

**Perancangan Sistem Kendali Terpusat dan Penjadwalan AC
Kampus Universitas Kristen Duta Wacana Menggunakan
Protokol MQTT**

Skripsi



Disusun oleh:

Daniel Felix Setiawan

71140062

PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN DUTA WACANA
YOGYAKARTA

2019

**Perancangan Sistem Kendali Terpusat dan Penjadwalan AC
Kampus Universitas Kristen Duta Wacana Menggunakan
Protokol MQTT**

Skripsi



Diajukan kepada Program Studi Informatika Fakultas Teknologi Informasi
Universitas Kristen Duta Wacana
Sebagai salah satu syarat dalam memperoleh gelar Sarjana Komputer

Disusun oleh:

Daniel Felix Setiawan

71140062

PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN DUTA WACANA
YOGYAKARTA

2019

PERNYATAAN KEASLIAN SKRIPSI

Saya menyatakan dengan sesungguhnya bahwa skripsi dengan judul:

PERANCANGAN SISTEM KENDALI TERPUSAT DAN PENJADWALAN AC KAMPUS UNIVERSITAS KRISTEN DUTA WACANA MENGUNAKAN PROTOKOL MQTT

yang saya kerjakan untuk melengkapi sebagian persyaratan menjadi Sarjana Komputer pada pendidikan Sarjana Program Studi Informatika Fakultas Teknologi Informasi Universitas Kristen Duta Wacana, bukan merupakan tiruan atau duplikasi dari skripsi keserjanaan di lingkungan Universitas Kristen Duta Wacana maupun di Perguruan Tinggi atau instansi manapun, kecuali bagian yang sumber informasinya dicantumkan sebagaimana mestinya.

Jika dikemudian hari didapati bahwa hasil skripsi ini adalah hasil plagiasi atau tiruan dari skripsi lain, saya bersedia dikenai sanksi yakni pencabutan gelar keserjanaan saya.

Yogyakarta, 9 Januari 2020



DANIEL FELIX SETIAWAN
71140062

HALAMAN PERSETUJUAN

Judul Skripsi : PERANCANGAN SISTEM KENDALI TERPUSAT
DAN PENJADWALAN AC KAMPUS
UNIVERSITAS KRISTEN DUTA WACANA
MENGUNAKAN PROTOKOL MQTT

Nama Mahasiswa : DANIEL FELIX SETIAWAN

N I M : 71140062

Matakuliah : Skripsi (Tugas Akhir)

Kode : TIW276

Semester : Gasal

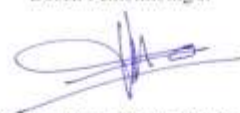
Tahun Akademik : 2019/2020

Telah diperiksa dan disetujui di
Yogyakarta,
Pada tanggal 9 Januari 2020

Dosen Pembimbing I


Laurentius Kuncoro Probo Saputra,
S.T., M.Eng.

Dosen Pembimbing II


Yuan Lukito, S.Kom., M.Cs.

HALAMAN PENGESAHAN

PERANCANGAN SISTEM KENDALI TERPUSAT DAN PENJADWALAN AC KAMPUS UNIVERSITAS KRISTEN DUTA WACANA MENGUNAKAN PROTOKOL MQTT

Oleh: DANIEL FELIX SETIAWAN / 71140062

Dipertahankan di depan Dewan Penguji Skripsi
Program Studi Informatika Fakultas Teknologi Informasi
Universitas Kristen Duta Wacana - Yogyakarta
Dan dinyatakan diterima untuk memenuhi salah satu syarat memperoleh gelar
Sarjana Komputer
pada tanggal 17 Desember 2019

Yogyakarta, 9 Januari 2020
Mengesahkan,

Dewan Penguji:

1. Laurentius Kuncoro Probo Saputra, S.T.,
M.Eng.
2. Yuan Lukito, S.Kom., M.Cs.
3. R. Gumbawan Santosa, Drs. M.M.
4. Sri Siswanto, Dr. Ir. M.Eng.

Dekan

(Restyandito, S.Kom., MSIS., Ph.D.)

Ketua Program Studi

(Gloria Virginia, Ph.D.)

UCAPAN TERIMA KASIH

Segala puji syukur penulis panjatkan kehadirat Tuhan Yang Maha Esa, karena atas berkat, rahmat, dan penyertaanNya penulis dapat menyelesaikan skripsi berjudul “Perancangan Sistem Kendali Terpusat dan Penjadwalan AC Kampus Universitas Kristen Duta Wacana Menggunakan Protokol MQTT” dengan baik.

Skripsi ini disusun untuk memenuhi salah satu syarat kelulusan dan mencapai gelar Sarjana Komputer (S.Kom) di Program Studi Informatika Fakultas Teknologi Informasi Universitas Kristen Duta Wacana Yogyakarta. Dalam penyusunan skripsi ini, penulis mendapatkan bantuan, bimbingan, serta dukungan dari berbagai pihak. Oleh karena itu, pada kesempatan ini penulis mengucapkan terima kasih kepada:

1. Tuhan Yang Maha Esa atas berkat dan penyertaanNya terhadap penulis selama mengerjakan penelitian ini
2. Kedua orang tua dan keluarga yang selalu memberikan dukungan dan semangat dalam mengerjakan tugas akhir ini
3. Bapak Ir. Hendry Feriadi, M.Sc., Ph.D selaku Rektor Universitas Kristen Duta Wacana
4. Bapak Restyandito, S.Kom, MSIS, Ph.D selaku Dekan Fakultas Teknologi Informasi Universitas Kristen Duta Wacana
5. Ibu Gloria Virginia, S.Kom, MAI, Ph.D selaku Kepala Program Studi Informatika Universitas Kristen Duta Wacana
6. Bapak L. Kuncoro Probo Saputra, S.T, M.Eng selaku dosen pembimbing I dan Bapak Yuan Lukito, S.Kom, M.Cs selaku dosen pembimbing II yang telah bersedia meluangkan waktu, tenaga, dan pikiran untuk memberikan bimbingan, arahan, dan masukan, serta memotivasi penulis selama mengerjakan penelitian ini
7. Bapak Kristian Adi Nugraha, S.Kom, M.T selaku dosen wali yang telah memberikan dukungan dan pengarahan selama masa perkuliahan
8. Seluruh dosen Informatika Universitas Kristen Duta Wacana yang telah memberikan banyak ilmu dan pengetahuan selama masa perkuliahan

9. Bapak Abet Narisworo selaku kepala PPLK UKDW yang telah mengizinkan penulis untuk mengimplementasikan sistem yang dibuat dalam penelitian ini pada laboratorium komputer Agape lantai 2 UKDW, meminjamkan peralatan yang dibutuhkan untuk pemasangan modul, serta memberikan dukungan dan arahan kepada penulis dalam melakukan penelitian ini
10. Para petugas laboratorium komputer Agape lantai 2 UKDW yang telah memberikan dukungan kepada penulis dalam melakukan penelitian ini, serta memberikan arahan dan masukan dalam mengimplementasikan sistem yang dibuat dalam penelitian ini
11. Para siswa SMK Negeri Pundong Bantul yang telah membantu penulis dalam pemasangan modul pada AC lab komputer Agape lantai 2 UKDW
12. Terkhusus kepada teman saya yang bernama Arthur Heruan Adipradana yang telah membantu penulis dalam melakukan *maintanance* sistem yang telah dibuat
13. Gembala sidang, pengurus, dan jemaat Gereja Injili Agape Sola Fide Surakarta yang senantiasa memberikan dukungan dan motivasi kepada penulis dalam melakukan penelitian ini
14. Para sahabat penulis yang bernama Stanley, Rendi, Meiske, dan Mercyella yang juga memberikan semangat, dukungan, serta motivasi kepada penulis dalam melakukan penelitian ini
15. Teman-teman mahasiswa Informatika UKDW angkatan 2014 atas dukungan dan kebersamaannya selama perkuliahan
16. Semua pihak yang tidak dapat disebutkan satu persatu yang telah membantu dan memberikan dukungan kepada penulis dalam melakukan penelitian ini.

Akhir kata, penulis menyadari bahwa skripsi ini masih banyak kekurangan dikarenakan terbatasnya pengetahuan dan pengalaman yang dimiliki penulis. Oleh karena itu, penulis mengharapkan saran dan masukan, bahkan kritik yang membangun dari berbagai pihak. Semoga penelitian ini dapat bermanfaat untuk menambah pengetahuan para pembaca.

DAFTAR ISI

HALAMAN JUDUL.....	i
PERNYATAAN KEASLIAN SKRIPSI.....	iii
HALAMAN PERSETUJUAN.....	iv
HALAMAN PENGESAHAN.....	v
UCAPAN TERIMA KASIH.....	vi
INTISARI.....	viii
DAFTAR ISI.....	ix
DAFTAR GAMBAR	xi
DAFTAR TABEL.....	xv
BAB 1 PENDAHULUAN	1
1.1 Latar belakang masalah	1
1.2 Rumusan masalah.....	2
1.3 Batasan masalah	3
1.4 Tujuan penelitian	3
1.5 Metodologi penelitian.....	3
1.6 Sistematika penulisan	7
Bab 2 Tinjauan Pustaka dan Landasan Teori.....	9
2.1 Tinjauan Pustaka	9
2.1.1 Penelitian 1	9
2.1.2 Penelitian 2	12
2.1.3 Penelitian 3	13
2.1.4 Perbedaan penelitian 1, 2, 3 dengan penelitian yang akan dilakukan ...	15
2.2 Landasan Teori	15
2.2.1 <i>Internet of Things</i>	15
2.2.2 Sistem Kontrol Otomatis	16
2.2.3 <i>Microcontroller</i>	16
2.2.3.1 NodeMCU ESP8266	16
2.2.4 Protokol MQTT	18
2.2.4.1 <i>Publish dan Subscribe</i>	18

2.2.4.2 <i>Broker</i>	19
2.2.4.3 <i>Topic/Subscriptions</i>	19
2.2.5 <i>Remote Control</i> Infra Merah.....	21
BAB 3 ANALISIS DAN PERANCANGAN SISTEM	24
3.1 Kebutuhan Sistem.....	24
3.1.1 Kebutuhan Perangkat Keras / Hardware	24
3.1.2 Kebutuhan Perangkat Lunak / Software	24
3.2 Topologi Sistem.....	25
3.3 Rancangan Aplikasi pada <i>Client Publisher</i>	26
3.3.1 Rancangan Antarmuka Aplikasi	26
3.3.2 Rancangan Basis Data	40
3.4 Perancangan Protokol MQTT.....	41
3.5 Perancangan Modul Infra Merah.....	43
3.6 Perancangan <i>Use Case</i>	45
3.7 <i>Activity Diagram</i>	68
3.7.1 <i>Activity Diagram</i> Sistem Kontrol Manual pada AC.....	68
3.7.2 <i>Activity Diagram</i> Sistem Penjadwalan AC.....	69
3.8 Rancangan Pengujian Sistem	70
BAB 4 HASIL DAN PEMBAHASAN.....	71
4.1 Hasil Implementasi Aplikasi untuk <i>Client Publisher</i>	71
4.2 Hasil Perakitan IR Modul	94
4.3 Hasil Pemasangan IR Modul pada AC	95
4.4 Hasil Pengujian Sistem.....	96
4.4.1 Hasil Pengujian Sistem Kontrol Manual	96
4.4.2 Hasil Pengujian Sistem Penjadwalan	100
BAB 5 KESIMPULAN DAN SARAN	103
5.1 Kesimpulan.....	103
5.2 Saran	104
DAFTAR PUSTAKA	105

DAFTAR GAMBAR

Gambar 2.1 Skema IR WiFi <i>module</i>	10
Gambar 2.2 Alur kerja sistem yang dibangun.....	10
Gambar 2.3 Kurva perubahan suhu pada selang waktu tertentu	12
Gambar 2.4 Halaman yang berisikan panel dengan berbagai tombol perintah.....	14
Gambar 2.5 Pin Mapping ESP8266	17
Gambar 2.6 Block Diagram ESP8266	17
Gambar 2.7 Diagram protokol MQTT.....	18
Gambar 2.8 Transmisi sinyal infra merah.....	22
Gambar 3.1 Topologi sistem	25
Gambar 3.2 Rancangan Antarmuka <i>Form</i> Beranda.....	27
Gambar 3.3 Daftar menu item pada menu Ruang.....	27
Gambar 3.4 Daftar menu item pada menu AC.....	28
Gambar 3.5 Daftar menu item pada menu Jadwal	28
Gambar 3.6 Rancangan Antarmuka <i>Form</i> Tambah Data Ruang	28
Gambar 3.7 Rancangan Antarmuka <i>Form</i> Daftar Ruang.....	29
Gambar 3.8 Rancangan Antarmuka <i>Form</i> Edit Nama Ruang.....	30
Gambar 3.9 Rancangan Antarmuka <i>Form</i> Tambah Sub Ruang	30
Gambar 3.10 Rancangan Antarmuka <i>Form</i> Daftar Sub Ruang	31
Gambar 3.11 Rancangan Antarmuka <i>Form</i> Edit Nama Sub Ruang	32
Gambar 3.12 Rancangan Antarmuka <i>Form</i> Tambah Data AC	32
Gambar 3.13 Rancangan Antarmuka <i>Form</i> Daftar AC	33
Gambar 3.14 Rancangan Antarmuka <i>Form</i> Edit Data AC	34
Gambar 3.15 Rancangan Antarmuka <i>Form</i> Panel Kontrol AC	34
Gambar 3.16 Rancangan Antarmuka <i>Form</i> Tambah Jadwal	35
Gambar 3.17 Rancangan Antarmuka <i>Form</i> Daftar Jadwal.....	37
Gambar 3.18 Rancangan Antarmuka <i>Form</i> Edit Jadwal Mingguan	38
Gambar 3.19 Rancangan Antarmuka <i>Form</i> Edit Jadwal Bulanan	39
Gambar 3.20 Rancangan Antarmuka <i>Form</i> Edit Jadwal Sesuai Tanggal.....	39
Gambar 3.21 <i>Entitiy Relationship Diagram</i> Sistem.....	40

Gambar 3.22 <i>Screen capture dashboard shiftr.io</i>	43
Gambar 3.23 Skema modul infra merah	44
Gambar 3.24 Layout PCB tampak atas	44
Gambar 3.25 Layout PCB tampak bawah.....	44
Gambar 3.26 <i>Use Case Diagram</i> Sistem	67
Gambar 3.27 <i>Activity Diagram</i> Sistem Kontrol Manual pada AC.....	68
Gambar 3.28 <i>Activity Diagram</i> Sistem Penjadwalan AC	69
Gambar 4.1 <i>Form</i> beranda aplikasi	71
Gambar 4.2 <i>Form</i> Tambah Data Ruang.....	72
Gambar 4.3 Pesan <i>error</i> jika nama ruang tidak diisi	72
Gambar 4.4 Pesan <i>error</i> yang muncul jika nama ruang yang akan ditambahkan sudah ada di <i>database</i>	73
Gambar 4.5 <i>Form</i> Daftar Ruang	73
Gambar 4.6 <i>Form</i> Edit Data Ruang	74
Gambar 4.7 Dialog Konfirmasi Hapus Data Ruang.....	74
Gambar 4.8 Form Tambah Sub Ruang	75
Gambar 4.9 Pesan error yang muncul jika nama sub ruang tidak diisi.....	75
Gambar 4.10 Pesan error yang muncul jika ruang belum dipilih	75
Gambar 4.11 Pesan error yang muncul jika nama sub ruang yang akan ditambahkan sudah ada di <i>database</i>	76
Gambar 4.12 Form Daftar Sub Ruang	76
Gambar 4.13 <i>Form</i> Edit Nama Sub Ruang.....	77
Gambar 4.14 Dialog Konfirmasi Hapus Sub Ruang.....	77
Gambar 4.15 Form Tambah Data AC.....	78
Gambar 4.16 Pesan error yang muncul jika ruang belum dipilih	78
Gambar 4.17 Pesan error yang muncul jika topic MQTT dikosongi.....	79
Gambar 4.18 Pesan error yang muncul jika nomor AC tersebut pada ruang yang dipilih sudah ada di <i>database</i>	79
Gambar 4.19 Pesan error yang muncul jika suhu minimum di bawah 16 atau di atas 20.....	79

Gambar 4.20 Pesan error yang muncul jika suhu maksimum di bawah 25 atau di atas 30.....	80
Gambar 4.21 Form Daftar AC	80
Gambar 4.22 Form Edit Data AC	81
Gambar 4.23 Dialog Konfirmasi Hapus Data AC	82
Gambar 4.24 Panel Kontrol AC	82
Gambar 4.25 Pesan error yang muncul jika ruang dan nomor AC belum dipilih.	83
Gambar 4.26 Pesan error yang muncul jika suhu setelan sudah berada pada nilai minimum	83
Gambar 4.27 Pesan error yang muncul jika suhu setelan sudah berada pada nilai maksimum.....	84
Gambar 4.28 Pesan error yang muncul jika koneksi ke broker terputus	84
Gambar 4.29 Pesan error yang muncul jika sistem gagal menghubungkan ke broker	85
Gambar 4.30 Form Tambah Jadwal	85
Gambar 4.31 Pesan <i>error</i> yang muncul jika pengulangan belum dipilih	86
Gambar 4.32 Pesan error yang muncul jika ruang belum dipilih	87
Gambar 4.33 Pesan error yang muncul jika pengguna memilih pengulangan mingguan, namun belum ada hari yang dipilih.....	87
Gambar 4.34 Pesan error yang muncul jika pengguna memilih pengulangan bulanan, namun tiap tanggal belum dipilih	87
Gambar 4.35 Pesan <i>error</i> yang muncul jika pengguna memilih pengulangan sesuai tanggal, namun tanggal yang dipilih adalah sebelum tanggal hari ini	88
Gambar 4.36 Pesan <i>error</i> yang muncul jika sampai tanggal berada sebelum dari tanggal.....	88
Gambar 4.37 Pesan error yang muncul jika jam berada di antara 00.00 sampai 04.59.....	89
Gambar 4.38 Pesan error yang muncul jika jenis kontrol belum dipilih	89
Gambar 4.39 Pesan error yang muncul jika suhu setelan di bawah 16 atau di atas 27	89

Gambar 4.40 Pesan <i>error</i> yang muncul jika jadwal yang akan ditambahkan bertabrakan dengan jadwal yang ada di <i>database</i>	90
Gambar 4.41 <i>Form</i> Daftar Jadwal.....	91
Gambar 4.42 <i>Form</i> Edit Jadwal Mingguan.....	92
Gambar 4.43 <i>Form</i> Edit Jadwal Bulanan.....	92
Gambar 4.44 <i>Form</i> Edit Jadwal Sesuai Tanggal.....	93
Gambar 4.45 Dialog konfirmasi hapus jadwal.....	93
Gambar 4.46 IR modul tampak atas.....	94
Gambar 4.47 IR modul tampak bawah	94
Gambar 4.48 IR modul terpasang pada AC dengan kondisi belum terhubung ke broker	95
Gambar 4.49 IR modul terpasang pada AC dengan kondisi telah terhubung ke <i>broker</i>	96

© UKD W

DAFTAR TABEL

Tabel 2.1 Hasil pengukuran <i>process time</i> NodeMCU	11
Tabel 2.2. Data <i>error</i> pada $T_{\text{cut-off}}$ dan $T_{\text{cut-in}}$ untuk <i>differential value</i> 2°C.....	13
Tabel 3.1: UC01 – Tambah Data Ruang.....	45
Tabel 3.2: UC02 – Lihat Daftar Ruang.....	45
Tabel 3.3: UC03 – Edit Nama Ruang	46
Tabel 3.4: UC04 – Hapus Data Ruang.....	46
Tabel 3.5: UC05 – Tambah Sub Ruang	47
Tabel 3.6: UC06 - Melihat daftar sub ruang yang berada di suatu ruang tertentu	48
Tabel 3.7: UC07 – Edit Nama Sub Ruang	49
Tabel 3.8: UC08 – Hapus Data Sub Ruang.....	49
Tabel 3.9: UC09 – Tambah Data AC.....	50
Tabel 3.10: UC10 – Lihat Daftar Seluruh AC	52
Tabel 3.11: UC11 – Lihat Daftar AC untuk Ruang dan Sub Ruang Tertentu	52
Tabel 3.12: UC12 – Edit Data AC.....	53
Tabel 3.13: UC13 – Hapus Data AC.....	54
Tabel 3.14: UC14 – Menghidupkan AC	54
Tabel 3.15: UC15 – Mematikan AC	56
Tabel 3.16: UC16 – Mengubah suhu setelan pada AC	58
Tabel 3.17: UC17 – Mengubah kecepatan kipas pada AC	60
Tabel 3.18: UC18 – Tambah Jadwal.....	61
Tabel 3.19: UC19 – Lihat Jadwal	64
Tabel 3.20: UC20 – Edit Jadwal	64
Tabel 3.21: UC21 – Hapus Jadwal.....	66
Tabel 4.1 Hasil Pengujian Sistem Kontrol Manual.....	96
Tabel 4.2 Hasil Pengujian Sistem Penjadwalan.....	100

BAB 1

PENDAHULUAN

1.1. Latar belakang masalah

Kenyamanan dalam kegiatan belajar mengajar di kampus merupakan hal yang sangat penting bagi mahasiswa. Hal tersebut sangat memengaruhi konsentrasi mahasiswa dalam mengikuti perkuliahan. Oleh karena itu, keberadaan AC (*Air Conditioner*) di setiap ruang kelas maupun laboratorium memiliki peranan yang sangat penting agar mahasiswa nyaman dan tidak merasa kepanasan saat menjalani perkuliahan. Sehingga mahasiswa dapat berkonsentrasi dan fokus dalam kegiatan belajar mengajar di kampus.

Saat ini setiap ruang di Universitas Kristen Duta Wacana sudah terpasang AC. Seharusnya, seluruh AC di setiap ruang Universitas Kristen Duta Wacana hanya dioperasikan/dihidupkan ketika ruang tersebut digunakan saja. Akan tetapi, kenyataannya AC di setiap ruang kelas UKDW dihidupkan pada pagi hari dan dimatikan pada malam hari sesuai sesi terakhir perkuliahan. Padahal, tidak semua ruang kelas digunakan pada setiap sesi. Ada sesi-sesi di mana beberapa ruang kelas tidak digunakan atau kosong, sedangkan AC di ruang kelas yang kosong tersebut tetap menyala terus. Di sini jelas terjadi pemborosan listrik.

Selain itu, petugas *cleaning service* harus berkeliling ke ruang-ruang di seluruh kampus hanya untuk menghidupkan dan mematikan AC setiap harinya. Hal tersebut sangatlah tidak efektif karena memakan cukup banyak waktu dan energi, mengingat jumlah ruang di Universitas Kristen Duta Wacana yang cukup banyak. Untuk menghidupkan dan mematikan AC saja memerlukan banyak petugas yang harus berkeliling kampus setiap harinya.

Oleh karena itu, akan lebih efektif, efisien, dan hemat listrik jika dilakukan implementasi sistem kendali terpusat serta penjadwalan untuk AC di setiap ruang Universitas Kristen Duta Wacana. Dengan diimplementasikannya sistem ini, pemborosan listrik dapat diminimalisir, karena AC hanya akan beroperasi sesuai jadwal yang telah diatur. Dengan kata lain, AC akan dimatikan secara otomatis jika

ruang tidak digunakan. Selain itu, juga sudah tidak memerlukan banyak petugas yang harus berkeliling kampus setiap hari untuk menghidupkan AC di pagi hari dan mematikan AC pada malam hari, karena sistem ini yang akan melakukan pekerjaan tersebut secara otomatis sesuai jadwal.

Semua AC yang ada di Universitas Kristen Duta Wacana sudah dilengkapi dengan sensor penangkap infra merah, sehingga dapat dikontrol menggunakan *remote control* berbasis infra merah. Dengan teknologi *microcontroller* yang ada saat ini, konsep dan cara kerja *remote control* tersebut dapat dengan mudah ditiru dan digantikan fungsinya oleh *microcontroller* seperti Arduino, Node MCU, Raspberry Pi, dan lain-lain. *Microcontroller* juga dapat terhubung satu sama lain melalui jaringan komputer, baik menggunakan kabel ataupun *wireless*. Dengan konsep tersebut, sistem kendali terpusat dan penjadwalan AC dapat dibuat dengan memanfaatkan rangkaian *microcontroller* sebagai *remote control* yang dapat mengontrol AC secara otomatis.

1.2. Rumusan masalah

Berdasarkan latar belakang yang telah dijabarkan sebelumnya, masalah yang menjadi perhatian pada penelitian ini adalah operasional AC di seluruh ruang kelas Universitas Kristen Duta Wacana yang dihidupkan pada pagi hari dan dimatikan pada malam hari sesuai sesi terakhir, sehingga terjadi pemborosan listrik ketika ruang kelas tidak digunakan namun AC tetap menyala. Selain itu, juga menghabiskan cukup banyak waktu dan energi karena membutuhkan beberapa petugas yang harus menghidupkan dan mematikan AC di seluruh ruang kelas setiap harinya. Oleh karena itu, masalah dapat dirumuskan menjadi:

1. Bagaimana merancang sistem kendali terpusat untuk AC di Universitas Kristen Duta Wacana menggunakan protokol MQTT?
2. Bagaimana mengimplementasikan sistem penjadwalan AC berdasarkan jadwal pemakaian ruang di Universitas Kristen Duta Wacana?

1.3. Batasan masalah

Batasan masalah pada penelitian ini adalah sebagai berikut:

1. AC yang dapat dikontrol adalah AC yang memiliki sensor infra merah
2. Merek AC yang dapat dikontrol adalah Panasonic dan TCL
3. Kontrol yang dapat dilakukan adalah menghidupkan, mematikan, mengubah suhu, dan kecepatan kipas pada AC
4. *Microcontroller* yang digunakan adalah NodeMCU ESP8266
5. *Microcontroller* yang digunakan harus terhubung ke internet
6. Protokol yang digunakan untuk mengirim pesan antar *microcontroller* adalah MQTT
7. Sistem belum memiliki *error handling* jika terdapat interferensi dari luar sistem. Misalnya: AC dihidupkan secara manual menggunakan *remote control* konvensional ketika ruang tidak digunakan, AC tidak mendapat aliran listrik, dan lain-lain

1.4. Tujuan penelitian

Tujuan yang ingin dicapai pada penelitian ini adalah mengimplementasikan sistem kendali terpusat dan penjadwalan terhadap AC di Universitas Kristen Duta Wacana menggunakan protokol MQTT.

1.5. Metodologi penelitian

Berikut adalah metode-metode penelitian yang akan dilakukan:

1. Tahap Studi Literatur

Pada tahap ini, penulis akan mengumpulkan beberapa artikel dan literatur yang berisi teori atau penelitian lain yang membahas tentang *Internet of Things*, protokol MQTT, kontrol AC menggunakan *microcontroller*, dan penjadwalan dalam sistem IoT yang dapat membantu penulis dalam melakukan penelitian ini.

2. Tahap Pengumpulan Data

Pada tahap ini, penulis akan melakukan pengumpulan data yang berkaitan dengan penelitian ini, seperti daftar merek AC yang digunakan di UKDW, data-

data sinyal yang digunakan untuk mengontrol AC, dan jadwal pemakaian ruang kelas di UKDW.

Untuk dapat mengetahui merek-merek AC yang digunakan di UKDW, penulis akan melakukan survei ke tiap-tiap ruang yang ada di UKDW.

Untuk mendapatkan data-data sinyal untuk setiap merek AC yang digunakan di UKDW, penulis akan meminjam *remote control* asli untuk setiap merek AC yang ada, kemudian merecord sinyal dari masing-masing *remote control* asli tersebut menggunakan *IR library* (*library* infra merah untuk Arduino). Perangkat keras yang dibutuhkan untuk merecord sinyal adalah 1 buah *microcontroller* yang terhubung ke *receiver* infra merah. Record sinyal dilakukan dengan mengupload program yang digunakan untuk merecord sinyal ke suatu *microcontroller*, misalnya Arduino Uno, NodeMCU ESP8266, atau yang lainnya, kemudian menekan tombol pada *remote control* dengan posisi *remote control* diarahkan ke *receiver* infra merah. Ketika suatu tombol pada *remote control* ditekan, maka *remote control* akan mengeluarkan sinyal infra merah. Data sinyal infra merah inilah yang akan direcord menggunakan *IR library* Arduino. Sinyal yang akan direcord adalah sinyal untuk menghidupkan dan mematikan AC, sinyal untuk setiap level temperatur yang didukung oleh masing-masing AC (misalnya dari 16 sampai 30 derajat celcius), dan sinyal untuk setiap level kecepatan kipas pada AC (*low*, *mid*, *high*, dan *auto*). Data-data sinyal ini nantinya akan digunakan untuk mengontrol AC menggunakan *microcontroller*.

Sedangkan untuk dapat memperoleh jadwal pemakaian ruang kelas di UKDW, penulis akan meminta ke Biro 1 (Biro Administrasi Akademik).

3. Tahap Perancangan dan Pengembangan Sistem

Sistem akan dibangun menggunakan 1 *client publisher* dan beberapa *client subscriber*. *Client publisher* berupa 1 perangkat komputer yang akan dipasang suatu aplikasi yang dibuat oleh penulis, sedangkan *client subscriber* berupa *microcontroller* NodeMCU ESP8266 yang terhubung ke *transmitter* infra merah.

Pada penelitian ini, penulis akan membuat sebuah aplikasi *desktop* dengan bahasa pemrograman Visual Basic .NET dan *database server* MySQL. Aplikasi tersebut nantinya akan dipasangkan pada *client publisher*.

Dalam aplikasi tersebut akan terdapat panel untuk menghidupkan AC, mematikan AC, mengubah temperatur AC, dan mengubah kecepatan kipas pada AC secara manual, sehingga pengguna dapat melakukan kontrol manual pada AC di seluruh ruang kelas Universitas Kristen Duta Wacana melalui aplikasi tersebut. Jika aplikasi tersebut menerima suatu perintah dari pengguna, misalnya perintah untuk menghidupkan AC di ruang B.3.1, maka aplikasi tersebut akan mengirimkan/mempublish *message* (berupa perintah untuk menghidupkan AC) ke *broker* pada *topic* yang sesuai untuk ruang tersebut.

Selain itu, pada aplikasi tersebut juga akan disediakan suatu *form* untuk memasukkan jadwal hidup atau matinya AC di suatu ruang kelas. Sehingga, aplikasi tersebut juga berperan untuk melakukan penjadwalan AC di seluruh ruang kelas berdasarkan jadwal yang terdapat pada *database*. Misalnya sesuai jadwal ruang D.1.1 selesai digunakan pada pukul 10.20, maka pada jam itu juga aplikasi pada *client publisher* tersebut akan mengirimkan *message* (berupa perintah untuk mematikan AC) ke *broker* pada *topic* yang sesuai untuk ruang tersebut.

Pada penelitian ini, penulis juga akan membuat program dengan Arduino IDE yang nantinya akan diupload ke seluruh NodeMCU ESP8266 yang berperan sebagai *client subscriber*. Program yang dibuat untuk *client subscriber* ini berfungsi untuk melakukan *subscribe* ke suatu *topic* sesuai dengan ruang kelas ditematkannya *client subscriber* tersebut, menerima *message* yang dikirimkan oleh *client publisher*, dan memerintahkan transmitter infra merah untuk mengeluarkan sinyal infra merah sesuai perintah pada *message* yang diterimanya. Sebagai contoh, suatu *client subscriber* akan ditempatkan di Gedung Didaktos ruang D.1.1, maka *client subscriber* tersebut akan diprogram untuk melakukan *subscribe* ke *topic* Didaktos/D.1.1. *Client subscriber* di ruang D.1.1 tersebut akan menerima *message* yang dikirimkan oleh aplikasi yang terpasang di *client publisher*. Kemudian, *client subscriber* akan memproses *message* yang diterima tersebut dan akan mengontrol AC (dengan mengeluarkan sinyal infra merah) sesuai perintah yang terdapat pada *message* tersebut. Misalnya, jika *message* yang diterima adalah perintah untuk mematikan AC, maka *client subscriber* akan

mengeluarkan sinyal infra merah untuk mematikan AC tersebut. Nantinya, pada setiap AC akan dipasang 1 *client subscriber* untuk mengontrol AC tersebut.

4. Tahap Pengujian dan Evaluasi Sistem

Pada tahap ini, sistem akan diuji dan dievaluasi untuk pengembangan selanjutnya.

Aspek yang akan dievaluasi antara lain:

a. *Error rate* kontrol manual

Akan dilakukan pengukuran *error rate* terhadap kontrol manual. Misalnya, pengguna mengubah temperatur AC di ruang D.1.1 menjadi 23°C melalui aplikasi yang ada di komputer *client publisher*. Kemudian akan dicek, apakah temperatur AC di ruang D.1.1 benar berubah menjadi 23°C atau tidak. Pengecekan ini akan dilakukan sebanyak 10 kali dan akan dicatat *error rate* nya.

b. *Error rate* penjadwalan

Akan dilakukan pengukuran *error rate* terhadap sistem penjadwalan. Misalnya, sesuai jadwal ruang H.1.1 mulai digunakan pada jam 07.30, maka pada jam tersebut akan dicek apakah AC di ruang H.1.1 benar akan menyala atau tidak. Pengecekan ini akan dilakukan terhadap 10 jadwal ruang yang berbeda-beda dan dicatat *error rate* nya.

c. Waktu delay rata-rata

Akan dilakukan pengukuran waktu *delay* terhadap kontrol manual yang dilakukan pengguna. Misalnya, seorang pengguna mengklik *button Power On* pada panel AC di ruang D.1.2 yang ada di aplikasi yang dibuat oleh penulis. Maka pada saat itu juga ada orang yang melakukan pengecekan di ruang D.1.2 apakah AC di ruang tersebut langsung menyala. Pada saat yang bersamaan, orang yang berada di ruang tersebut akan mengaktifkan *stopwatch* untuk mengukur *delay* yang terjadi antara pengguna mengklik *button Power On* pada aplikasi hingga AC menyala. Pengukuran ini dilakukan terhadap 10 AC di ruang yang berbeda-beda dengan jenis kontrol yang bervariasi, yaitu menghidupkan, mematikan, mengubah temperatur, dan mengubah kecepatan kipas pada AC. Kemudian akan dihitung waktu delay rata-rata terhadap pengujian kontrol manual yang dilakukan. Setelah itu akan dilakukan evaluasi terhadap waktu delay rata-rata tersebut.

Selain itu, akan dilakukan pengukuran waktu *delay* untuk penjadwalan yang telah diatur. Misalnya sesuai jadwal, ruang B.3.1 selesai digunakan pada pukul 16.20, maka pada pukul 16.20 tepat akan dicek apakah AC di ruang B.3.1 akan mati. Jika belum mati, akan ditunggu beberapa saat sampai AC di ruang tersebut mati, kemudian dihitung waktu *delay* nya. Namun sebelumnya, jam yang akan digunakan sebagai acuan untuk pengukuran delay ini akan disesuaikan dengan waktu pada komputer *client publisher* terlebih dahulu, agar pengukuran menjadi lebih akurat. Pengukuran ini dilakukan untuk 10 jadwal ruang. Kemudian akan dihitung waktu delay rata-rata untuk sistem penjadwalan yang telah dibuat. Setelah itu akan dilakukan evaluasi terhadap waktu delay rata-rata yang telah diuji tersebut.

5. Tahap Penarikan Kesimpulan

Dari data yang didapat pada tahap evaluasi akan ditarik kesimpulan performa penerapan sistem. Kesimpulan ini dapat digunakan sebagai acuan untuk perbaikan sistem pada penelitian selanjutnya.

1.6. Sistematika Penulisan

Laporan skripsi ini terdiri dari 5 bab, yaitu Bab 1 Pendahuluan, Bab 2 Tinjauan Pustaka dan Landasan Teori, Bab 3 Analisis dan Perancangan Sistem, Bab 4 Hasil dan Pembahasan, dan Bab 5 Kesimpulan dan Saran.

Bab 1 berisi tentang pendahuluan dari penelitian yang akan dilakukan. Bab ini meliputi latar belakang penelitian, rumusan masalah, batasan masalah, tujuan penelitian, metodologi penelitian, dan sistematika penulisan laporan skripsi ini. Bab ini bertujuan untuk memberikan informasi mengenai latar belakang masalah yang terjadi, apa yang akan dilakukan, dan tujuan dilakukannya penelitian ini.

Bab 2 berisi tentang tinjauan pustaka dan landasan teori yang diperlukan untuk melakukan penelitian ini. Tinjauan pustaka berisi uraian mengenai penelitian yang serupa dengan penelitian yang akan dilakukan. Tinjauan pustaka diambil dari beberapa jurnal dan artikel ilmiah yang telah dipublikasikan. Sedangkan landasan teori berisi teori-teori yang berhubungan dengan penelitian ini. Teori-teori ini dapat berupa definisi, konsep dasar dan cara kerja suatu sistem, atau algoritma tertentu.

Bab 3 berisi tentang perancangan sistem yang akan dibangun pada penelitian ini. Perancangan sistem meliputi kebutuhan fungsional dan non fungsional, alur kerja sistem, arsitektur sistem, struktur data yang akan digunakan, rancangan antarmuka sistem yang akan dibuat, dan rancangan pengujian sistem.

Bab 4 berisi tentang hasil penelitian, pembahasan / analisis dari penelitian yang dilakukan, dan hasil pengujian sistem. Pembahasan tentang hasil yang diperoleh berupa penjelasan secara kuantitatif, kualitatif, atau secara statistik. Hasil pengujian dijabarkan sesuai dengan rancangan pengujian yang ditulis pada bab 3.

Bab 5 berisi kesimpulan dari penelitian yang telah dilakukan dan saran-saran untuk penelitian selanjutnya. Kesimpulan merupakan pernyataan singkat dan tepat yang dijabarkan dari hasil analisis kegiatan penelitian. Sedangkan saran memuat langkah-langkah, metode, dan teknik pengembangan yang belum dilakukan dalam penelitian ini, tetapi dirasa akan memperbaiki kinerja sistem jika langkah-langkah tersebut dilaksanakan pada penelitian selanjutnya.

© UKD W

BAB 5

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Dari hasil penelitian yang dilakukan, dapat disimpulkan bahwa sistem kontrol dan penjadwalan AC yang dibuat dalam penelitian ini berhasil diimplementasikan di lab komputer Agape lantai 2 UKDW dan dapat bekerja dengan baik dengan performa yang cukup memuaskan. Sistem kontrol manual yang dibuat memiliki tingkat keberhasilan yang cukup tinggi, yaitu sebesar 97,5% dengan waktu *delay* rata-rata yang relatif singkat, yaitu 3,1 detik. Sedangkan sistem penjadwalan yang dibuat memiliki tingkat keberhasilan yang cukup tinggi juga, yaitu sebesar 90% dan waktu *delay* rata-rata yang masih tergolong relatif singkat, yaitu 12,36 detik (tidak sampai setengah menit). Dengan demikian, dapat disimpulkan bahwa protokol MQTT adalah protokol yang handal untuk mengirim pesan dari PC ke *microcontroller* atau sebaliknya, terbukti sistem yang dibuat dapat bekerja dengan baik.

Dengan diimplementasikannya sistem penjadwalan AC tersebut, petugas lab komputer tidak perlu lagi menghidupkan dan mematikan AC secara manual satu per satu setiap harinya, karena sistem yang akan melakukannya secara otomatis sesuai jadwal yang telah diinputkan. Dengan demikian dapat menghemat waktu dan energi. Selain itu, dengan diimplementasikannya sistem penjadwalan AC tersebut dapat meminimalisir pemborosan listrik, karena AC akan mati secara otomatis ketika ruang lab sudah tidak digunakan.

Namun, sistem yang telah dibuat masih memiliki beberapa kekurangan, antara lain:

1. Pengguna tidak dapat mengetahui status/kondisi AC secara *realtime*, apakah AC sedang dalam kondisi menyala atau mati, karena tidak ada sensor yang digunakan untuk mendeteksinya
2. Tidak ada mekanisme keamanan yang diimplementasikan pada sistem tersebut, sehingga jika ada yang mengetahui username dan password MQTT untuk sistem

ini beserta topic untuk setiap AC nya, maka AC dapat dikontrol dengan mengirimkan *message* secara manual ke topic AC yang dituju

3. Belum ada fitur log untuk sistem penjadwalan yang telah dibuat, sehingga pengguna tidak dapat mengetahui apakah penjadwalan berhasil dijalankan atau tidak
4. Menggunakan *source code* yang berbeda untuk setiap merek AC yang digunakan, sehingga ketika hendak mengupload program ke *microcontroller* harus memerhatikan merek AC nya terlebih dahulu.

5.2. Saran

Adapun saran untuk pengembangan sistem ini selanjutnya, antara lain:

1. Untuk setiap modul yang terpasang di AC ditambahkan sensor suhu, untuk mengetahui kondisi AC apakah menyala atau mati.
2. Ditambahkan fitur log untuk sistem penjadwalan yang telah dibuat. Sehingga, dapat dilihat apakah penjadwalan berhasil dijalankan atau tidak.
3. Ditambahkan autentikasi pengguna, sehingga aplikasi sistem kontrol dan penjadwalan AC hanya dapat digunakan oleh pengguna yang terdaftar pada sistem.
4. Ditambahkan mekanisme pengamanan sistem, agar sistem tidak mudah disusupi oleh pengguna yang mengetahui username dan password MQTT untuk sistem ini beserta topic untuk setiap AC nya.
5. *Source code* untuk semua merek AC yang digunakan digabung menjadi satu, sehingga bersifat universal. Data-data sinyal untuk seluruh merek AC yang ada disimpan pada EEPROM *microcontroller*. Untuk melakukan pengaturan merek AC dilakukan melalui komunikasi serial.

DAFTAR PUSTAKA

A., Junaidi. (2015). INTERNET OF THINGS, SEJARAH, TEKNOLOGI DAN PENERAPANNYA : REVIEW. *Jurnal Ilmiah Teknologi Informasi Terapan*, 1(3), 1-2.

B., Rizkydio. (2016). Rancang Bangun Sistem Backup Power dan Manometer Digital Kompresor Udara Portable Berbasis Mikrokontroler ATMEGA8535. Politeknik Negeri Sriwijaya. Diakses pada 27 Juli 2018, dari [http://eprints.polsri.ac.id/2838/3/BAB II.pdf](http://eprints.polsri.ac.id/2838/3/BAB%20II.pdf)

B., Perangin. (n.d.). Universitas Sumatera Utara. Diakses pada 27 Juli 2018, dari repository.usu.ac.id/bitstream/handle/123456789/35001/Chapter%20II.pdf?sequence=4

Listiono & Gandung. (2016). Analisis Penggerak pada Sistem Pengaman Pintu Ber-Password. Universitas Negeri Yogyakarta. Diakses pada 27 Juli 2018, dari [http://eprints.uny.ac.id/30119/3/4. BAB II.pdf](http://eprints.uny.ac.id/30119/3/4.%20BAB%20II.pdf)

T. T., Saputro. (2018). Mengenal NodeMCU: Pertemuan Pertama. Diakses pada 27 Juli 2018, dari <https://embeddednesia.com/v1/?p=2050>

A. B., Priandana. (2017). ESP8266 Primadona Baru. Diakses pada 28 Juli 2018, dari <https://blog.framework.id/esp8266-primadona-baru-iot-286f7055c3f7>

D. F., Arranda. (2017). Kontrol Lampu Ruangan Berbasis Web Menggunakan Nodemcu Esp8266. STMIK AKAKOM. Diakses pada 28 Juli 2018, dari http://eprints.akakom.ac.id/4904/3/3_143310003_BAB_II.pdf

Equan. (2015). Mengenal MQTT - Protokol IoT. Diakses pada 28 Juli 2018, dari <https://medium.com/pemrograman/mengenal-mqtt-998b6271f585>

T. T., Saputro. (2018). Mengenal MQTT, Protokol Komunikasi untuk IoT. Diakses pada 28 Juli 2018, dari <https://embeddednesia.com/v1/?p=3110>

A. P. Utama. (2015). Sistem Pengontrolan Pintu Gerbang Otomatis dengan Metode Timer Menggunakan Interface Visual Basic.Net Berbasis Arduino Uno

pada Pt.Furin Jaya. Diakses pada 1 Agustus 2018, dari <http://widuri.raharja.info/index.php/SI1233472982>

L. K. P., Saputra, & Y., Lukito. (2017). Implementation of Air Conditioning Control System Using REST Protocol Based on NodeMCU ESP8266. *International Conference on Smart Cities, Automation & Intelligent Computing Systems*, 126-130

E., Erham, M., & W. P., Sopianti. (2018). Design of a new on-off controller based on Arduino UNO R3 with application to Window A/C. Pada *The International Conference on Industrial and System Engineering* (hal. 174-182). Surabaya: Institut Teknologi Sepuluh Nopember. Diunduh pada 18 November 2018, dari <http://iptek.its.ac.id/index.php/jps/article/view/3726/0>

N. V. K., Ramesh, S. V. T., Kumar, V., Vamsi, & S., Akarsh. (2017). WI-FI CONTROLLED UNIVERSAL REMOTE USING ESP8266. *ARPN Journal of Engineering and Applied Sciences*, 12(24), 7233-7237. Diunduh pada 20 November 2018 dari http://www.arnjournals.org/jeas/research_papers/rp_2017/jeas_1217_6607.pdf

ESP8266EX Datasheet. (2018). Espressif Systems. Diunduh pada 24 November 2018 dari https://www.espressif.com/sites/default/files/documentation/0a-esp8266ex_datasheet_en.pdf