih pilihan otomatis kemudian tekan tombol ubah, maka atap jemuran modenya berubah menjadi otomatis. Tombol buka dan tutup pada menu kendali akan aktif apabila mode status modenya manual. Sebaliknya, tombol buka dan tutup pada menu kendali akan tidak aktif apabila mode status modenya otomatis. Menu informasi pada aplikasi *Smart Clothesline* berisi informasi tentang deskripsi dari aplikasi.

4.2 Pembahasan Sistem

4.2.1 Implementasi Hardware

Komponen – komponen utama yang digunakan pada *hardware* Sistem Kendali Jemuran Antisipasi Hujan adalah sebagai berikut :

- a. NodeMCU ESP8266 merupakan komponen utama dalam Sistem Kendali Jemuran Antisipasi Hujan. NodeMCU merupakan kontroler yang akan memberikan perintah

dalam hal kendali atap jemuran, mengirim data ke database, dan mengambil data dari database.

- b. Sensor Hujan FC-37 merupakan sensor yang digunakan untuk mendeteksi kondisi adanya hujan. Apabila terindikasi adanya hujan, maka sensor ini akan mengirim perintah pada NodeMCU untuk menghidupkan motor servo agar atap jemuran bergerak menjadi terbuka. Begitu juga sebaliknya.
- c. Sensor Cahaya LDR merupakan sensor yang digunakan untuk mendeteksi kondisi cuaca pagi atau malam hari. Apabila kondisi cuaca pagi hari, maka sensor ini akan mengirim perintah pada NodeMCU untuk menghidupkan motor servo agar atap jemuran bergerak menjadi terbuka. Begitu juga sebaliknya.
- d. Sensor Suhu dan Kelembaban DHT11 merupakan sensor yang digunakan untuk mendeteksi cuaca mendung. Apabila kondisi cuaca mendung, maka sensor ini akan mengirim perintah pada NodeMCU untuk menghidupkan motor servo agar atap jemuran bergerak menjadi tertutup.
- e. Motor Servo SG90 merupakan komponen yang digunakan untuk menggerakkan atap jemuran. Motor servo ini dapat dikendalikan melalui aplikasi android dan dari hasil pengukuran sensor hujan, sensor cahaya, dan sensor suhu dan kelembaban.

4.2.2 Dimensi Hardware

Dimensi *hardware* merupakan visualisasi dari Sistem Kendali Jemuran Antisipasi Hujan yang memiliki bentuk, ukuran, dan bagian dari alat seperti pada Gambar 4.1.



Gambar 4.1 Maket Sistem Kendali Jemuran Antisipasi Hujan

Hardware dari Sistem Kendali Jemuran Antisipasi Hujan ditempatkan pada maket rumah *laundry* dengan ukuran 30cm x 20cm yang terbuat dari triplek dan *impraboard*. Pada bagian dalam samping kanan dan kiri rumah terdapat motor servo SG90 yang digunakan untuk membuka dan menutupnya atap jemuran. Sensor hujan FC-37, sensor cahaya LDR, dan sensor suhu dan kelembaban DHT11 ditempatkan diluar rumah tepatnya pada atap rumah *laundry*. Pada bagian dalam belakang dari maket rumah laundry berfungsi untuk menempatkan NodeMCU, dan *board*.

4.2.3 Pengujian Fungsionalitas Hardware

Hasil pengujian fungsionalitas *hardware* pada maket rumah *laundry* dapat dilihat pada Tabel 4.1.

Tabel 4.1 Hasil Pengujian Fungsionalitas *Hardware*

No	Pengujian	Tujuan	Hasil
1.	Pengujian sensor hujan FC-37	Sebagai pendeteksi cuaca hujan sehingga motor servo	Berhasil mendeteksi jika cuaca hujan, maka motor servo bergerak menutup

		dapat membuka dan menutup.	atap jemuran. Apabila tidak hujan, maka motor servo bergerak membuka atap jemuran.
2.	Pengujian sensor cahaya LDR	Sebagai pendeteksi cuaca pagi atau malam hari sehingga motor servo dapat membuka dan menutup.	Berhasil mendeteksi jika cuaca pagi hari (nilai sensor lebih dari 500), maka motor servo bergerak membuka atap jemuran. Apabila malam hari (nilai sensor kurang dari 500), maka motor servo bergerak menutup atap jemuran.
3.	Pengujian sensor suhu dan kelembaban DHT11	Sebagai pendeteksi cuaca mendung sehingga motor servo dapat membuka dan menutup.	Berhasil mendeteksi jika cuaca mendung (suhu kurang dari 25 derajat), maka motor servo bergerak menutup atap jemuran. Apabila tidak mendung (suhu lebih dari 25 derajat), maka motor servo bergerak menutup atap jemuran.
4.	Pengujian motor servo SG90	Sebagai penggerak atap jemuran yang dapat dikendalikan secara otomatis atau manual.	Berhasil jika dikontrol melalui aplikasi android maupun dari hasil pengukuran sensor hujan, sensor cahaya, dan sensor

			suhu dan kelembaban, motor servo akan membuka dan menutup sesuai perintah pada sistem.
--	--	--	--

Selanjutnya dilakukan pengujian fungsionalitas *hardware* sebagai Sistem Kendali Jemuran Antisipasi Hujan.

1) Pengujian Sensor Hujan FC-37

Pengujian sensor hujan FC-37 dikatakan berhasil apabila nilai dari sensor hujan nol, maka motor servo akan bergerak menutup atap jemuran. Apabila nilai dari sensor hujan satu, maka motor servo akan bergerak membuka atap jemuran. Pengujian sensor hujan dilakukan dengan meneteskan air ke sensor hujan. Berikut Tabel 4.2 merupakan hasil pengujian sensor hujan.

Tabel 4.2 Hasil Pengujian Sensor Hujan

No	Jumlah Tetesan	Kondisi Atap (Setelah Ditetesi Air)	Kondisi Atap (Setelah 1 Menit)
1	3 tetesan	Tertutup	Terbuka (8 detik)
2	6 tetesan	Tertutup	Terbuka (15 detik)
3	9 tetesan	Tertutup	Terbuka (21 detik)
4	12 tetesan	Tertutup	Terbuka (29 detik)
5	15 tetesan	Tertutup	Terbuka (35 detik)
6	18 tetesan	Tertutup	Terbuka (43 detik)
7	22 tetesan	Tertutup	Terbuka (51 detik)
8	25 tetesan	Tertutup	Terbuka (59 detik)
9	28 tetesan	Tertutup	Tertutup
10	31 tetesan	Tertutup	Tertutup

Seperti yang terlihat pada Tabel 4.2, pengujian dilakukan sebanyak sepuluh kali. Pada pengujian pertama ditetaskan air sebanyak tiga kali, kondisi atap langsung tertutup. Kemudian tetesan air dinaikkan setiap tiga kali dengan jumlah pengujian sepuluh kali dan kondisi atap setelah ditetesi air tertutup semua. Kemudian kondisi atap setelah 1 menit pada pengujian satu sampai delapan atap menjadi terbuka. Ini membuktikan bahwa pengujian sensor hujan apabila telah ditetesi air maka kondisi menutup. Gambar 4.2 menunjukkan atap telah tertutup.



Gambar 4.2 Pengujian Sensor Hujan dengan Tiga Tetesan Kondisi Atap Tertutup

2) Pengujian Sensor Cahaya LDR

Pengujian sensor cahaya LDR dikatakan berhasil apabila pagi hari (nilai dari sensor lebih dari 500), maka motor servo akan bergerak membuka atap jemuran. Apabila malam hari (nilai dari sensor kurang dari 500), maka motor servo akan bergerak menutup atap jemuran. Pengujian dilakukan di luar ruangan dengan kondisi pagi hari, siang hari, dan malam hari. Tabel 4.3 berikut adalah hasil pengujian sensor ldr.

Tabel 4.3 Hasil Pengujian Sensor LDR

No	Waktu	Intensitas Sensor LDR	Kondisi Atap
1	08.00 WIB	521	Terbuka
2	08.05 WIB	524	Terbuka

3	08.10 WIB	520	Terbuka
4	08.15 WIB	521	Terbuka
5	13.00 WIB	703	Terbuka
6	13.05 WIB	703	Terbuka
7	13.10 WIB	703	Terbuka
8	13.15 WIB	704	Terbuka
9	17.00 WIB	470	Tertutup
10	17.05 WIB	455	Tertutup
11	17.10 WIB	443	Tertutup
12	17.15 WIB	431	Tertutup

Tabel 4.3 merupakan tabel pengujian sensor LDR. Pengambilan data dilakukan pada setiap 5 menit sekali. pengujian ini berdasarkan data pengukuran intensitas cahaya dengan sensor LDR. Terlihat pada Tabel 4.3 bahwa kondisi atap terbuka terbuka pada saat data diatas nilai 500 Lux. Atap akan menutup pada saat nilai intensitas cahaya dibawah 500 Lux. Gambar 4.3 Hasil Pengujian Pertama Sensor LDR Atap Terbuka.



Gambar 4.3 Hasil Pengujian Pertama Sensor LDR Atap Terbuka

3) Pengujian Sensor Suhu dan Kelembaban DHT11

Pengujian sensor suhu dan kelembaban DHT11 dikatakan berhasil apabila cuaca mendung (nilai suhu dari sensor kurang dari 25 derajat), maka motor servo akan bergerak menutup atap jemuran. Apabila tidak mendung (nilai suhu dari sensor lebih dari 25 derajat), maka motor servo akan bergerak membuka atap jemuran. Pengujian dilakukan di luar ruangan dengan kondisi pagi hari, siang hari, dan malam hari. Tabel 4.4 berikut adalah hasil pengujian sensor suhu dan kelembaban DHT11.

Tabel 4.4 Hasil Pengujian Sensor DHT11

No	Waktu	Suhu (Celcius)	Kondisi Atap
1	08.00 WIB	26	Terbuka
2	08.05 WIB	26	Terbuka
3	08.10 WIB	26	Terbuka
4	08.15 WIB	26	Terbuka
5	13.00 WIB	29	Terbuka

6	13.05 WIB	29	Terbuka
7	13.10 WIB	29	Terbuka
8	13.15 WIB	29	Terbuka
9	17.00 WIB	27	Tertutup
10	17.05 WIB	27	Tertutup
11	17.10 WIB	27	Tertutup
12	17.15 WIB	26	Tertutup

Tabel 4.4 merupakan tabel pengujian sensor suhu dan kelembapan DHT11. Pengambilan data dilakukan pada pukul 08.00 WIB, setiap 5 menit sekali dari keadaan awal sampai keadaan menuju set waktu. Tabel di atas menjelaskan suhu apabila suhu lebih dari 25 derajat celcius, maka atap akan terbuka. Sebaliknya, jika suhu dibawah 25 derajat celcius, maka atap akan tertutup.



Gambar 4.4 Pengujian Sensor DHT11 Atap Terbuka

4) Pengujian Motor Servo

Pengujian motor servo dikatakan berhasil apabila motor servo bergerak menutup atap jemuran ketika nilai status di database adalah nol. Apabila motor servo bergerak membuka atap jemuran ketika nilai status di database adalah satu. Apabila motor

servo dalam mode manual, maka hasil pengukuran sensor hujan FC-37, sensor cahaya LDR, dan sensor suhu dan kelembaban DHT11 tidak berpengaruh terhadap pergerakan motor servo. Apabila motor servo dalam mode otomatis, maka hasil pengukuran sensor hujan, sensor cahaya, dan sensor suhu dan kelembaban berpengaruh terhadap pergerakan motor servo.

Pengujian dilakukan dengan melakukan pergerakan membuka dan menutupnya atap jemuran dengan motor servo. Ketika status mode di database adalah satu, maka modenya adalah manual. Ketika mode manual, status kendali atap di database adalah nol, maka atap jemuran tertutup seperti yang terlihat pada Gambar 4.5 dan Gambar 4.6. Ketika status kendali atap di database adalah satu, maka atap jemuran terbuka seperti yang terlihat pada Gambar 4.7 dan Gambar 4.8. Ketika status mode di database adalah nol, maka modenya adalah otomatis. Ketika mode otomatis, maka sensor berpengaruh terhadap pergerakan atap jemuran motor servo seperti yang terlihat pada Gambar 4.9.

nama_kendali character varying (191)	status_kendali character varying (255)
Motor Servo	1

Gambar 4.5 Hasil Pengujian Mode Manual dengan Status Kendali Atap adalah Satu



Gambar 4.6 Hasil Pengujian Mode Manual Atap Jemuran Tertutup

nama_kendali	status_kendali
character varying (191)	character varying (255)
Motor Servo	0

Gambar 4.7 Hasil Pengujian Mode Manual dengan Status Kendali Atap adalah Nol



Gambar 4.8 Hasil Pengujian Mode Manual Atap Jemuran Terbuka

nama_kendali character varying (191)	status_kendali character varying (255)
Mode	0

Gambar 4.9 Hasil Pengujian Mode Otomatis di Database

4.2.4 Implementasi Aplikasi

Aplikasi *Smart Clothesline* dapat dijalankan di beberapa OS Android seperti yang terlihat pada Tabel 4.5.

Tabel 4.5 Implementasi Smart Clothesline

No	OS Android	Keterangan
1	Jelly Bean	Berhasil
2	KitKat	Berhasil
3	Lollipop	Berhasil
4	Marshmallow	Berhasil
5	Nougat	Berhasil

4.2.5 Pengujian Fungsionalitas Aplikasi

Hasil pengujian fungsionalitas aplikasi *Smart Clothesline* seperti pada Tabel 4.6.

Tabel 4.6 Hasil Pengujian Fungsionalitas Aplikasi

No.	Pengujian	Tujuan	Hasil
1	Pengujian halaman <i>home</i> .	Mengetahui monitoring cuaca, suhu, kelembaban, status atap, dan status mode.	Berhasil menampilkan informasi monitoring cuaca, suhu, kelembaban, status atap, dan status mode.
2	Pengujian kendali atap jemuran dengan mode manual.	Mengendalikan buka dan tutup atap	Berhasil mengendalikan buka dan tutup atap jemuran.

		jemuran melalui android.	
3	Pengujian kendali atap jemuran dengan mode otomatis.	Mengendalikan buka dan tutup atap jemuran secara otomatis.	Berhasil mengendalikan buka dan tutup atap jemuran dengan mengaktifkan mode otomatis.
4	Pengujian halaman informasi.	Mengetahui informasi tentang aplikasi Smart Clothesline	Berhasil menampilkan informasi tentang aplikasi Smart Clothesline

Lalu akan dilakukan pengujian fungsionalitas aplikasi dalam menjalankan sistem aplikasi smart clothesline seperti berikut ini :

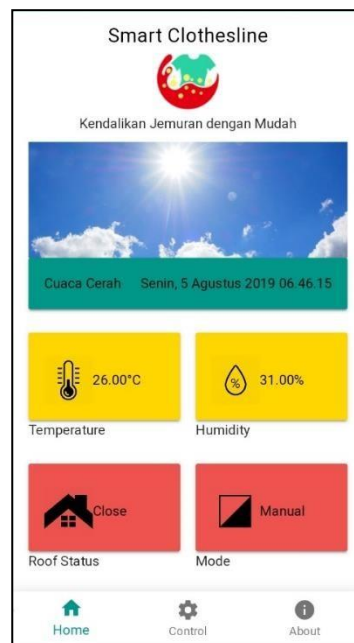
1) Pengujian halaman home

Pada pengujian halaman home dilakukan dengan mengakses halaman home. Pengujian ini dikatakan berhasil apabila dapat membaca deteksi sensor sesuai dengan perintah yang ada pada sistem. Apabila cuaca hujan, maka akan menampilkan gambar hujan seperti pada Gambar 4.10.



Gambar 4.10 Pengujian Aplikasi Halaman Home Apabila Cuaca Hujan

Apabila cuaca cerah, maka akan menampilkan cuaca cerah seperti pada Gambar 4.11



Gambar 4.11 Pengujian Aplikasi Halaman Home Apabila Cuaca Cerah

2) Pengujian halaman kendali

Pada pengujian halaman kendali dilakukan pengujian dengan mengakses halaman kendali. Apabila menekan menekan tombol buka pada mode manual maka akan tampil seperti pada Gambar 4.21. Status atap menampilkan atap telah terbuka.



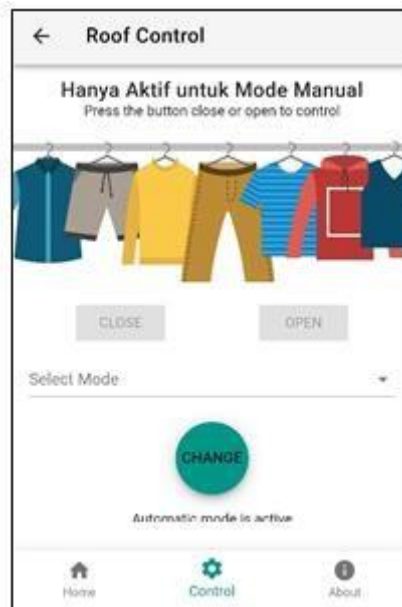
Gambar 4.12 Pengujian Aplikasi Halaman Kendali Tombol Buka Mode Manual

Apabila menekan menekan tombol tutup pada mode manual maka akan tampil seperti pada Gambar 4.13. Status atap menampilkan atap telah tertutup.



Gambar 4.13 Pengujian Aplikasi Halaman Kendali Tombol Tutup Mode Manual

Apabila memilih pilihan otomatis dan tekan tombol ubah, maka sistem akan membaca mode otomatis seperti pada Gambar 4.14. Tombol buka dan tutup menjadi tidak aktif.



Gambar 4.14 Pengujian Aplikasi Halaman Kendali Mode Otomatis

3) Pengujian halaman informasi

Pada pengujian halaman informasi dilakukan dengan mengakses halaman informasi seperti yang terdapat pada Gambar 4.15. Halaman ini digunakan untuk menampilkan informasi aplikasi.



Gambar 4.15 Pengujian Aplikasi Halaman Informasi

4.2.6 Hasil Pengujian Tingkat Kepuasan Pengguna

Pada bagian ini berisi hasil pengujian Sistem Smart Clothesline dan Aplikasi Smart Clothesline yang dapat dilihat di Tabel 4.7.

Tabel 4.7 Hasil Pengujian Tingkat Kepuasan Pengguna

No	Nama	Daftar Pertanyaan					Total Nilai
		1	2	3	4	5	
1	Berkah laundry	4	4	5	5	3	21
2	Shinta laundry	3	4	4	5	3	19
3	Kurnia laundry	4	4	4	5	3	20

4	Aksara laundry	4	5	5	5	3	22
5	Ocean laundry	4	4	5	5	4	22
6	Rhizal laundry	4	5	4	5	4	22
7	Laundry pojok	4	4	4	5	4	21
8	Laundry mandiri	4	4	4	4	4	20
9	Nino Laundry	4	4	4	4	4	20
10	Hai Laundry	4	4	4	5	4	21
Total							208

Jumlah kepuasan maksimal 25 (indikator penilaian) x 10 (pengguna) = 250

$$\begin{aligned}
 \text{Presentase kepuasan pengguna (\%)} &= \frac{\text{Total kepuasan pengguna}}{\text{Total kepuasan maksimum pengguna}} \times 100\% \\
 &= \frac{208}{250} \times 100\% \\
 &= 83\%
 \end{aligned}$$

Dari hasil jajak pendapat di atas, diperoleh nilai 83%. Berdasarkan Tabel 4.7 dapat dikategorikan sangat memuaskan, sehingga aplikasi ini sesuai yang diharapkan sesuai dengan kebutuhan.

BAB V PENUTUP

5.1 Kesimpulan

Berdasarkan hasil pembuatan Tugas Akhir ini, dapat diambil kesimpulan bahwa Sistem Kendali Jemuran Antisipasi Hujan dapat bekerja dalam dua mode yaitu manual dan otomatis. Kondisi manual dapat bekerja ketika user memerintahkan untuk menutup atap jemuran, maka atap akan menutup. Ketika user memerintahkan untuk membuka atap jemuran, maka atap akan membuka. Kondisi otomatis dapat bekerja ketika sensor hujan mendeteksi adanya hujan, maka atap jemuran akan menutup, ketika sensor hujan tidak mendeteksi adanya hujan, maka atap jemuran akan membuka, ketika sensor ldr menunjukkan nilai lebih dari sama dengan 500 lux, maka atap jemuran akan terbuka, ketika sensor ldr menunjukkan nilai kurang dari 500 lux, maka atap jemuran akan tertutup, ketika sensor suhu dan kelembaban DHT11 menunjukkan nilai lebih dari 25 derajat celcius, maka atap jemuran akan terbuka. Dalam pengujian yang dilakukan dapat dinyatakan bahwa Sistem Kendali Jemuran Antisipasi Hujan dapat bekerja 100% dengan kasus uji sebanyak 10 *laundry*. Hasil pengujian tingkat kepuasan pengguna Sistem Kendali Jemuran Antisipasi Hujan menunjukkan angka 83% (sangat memuaskan) yang tersebar sebanyak 10 *laundry* di kawasan Tembalang.

5.2 Saran

Mengingat berbagai keterbatasan dalam penulisan dan penyusunan Tugas Akhir ini, maka diberikan untuk pengembangan sistem sebagai berikut :

- 1) Perlu adanya penambahan fitur *blower* sebagai solusi peningkatan mutu produksi *laundry*.
- 2) Perlu adanya perubahan regulasi pada sumber daya dengan menggunakan PLTS (Pembangkit Listrik Tenaga Surya) sebagai solusi penghematan dan *backup* energi.
- 3) Perlu adanya sosialisasi terhadap *laundry* secara luas.

DAFTAR PUSTAKA

- Candra, J. dan Very, K. (2018) 'Desain System Smart Clothesline Berbasis Arduino Dengan Metode Logika Fuzzy'.
- Darusman, A. D., Dahlan, M. dan Hilyana, F. S. (2018) 'Rancang Bangun Prototype Alat Penjemur Pakaian Otomatis Berbasis Arduino Uno', *Simetris: Jurnal Teknik Mesin, Elektro dan Ilmu Komputer*.
- Faisal, S., Aliyadi dan Angga, P. (2018) 'Rancang Bangun Jemuran Otomatis Berbasis Web dengan Kendali Raspberry PI'.
- Fraden, Jacob (2003). "Handbook of Modern Sensors: Third edition". California: Springer.
- Frank D. Petruzella., (2001), Elektronika Industri, Penerbit Andi, Penerjemah Suminto, Drs. MA., Yogyakarta.
- Guk-guk, E. H. R. (2015) 'Jemuran Otomatis Dengan Menggunakan Sensor Ldr, Sensor Hujan Dan Sensor Kelembaban'.
- Hapsari, N. (2013). Evaluasi Program Pengolahan Sampah Berskala Keluarga Di Kelurahan Tembalang. Teknik Perencanaan Wilayah Kota, 165–176.
- Kobandaha, T., Mosey, H. I. R. dan Suoth, V. A. (2018) 'Sistem Kontrol Atap Otomatis Tempat Penjemuran Berbasis Mikrokontroler Arduino UNO dan Node Sensor'.
- Listia, Y. (2017). Analisis Strategi Bersaing pada UKM Laundry. *Jurnal Ilmu Administrasi Bisnis Undip*.
- Mufida, E. dan Abas, A. (2017) 'Alat Pengendali Atap Jemuran Otomatis Dengan Sensor Cahaya Dan Sensor Air Berbasiskan Mikrokontroler ATmega16'.
- Novianty Keyza, Lubis, Tony. (2012). Perancangan Prototipe Sistem Penerangan Otomatis Ruangan Berjendela Berdasarkan Intensitas Cahaya. Seminar Nasional Teknologi Informasi 2012. Universitas Tarumanegara Fakultas Teknologi Informasi.
- Prasetya, N. A., dan Sunaryo, P. B. (2013). Faktor-Faktor Yang Mempengaruhi Harga Lahan Di Kawasan Banjarsari Kelurahan Tembalang, Semarang. Teknik PWK Undip, 223–232.
- Roger S. Pressman, P.D. (2010). Rekayasa Perangkat Lunak : Pendekatan Praktisi.

Yogyakarta: Andi.

Sulastri, R. (2016). Prototype Kendali Buka Tutup Atap dan Penyiraman Tanaman Cabai Berbasis Mikrokontroler dan SMS Gateway. Politeknik Negeri Sriwijaya.

Yuwono, Y. C. dan Alam, S. (2018) 'Rancang Bangun Sistem Jemuran Otomatis Berbasis Arduino Uno', *Ejournal Kajian Teknik Elektro*.

Lampiran 1 Koding Arduino

```
// Include library yang diperlukan
#include <Arduino.h>
#include <ESP8266WiFi.h>
#include <WiFiClient.h>
#include <ESP8266HTTPClient.h>
#include <ESP8266WebServer.h>
#include <DHT.h>
#include <Servo.h>

const char *ssid = "TA";
const char *password = "123456789";

//Pin Hujan
int HujanDigital = D6;

//Pin Servo
int servoPinSatu = D7;
int servoPinDua = D0;
Servo ServoSatu;
Servo ServoDua;

//#define DHTPIN 0 //D3 Pin DHT11
#define DHTPIN D1 //Pin DHT11
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);

// Pin LDR
int CahayaAnalog = A0;

// Buat object http
HTTPClient http;

// API
String url1 =
"http://apismartclothesline.herokuapp.com/api/insert";
String url2 =
"http://apismartclothesline.herokuapp.com/api/motorservo";
String url3 =
"http://apismartclothesline.herokuapp.com/api/mode";

void setup() {
  Serial.begin(115200);
```

```

Serial.setDebugOutput(false);
for (uint8_t t = 4; t > 0; t--) {
    Serial.printf("[SETUP] Tunggu %d...\n", t);
    Serial.flush();
    delay(1000);
}
WiFi.mode(WIFI_STA);

WiFi.begin(ssid, password);
Serial.println("");
Serial.print("Connecting");

while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
}

//If connection successful show IP address in serial
monitor
Serial.println("");
Serial.print("Connected to ");
Serial.println(ssid);
Serial.print("IP address: ");
Serial.println(WiFi.localIP());

dht.begin(); //memulai pembacaan sensor dht11
ServoSatu.attach(servoPinSatu);
ServoDua.attach(servoPinDua);
}

void monitorData() {

    // Hujan
    int hujan = digitalRead(HujanDigital);
    // Suhu
    float suhu = dht.readTemperature();
    // Kelembaban
    float kelembaban = dht.readHumidity();
    // LDR
    int cahaya = analogRead(CahayaAnalog);
    //Hujan = 1 (Atap Terbuka), Hujan = 0 (Atap Tertutup)
    Serial.print("Status Hujan= ");
    if(hujan == 1) {
        Serial.println("Tidak Hujan");
    } else {

```

```

        Serial.println("Hujan");
    }
    Serial.print("Status Hujan= ");
    Serial.println(hujan);
    Serial.print("Kelembaban: ");
    Serial.print(kelembaban);
    Serial.println("%");
    Serial.print("Suhu : ");
    Serial.print(suhu);
    Serial.println("'C");
    Serial.print("Cahaya= ");
    Serial.println(cahaya);

    String postData;
    postData = "hujan=" + String(hujan) + "&kelembaban="
+ String(kelembaban) + "&suhu=" + String(suhu) +
"&cahaya=" + String(cahaya);
    http.begin(url1);
    http.addHeader("Content-Type", "application/x-www-
form-urlencoded");
    int httpCode = http.POST(postData);
    String payload = http.getString();
    Serial.println(httpCode);
    Serial.println(payload);
    http.end();
}

void kendali()
{
    http.begin(url3);
    http.addHeader("Content-Type", "application/x-www-
form-urlencoded");
    int httpCode = http.GET();
    String modepayload = http.getString(); // get data
from webhost continuously
    Serial.print("Status Mode : "); //0 = Otomatis, 1 =
Manual
    if(modepayload == "0") {
        Serial.print("Otomatis");
        Serial.print('\n');
    } else {
        Serial.print("Manual");
        Serial.print('\n');
    }
}

```

```

    Serial.print("Status Mode : "); //0 = Otomatis, 1 =
Manual
    Serial.print(modepayload);
    Serial.print('\n');
    if (modepayload == "1")
    {
        http.begin(url2);
        http.addHeader("Content-Type", "application/x-www-
form-urlencoded");
        int httpCode = http.GET();
        String payload = http.getString(); // get data from
webhost continuously
        Serial.print("Motor Servo : ");
        Serial.print(payload);
        Serial.print('\n');
        if (payload == "0")
        {
            tutup();
        }
        else
        {
            buka();
        }
    }
    else // Mode Otomatis = 0
    {
        int hujan = digitalRead(HujanDigital);
        int cahaya = analogRead(CahayaAnalog);
        float suhu = dht.readTemperature();
        // Hujan = 1 (Atap Terbuka / Tidak Hujan), Hujan =
0 (Atap Tertutup / Hujan)
        if (hujan == 1 && cahaya >= 500 && suhu >= 25) {
            Serial.print("Atap Terbuka | Tidak Hujan | Siang
| Tidak Mendung");
            Serial.print('\n');
            buka();
        }

        if (hujan == 0 && cahaya <= 500 && suhu <= 25) {
            Serial.print("Atap Tertutup | Hujan | Malam |
Mendung");
            Serial.print('\n');
            tutup();
        }
    }

```

```

        if (hujan == 0) {
            Serial.print("Atap Tertutup | Hujan | Malam |
Mendung");
            Serial.print('\n');
            tutup();
        }

        if (cahaya <= 500) {
            Serial.print("Atap Tertutup | Hujan | Malam |
Mendung");
            Serial.print('\n');
            tutup();
        }

        if (suhu <= 25) {
            Serial.print("Atap Terbuka | Tidak Hujan | Siang
| Tidak Mendung");
            Serial.print('\n');
            tutup();
        }

    }
    http.end();
}

//fungsi utk servo
void buka()
{
    ServoSatu.write(0);
    ServoDua.write(145);
}

void tutup()
{
    ServoDua.write(10);
    ServoSatu.write(155);
}

void loop()
{
    kendali();
    monitorData();
    delay(10000);
}

```

Lampiran 2 Koding Aplkasi

1. Koding halaman home.vue

```
<template>
  <div>
    <v-card>
      <v-container fluid grid-list-lg>
        <v-flex xs12 text-xs-center>
          <v-toolbar-title class="black--text title">Smart
Clothesline</v-toolbar-title>
          <v-img
            :src="require('@assets/static/logoapp.png')"
            width="70"
            height="70"
            class="imageposition"
          ></v-img>Kendalikan Jemuran dengan Mudah
        </v-flex>
        <v-layout row wrap>
          <v-flex xs12>
            <v-img
              v-if="monitor.hujan === '1.00'"
              :src="require('@assets/static/cerah.jpg')"
              aspect-ratio="2.75"
            ></v-img>
            <v-img v-else
              :src="require('@assets/static/hujan.jpeg')" aspect-
ratio="2.75"></v-img>
            <v-card color="teal" class="black--text">
              <v-card-actions>
                <v-flex>
                  <span v-if="monitor.hujan ===
'1.00'">Cuaca Cerah</span>
                  <span v-else>Cuaca Hujan</span>
                </v-flex>
                <v-flex class="text-xs-right">
                  <span>{{ tanggal }} {{ waktu}}</span>
                </v-flex>
              </v-card-actions>
            </v-card>
          </v-flex>
          <!-- Temperature -->
          <v-flex xs6 mt-2>
            <v-card color="yellow accent-4" class="black--
text">
              <v-card-title primary-title>
                <div xs6>
```

```

        <v-img
:src="require('@assets/static/term.png')" width="50"
height="50"></v-img>
    </div>
    <div xs6>
        <span>{{ monitor.suhu }}&#176;C</span>
    </div>
</v-card-title>
</v-card>
<p>Temperature</p>
</v-flex>
<!-- Humidity -->
<v-flex xs6 mt-2>
    <v-card color="yellow accent-4" class="black--
text">
        <v-card-title primary-title>
            <div xs6>
                <v-img
:src="require('@assets/static/humidity.png')" width="50"
height="50"></v-img>
            </div>
            <div xs6>
                <span>{{ monitor.kelembaban }}%</span>
            </div>
        </v-card-title>
    </v-card>
    <p>Humidity</p>
</v-flex>

<!-- Kendali Atap -->
<v-flex xs6 pt-0>
    <v-card color="red lighten-1" class="black--
text">
        <!-- Manual -->
        <v-card-title primary-title v-
if="mode.status_kendali === '1'">
            <div xs6>
                <v-img
:src="require('@assets/static/roof.png')" width="50"
height="50"></v-img>
            </div>
            <div xs6 v-if="motorservo.status_kendali ===
'0'">
                <span>Close</span>
            </div>
            <div xs6 v-if="motorservo.status_kendali ===
'1'">
                <span>Open</span>
            </div>

```



```

        </v-card-title>
        <!-- Otomatis -->
        <v-card-title primary-title v-else>
          <div xs6>
            <v-img
:src="require('@assets/static/roof.png')" width="50"
height="50"></v-img>
          </div>
          <div xs6>
            <span>Non Aktif</span>
          </div>
        </v-card-title>
      </v-card>
    <p>Roof Status</p>
  </v-flex>
  <!-- End Kendali Atap -->

  <!-- Mode -->
  <v-flex xs6 pt-0>
    <v-card color="red lighten-1" class="black--
text">
      <v-card-title primary-title>
        <div xs6>
          <v-img
:src="require('@assets/static/mode.png')" width="50"
height="50"></v-img>
          </div>
          <div xs6 v-if="mode.status_kendali === '0'">
            <span>Otomatis</span>
          </div>
          <div xs6 v-else>
            <span>Manual</span>
          </div>
        </v-card-title>
      </v-card>
      <p>Mode</p>
    </v-flex>
  <!-- End Mode -->
</v-layout>
</v-container>
</v-card>
</div>
</template>
<script>
export default {
  data: () => ({
    waktu: "",
    tanggal: "",
    monitor: [],

```

```

    motorservo: [],
    mode: []
  }},
  methods: {
    load() {
      let url = "/monitor";
      let motorservo = "/jsonmotorservo";
      let mode = "/jsonmode";
      this.axios
        .get(url)
        .then(response => {
          let monitor = response.data;
          this.monitor = monitor;
        })
        .catch(error => {
          let responses = error.response;
          console.log(responses);
        });
      this.axios
        .get(motorservo)
        .then(response => {
          let motorservo = response.data;
          this.motorservo = motorservo;
        })
        .catch(error => {
          let responses = error.response;
          console.log(responses);
        });
      this.axios
        .get(mode)
        .then(response => {
          let mode = response.data;
          this.mode = mode;
        })
        .catch(error => {
          let responses = error.response;
          console.log(responses);
        });
    },
    waktuFunction: function() {
      var datahari = [
        "Minggu",
        "Senin",
        "Selasa",
        "Rabu",
        "Kamis",
        "Jumat",
        "Sabtu"
      ];

```

```

        var databulan = [
            "Januari",
            "Februari",
            "Maret",
            "April",
            "Mei",
            "Juni",
            "Juli",
            "Agustus",
            "September",
            "Oktober",
            "November",
            "Desember"
        ];
        var v = this;
        var myDate = new Date().getDate();
        var _hari = new Date().getDay();
        var _bulan = new Date().getMonth();
        var _tahun = new Date().getFullYear();
        var hari = datahari[_hari];
        var bulan = databulan[_bulan];
        var tahun = _tahun < 1000 ? _tahun + 1900 : _tahun;
        v.tanggal = hari + ", " + myDate + " " + bulan + " " +
tahun;
        v.waktu = new Date().toLocaleTimeString();
    },
    mounted() {
        this.waktuFunction();
        this.load();
    }
};
</script>
<style scoped>
.imageposition {
    display: block;
    margin-left: auto;
    margin-right: auto;
    width: 50%;
}
</style>

```

2. Koding halaman control.vue

```
<template>
  <div>
    <v-toolbar class="black--text">
      <v-btn icon exact to="/">
        <v-icon>arrow_back</v-icon>
      </v-btn>
      <v-toolbar-title>Roof Control</v-toolbar-title>
    </v-toolbar>
    <v-container text-xs-center v-if="mode.status_kendali
=== '0'">
      <p color="teal" class="title mb-0">Hanya Aktif untuk
Mode Manual</p>
      <div>Press the button close or open to control</div>
    </v-container>
    <v-container text-xs-center v-if="mode.status_kendali
=== '1'">
      <p color="teal" class="title mb-0" v-
if="motorservo.status_kendali === '0'">Roof is Closed</p>
      <p color="teal" class="title mb-0" v-
if="motorservo.status_kendali === '1'">Roof is Opened</p>
      <div>Press the button close or open to control</div>
    </v-container>

    <v-img :src="require('@assets/static/pakaian.jpg')"
aspect-ratio="2.75" width="100%"></v-img>
    <v-container>
      <v-layout row wrap>
        <v-flex xs6>
          <div class="text-xs-center" v-
if="mode.status_kendali === '1'">
            <v-btn color="teal" v-
on:click="btnClose">Close</v-btn>
          </div>
          <div class="text-xs-center" v-else>
            <v-btn color="teal" disabled>Close</v-btn>
          </div>
        </v-flex>
        <v-flex xs6>
          <div class="text-xs-center" v-
if="mode.status_kendali === '1'">
            <v-btn color="teal" v-
on:click="btnOpen">Open</v-btn>
          </div>
          <div class="text-xs-center" v-else>
            <v-btn color="teal" disabled>Open</v-btn>
          </div>
        </v-flex>
      </v-layout>
    </v-container>
  </div>
</template>
```

```

        <v-flex xs12>
          <v-form ref="form" v-
on:submit.prevent="updateMode">
            <v-select
              v-model="status_kendali"
              :items="items"
              item-value="id"
              item-text="kendali"
              label="Select Mode"
            ></v-select>
            <div class="text-xs-center">
              <v-btn color="teal" fab large
type="submit">Change</v-btn>
            </div>
          </v-form>
        </v-flex>
        <v-flex xs12 pt-2>
          <p class="mb-0 text-xs-center" v-
if="mode.status_kendali === '1'">Manual mode is active</p>
          <p class="mb-0 text-xs-center" v-else>Automatic
mode is active</p>
        </v-flex>
      </v-layout>
    </v-container>
  </div>
</template>
<script>
export default {
  data: () => ({
    motorservo: [],
    mode: [],
    status_kendali: null,
    items: [
      {
        id: "0",
        kendali: "Automatic"
      },
      {
        id: "1",
        kendali: "Manual"
      }
    ]
  }),
  methods: {
    load() {
      let motorservo = "/jsonmotorservo";
      let mode = "/jsonmode";
      // Motor Servo otomatis 0 = Menutup, 1 = Membuka
      this.axios

```

```

        .get(motorservo)
        .then(response => {
            let motorservo = response.data;
            this.motorservo = motorservo;
        })
        .catch(error => {
            let responses = error.response;
            console.log(responses);
        });
    // Mode Manual dan otomatis 0 = Otomatis, 1 = Manual
    this.axios
        .get(mode)
        .then(response => {
            let mode = response.data;
            this.mode = mode;
        })
        .catch(error => {
            let responses = error.response;
            console.log(responses);
        });
    },
    btnOpen() {
        this.axios
            .post("/servobuka")
            .then(response => {
                this.load();
                console.log(response);
            })
            .catch(error => {
                console.log(error);
            });
    },
    btnClose() {
        this.axios
            .post("/servotutup")
            .then(response => {
                this.load();
                console.log(response);
            })
            .catch(error => {
                console.log(error);
            });
    },
    updateMode(e) {
        e.preventDefault();
        this.axios
            .post("/pilihmode", {
                status_kendali: this.status_kendali
            })

```

```

        .then(response => {
            this.load();
            this.status_kendali = "Select Mode";
            console.log(response);
        })
        .catch(error => {
            console.log(error);
        });
    }
},
mounted() {
    this.load();
}
};
</script>

```

3. Koding halaman about.vue

```

<template>
  <div>
    <v-toolbar class="black--text">
      <v-btn icon exact to="/">
        <v-icon>arrow_back</v-icon>
      </v-btn>
      <v-toolbar-title>About</v-toolbar-title>
    </v-toolbar>
    <v-container fluid grid-list-lg>
      <v-flex xs12>
        <v-img
          :src="require('@assets/static/logoapp.png')"
          width="70"
          height="70"
          class="imageposition"
        ></v-img>
        <p class="black--text title text-xs-center">Smart Clothesline</p>
        <p>Smart Clothesline adalah aplikasi kendali jemuran antisipasi hujan. Aplikasi ini dapat mengendalikan atap jemuran secara otomatis ataupun manual.</p>
      </v-flex>
    </v-container>
  </div>
</template>
<style scoped>
.imageposition {

```

```
display: block;  
margin-left: auto;  
margin-right: auto;  
width: 50%;  
}  
</style>
```