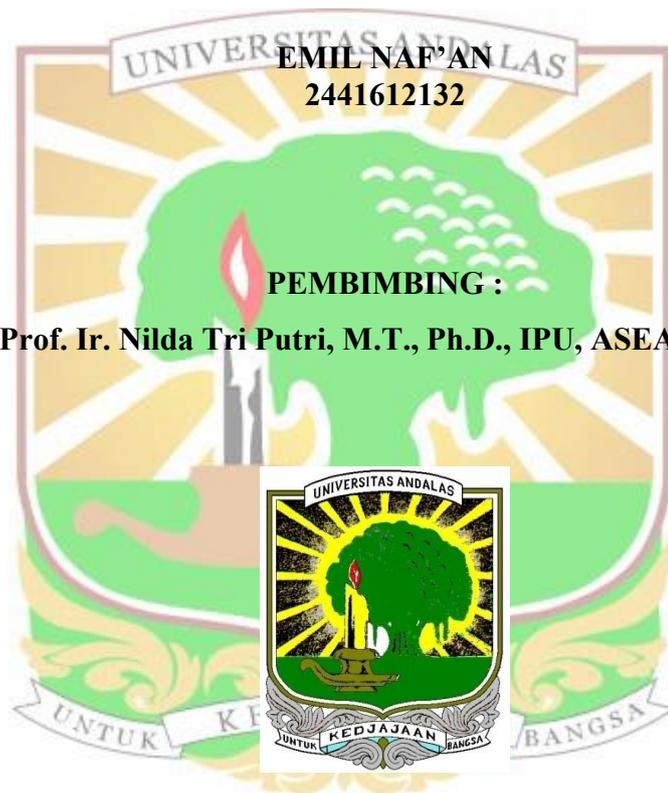


**OPTIMASI SISTEM PENJADWALAN SHOLAT DIGITAL BERBASIS
INTERNET OF THINGS (IoT) PADA MASJID RAHMATAN
LIL 'ALAMIN (RLA) UNIVERSITAS PUTRA
INDONESIA YPTK PADANG**

LAPORAN PENELITIAN



EMIL NAF'AN
2441612132

PEMBIMBING :

Prof. Ir. Nilda Tri Putri, M.T., Ph.D., IPU, ASEAN Eng.

**PROGRAM STUDI PENDIDIKAN PROFESI INSINYUR
SEKOLAH PASCASARJANA
UNIVERSITAS ANDALAS
2025**

**OPTIMASI SISTEM PENJADWALAN SHOLAT DIGITAL BERBASIS
INTERNET OF THINGS (IoT) PADA MASJID RAHMATAN
LIL 'ALAMIN (RLA) UNIVERSITAS PUTRA
INDONESIA YPTK PADANG**

**Emil Naf'an
2441612132**



**PROGRAM STUDI PENDIDIKAN PROFESI INSINYUR
SEKOLAH PASCASARJANA
UNIVERSITAS ANDALAS
2025**

HALAMAN PERSETUJUAN

Judul Laporan Akhir : OPTIMASI SISTEM PENJADWALAN
SHOLAT DIGITAL BERBASIS
INTERNET OF THINGS (IoT) PADA
MASJID RAHMATAN LIL 'ALAMIN
(RLA) UNIVERSITAS PUTRA
INDONESIA YPTK PADANG

Nama Mahasiswa : Emil Nafan

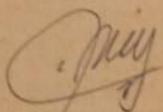
Nomor Induk Mahasiswa : 2441612132

Program Studi : PENDIDIKAN PROFESI INSINYUR

Laporan Penelitian telah diuji dan dipertahankan di depan sidang panitia ujian Profesi Insinyur pada Sekolah Pascasarjana Universitas Andalas dan dinyatakan lulus pada tanggal 12 Juni 2025.

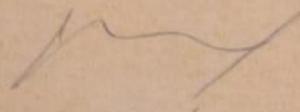
Menyetujui,

1. Pembimbing



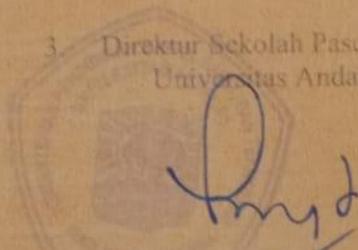
Prof. Ir. Nilda Tri Putri, M.T.,
Ph.D., IPU, ASEAN Eng.
NIP. 197707162003122003

2. Koordinator Program Studi



Ir. Benny Dwika Leonanda, M.T., IPM,
ASEAN Eng.
NIP. 196608061994121000

3. Direktur Sekolah Pascasarjana
Universitas Andalas



Prof. apt. Henny Lucida, Ph.D.
NIP. 19670115199103200

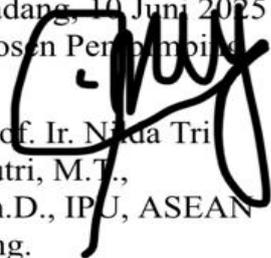
HALAMAN PERSETUJUAN PEMBIMBING SEBAGAI PERSYARATAN UJIAN INSINYUR

Judul Laporan Penelitian : **OPTIMASI SISTEM PENJADWALAN
SHOLAT DIGITAL BERBASIS INTERNET
OF THINGS (IOT) PADA MASJID
RAHMATAN LIL ‘ALAMIN (RLA)
UNIVERSITAS PUTRA INDONESIA YPTK
PADANG**

Nama Mahasiswa : EMIL NAF‘AN
Nomor Induk Mahasiswa : 2441612132
Program Studi : Pendidikan Profesi Insinyur

Laporan Penelitian ini telah diperiksa dan dinyatakan telah memenuhi untuk mengikuti Ujian Profesi Insinyur pada Program Studi Pendidikan Profesi Insinyur, Sekolah Pascasarjana, Universitas Andalas.

Padang, 10 Juni 2025
Dosen Pembimbing
-
-
Prof. Ir. Nida Tri
Putri, M.T.,
Ph.D., IPU, ASEAN
Eng.
NIP.
197707162003122003



^{*)} Pilih salah satu

SURAT PERNYATAAN KEASLIAN TUGAS LAPORAN PENELITIAN

Saya yang bertanda tangan di bawah ini:

Nama : Emil Naf'an
NIM : 2441612132
Tempat Tgl Lahir : Air Bangis, 17 Desember 1974
Alamat : Komplek Fadilla Mandiri Gg. Sarai No. 25 RT.02/RW.12,
Kel. Banuaran Nan XX, Kec. Lubuk Begalung, Kota
Padang.

Dengan ini menyatakan bahwa Tugas Laporan Penelitian dengan judul **'OPTIMASI SSTEM PENJADWALAN SHOLAT DIGITAL BERBASIS *INTERNET OF THINGS* (IoT) PADA MASJID RAHMATAN LIL 'ALAMIN (RLA) UNIVERSITAS PUTRA INDONESIA YPTK PADANG'** adalah hasil pekerjaan saya; dan seluruh ide, pendapat, atau materi dari sumber lain telah dikutip dengan cara penulisan referensi yang sesuai.

Pernyataan ini saya buat dengan sebenar-benarnya dan jika pernyataan ini tidak sesuai dengan kenyataan, maka saya bersedia menanggung sanksi yang akan dikenakan kepada saya termasuk pencabutan gelar Profesi Insinyur yang nanti saya dapatkan.

Padang, 06 Juni 2025
Yang Menyatakan



Emil Naf'an

ABSTRAK

Penelitian ini bertujuan untuk melakukan optimasi terhadap sistem jadwal sholat digital yang telah terpasang di Masjid Rahmatan Lil 'Alamin Universitas Putra Indonesia YPTK Padang. Sistem sebelumnya terdiri dari dua unit yang dikembangkan pada tahun 2016 dan 2019, masing-masing menggunakan Arduino Mega2560, RTC (DS1307), dan panel 7-segment common anoda yang dikendalikan melalui shift register 74HC595 dan driver ULN2003. Meskipun sistem ini telah berjalan dengan baik, terdapat dua kelemahan utama yang menjadi dasar perlunya optimasi, yaitu tidak adanya sinkronisasi waktu secara otomatis terhadap waktu global, serta pengaturan jeda iqomah yang masih bersifat manual dan kurang fleksibel.

Optimasi dilakukan dengan menambahkan modul ESP-32 sebagai pengendali berbasis *Internet of Things* (IoT) yang memungkinkan sistem terhubung ke jaringan WiFi. Modul ini bertugas menyinkronkan waktu secara otomatis melalui server NTP (*Network Time Protocol*), serta menyediakan antarmuka web sederhana yang dapat diakses melalui *smartphone* oleh pengurus masjid untuk mengatur jeda iqomah secara mudah dan cepat. Komunikasi antara ESP-32 dan Arduino Mega2560 dilakukan melalui protokol UART.

Hasil pengujian menunjukkan bahwa sistem baru mampu menjaga ketepatan waktu dengan deviasi kurang dari satu detik selama periode pengujian. Pengaturan jeda iqomah melalui antarmuka web dapat dilakukan dalam waktu kurang dari dua detik dengan tingkat keberhasilan 100%. Selain itu, pengurus masjid memberikan umpan balik yang positif terkait kemudahan penggunaan dan efisiensi waktu operasional. Dengan optimasi ini, sistem jadwal sholat digital menjadi lebih andal, fleksibel, dan sesuai dengan kebutuhan masjid modern yang mengutamakan efisiensi dan ketepatan waktu

Kata Kunci: Jadwal Sholat Digital, IoT, ESP-32, Arduino, Sinkronisasi Waktu, Iqomah.

ABSTRACT

This research aims to optimize the digital prayer schedule system already installed at Masjid Rahmatan Lil 'Alamin Universitas Putra Indonesia YPTK Padang. The existing system consists of two units developed in 2016 and 2019, respectively. Both systems utilize Arduino Mega2560, RTC modules (DS1307), and 6-digit common anode 7-segment displays driven via 74HC595 shift registers and ULN2003 transistor arrays. While the system functioned reliably, it had two primary limitations: the lack of automatic synchronization with global time, and the manual, inflexible configuration of iqomah delay settings.

The optimization was achieved by integrating an ESP-32 module as an Internet of Things (IoT) controller, enabling the system to connect to local WiFi networks. The ESP-32 module performs automatic time synchronization via an NTP (Network Time Protocol) server and provides a simple web-based interface accessible via smartphone. Through this interface, mosque administrators can configure iqomah delays efficiently without the need for physical interaction with the hardware. Communication between the ESP-32 and Arduino Mega2560 is established through UART serial protocol.

Testing results indicate that the optimized system maintains time accuracy with a deviation of less than one second over the testing period. Iqomah delay adjustments via the web interface are completed in under two seconds with 100% success. Furthermore, feedback from mosque staff highlights improved usability and operational efficiency. With this optimization, the digital prayer schedule system becomes more reliable, flexible, and aligned with the modern needs of mosques that prioritize punctuality and ease of operation.

Keywords: Digital Prayer Schedule, IoT, ESP-32, Arduino, Time Synchronization, Iqomah.



DAFTAR ISI

HALAMAN PERSETUJUAN.....	iii
HALAMAN PERSETUJUAN PEMBIMBING SEBAGAI PERSYARATAN UJIAN INSINYUR	iv
SURAT PERNYATAAN KEASLIAN.....	v
ABSTRAK	vi
ABSTRACT	vii
DAFTAR ISI.....	viii
DAFTAR TABEL.....	x
DAFTAR GAMBAR	xi
DAFTAR SINGKATAN	xii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	5
1.3 Batasan Masalah	6
1.4 Tujuan Penelitian.....	7
1.5 Manfaat Penelitian.....	7
1.6 Sistematika Penulisan.....	8
BAB II TINJAUAN PUSTAKA.....	10
2.1 Jadwal Sholat Digital.....	10
2.2 Arduino Mega2560.....	11
2.3 Modul ESP32.....	15
2.4 Konsep Internet of Things (IoT).....	18
2.5 Komunikasi Serial UART	19
2.6 Penelitian Terkait.....	20

BAB III METODE PENELITIAN.....	25
3.1 Metode Penelitian	25
3.2 Studi Sistem Lama.....	27
3.3 Identifikasi Permasalahan.....	28
3.4 Perancangan Sistem Optimasi	29
3.5 Implementasi dan Pengujian Sistem.....	30
BAB IV HASIL DAN PEMBAHASAN	32
4.1 Implementasi Sistem.....	32
4.2 Dokumentasi Pengujian Alat.....	32
4.3 Analisis Perbandingan Sistem	40
4.4 Evaluasi Kinerja Sistem.....	42
4.5 Pembahasan Hasil.....	43
BAB V KESIMPULAN DAN SARAN.....	45
5.1 Kesimpulan.....	45
5.2 Saran	45
DAFTAR PUSTAKA	47
LAMPIRAN.....	50



DAFTAR TABEL

Tabel 2.1 Data Teknis Board Arduino Mega 2560	10
Tabel 2.2 Perbandingan Fitur ESP32 dengan Arduino Uno, Mega dan ES8266	15
Tabel 2.3 Perbandingan Fitur dari beberapa Penelitian Terdahulu.....	19
Tabel 3.1 Jadwal Pelaksanaan Penelitian.....	29
Tabel 4.1 Perbandingan Sistem Penjadwalan Sholat	34
Tabel 4.2 Pengujian Data Jadwal Sholat	37
Tabel 4.3 Pengujian <i>Buzzer</i>	37
Tabel 4.4 Perbandingan Sistem Penjadwalan Sholat	42

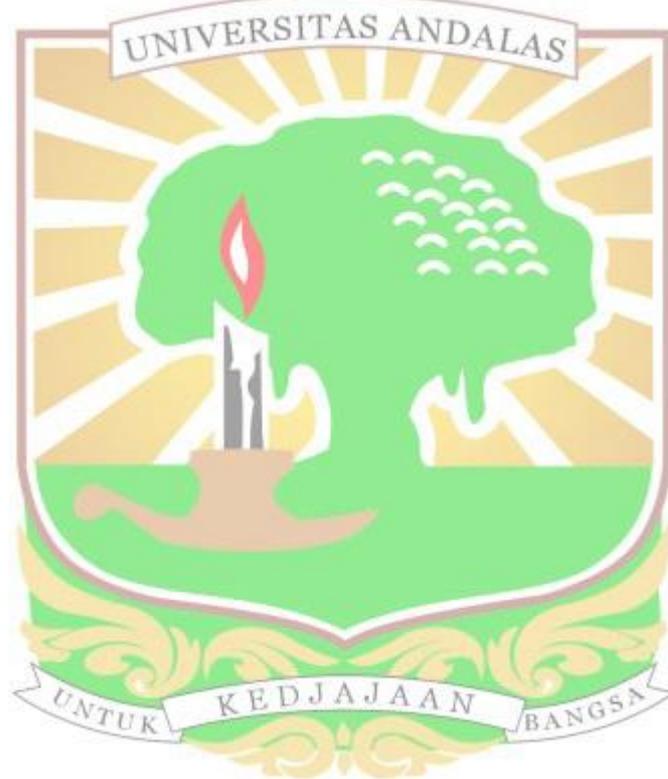


DAFTAR GAMBAR

Gambar 1.1 Jadwal Sholat Digital Masjid RLA Versi 1.0.....	1
Gambar 1.2 Jadwal Sholat Digital Masjid RLA Versi 2.0.....	2
Gambar 1.3 Jadwal Sholat Masjid RLA Versi 1.0 dan Versi 2.0	3
Gambar 1.4 Perbedaan waktu pada Jadwal Sholat Digital Masjid RLA Versi 1.0 dan Versi 2.0	4
Gambar 2.1 Tampilan Umum Jadwal Sholat Digital di Masjid.....	8
Gambar 2.2 Blok Diagram Sistem Jadwal Sholat Digital Konvensional.....	9
Gambar 2.3 Arduino Mega 2560	10
Gambar 2.4 Modul ESP32 Dev Board.....	14
Gambar 2.5 Diagram Komunikasi Serial UART antara ESP32 dan Arduino	18
Gambar 4.1 Tampilan sistem jadwal sholat tahun 2016	33
Gambar 4.2 Tampilan sistem jadwal sholat tahun 2019	33
Gambar 4.3 Pengujian Tampilan <i>Seven Segment</i> Waktu (jam, menit, detik)	35
Gambar 4.4 Pengujian Tampilan <i>Seven Segment</i> Tanggal / Hari	35
Gambar 4.5 Pengujian Tampilan <i>Seven Segment</i> Jadwal Sholat	36
Gambar 4.6 Tampilan Alamat IP Akses ESP32	38
Gambar 4.7 Tampilan Login ke Sistem	38
Gambar 4.8 Tampilan Dashboard Pengaturan Sinkronisasi Waktu NTP	39
Gambar 4.9 Tampilan Hasil Pengaturan Sinkronisasi Waktu NTP Pada Alat	39
Gambar 4.10 Tampilan Dashboard Pengaturan Sinkron Waktu Jeda Iqomah	40

DAFTAR SINGKATAN

GPS	<i>Global Positioning System</i>
IoT	<i>Internet of Things</i>
NTP	<i>Network Time Protocol</i>
RLA	Rahmaan Lil 'Alamin
RTC	<i>Real Time Clock</i>
UART	<i>Universal Asynchronous Receiver Transmitter</i>
USB	<i>Universal Serial Bus</i>



BAB I

PENDAHULUAN

1.1 Latar Belakang

Sistem jadwal sholat digital telah menjadi perangkat penting dalam mendukung aktivitas ibadah di masjid. Salah satu fungsinya adalah menampilkan waktu sholat dengan akurat dan konsisten berdasarkan lokasi tertentu. Di Masjid Rahmatan Lil 'Alamin (RLA) Universitas Putra Indonesia YPTK Padang, telah terpasang dua unit alat jadwal sholat digital, masing-masing dibuat dan dipasang pada tahun 2016 dan 2019. Kedua alat tersebut telah beroperasi selama bertahun-tahun dan secara umum telah memberikan manfaat yang signifikan bagi jamaah masjid. Gambar 1.1 merupakan dokumentasi alat jadwal sholat yang pertama dipasang (Versi 1.0), yaitu pada tahun 2016. [1]



Gambar 1.1 Jadwal Sholat Digital Masjid RLA Versi 1.0

Waktu pemasangan alat ini dapat dilihat pada tanggal yang tertera pada alat jadwal sholat digital yang tertera pada alat jadwal sholat digital tersebut, yaitu tanggal 21 November 2016. Pada tahun 2016 tersebut masjid masih berukuran relatif kecil, sehingga hanya memerlukan satu buah jadwal sholat saja. Masjid ini terletak di dalam lokasi kampus bagian belakang, berdekatan dengan gedung A

Kampus UPI YPTK Padang. Masjid ini hanya dimanfaatkan oleh kalangan civitas akademika saja dan jarang dimanfaatkan oleh masyarakat umum. Ini disebabkan letaknya yang relatif jauh dari jalan raya, sehingga relatif sulit untuk diakses oleh masyarakat umum.

Seiring dengan perkembangan kampus Universitas Putra Indonesia, pada tahun 2019, didirikan Masjid yang baru dengan ukuran yang lebih besar dari sebelumnya. Masjid yang lama berubah fungsi menjadi tempat mahasiswa belajar mata kuliah agama. Masjid baru ini masih menggunakan nama sebelumnya. Masjid ini terletak masih di bagian depan kampus, di Jalan Raya Lubuk Begalung. Ini tentu saja memudahkan akses masyarakat luas, ditambah dengan sarana parkir yang memadai.

Sejalan dengan pemindahan masjid tersebut, maka jadwal sholat juga ikut berpindah ke lokasi yang baru. Posisi peletakan berada di sisi kiri mihrab masjid. Pihak Yayasan Perguruan Tinggi Komputer Padang (YPTK) minta dibuatkan lagi Jadwal sholat yang baru yang dilengkapi dengan *Running Text* dalam ukuran yang relatif besar, yaitu 1.3 x 3.5 meter. Gambar 1.2 merupakan dokumentasi alat jadwal sholat yang kedua (Versi 2.0), yaitu pada tahun 2019 [2].



Gambar 1.2 Jadwal Sholat Digital Masjid RLA Versi 2.0

Namun, berdasarkan pengalaman langsung dalam pengelolaan alat tersebut, terdapat beberapa kendala yang masih terjadi hingga saat ini. Salah satu kendala utama adalah proses penyesuaian waktu global yang masih dilakukan secara

manual, yaitu dengan menekan tombol-tombol fisik pada alat. Penyesuaian ini rutin dilakukan sekitar tiga bulan sekali, karena dalam periode tersebut alat mengalami deviasi waktu yang cukup signifikan akibat tidak tersinkron dengan waktu dunia secara otomatis.

Selain itu, hasil pengamatan teknis menunjukkan bahwa tingkat akurasi waktu pada masing-masing alat berbeda, meskipun keduanya berada di lokasi yang sama. Ketidaksesuaian ini berpotensi menimbulkan kebingungan di kalangan jamaah, terutama ketika terjadi perbedaan waktu azan beberapa detik antara dua alat dalam satu area masjid. Gambar 1.3 menunjukkan tampilan kedua alat jadwal sholat digital di Masjid Rahmatan Lil 'Alamin (RLA) UPI YPTK Padang ini.



Gambar 1.3 Jadwal Sholat Masjid RLA Versi 1.0 dan Versi 2.0

Jika dilihat dari dekat, maka terdapat perbedaan waktu (dalam detik) antara alat jadwal sholat digital Versi 1.0 dan Versi 2.0. Perbedaan ini dapat dilihat pada gambar 1.4. Pada gambar 1.4 dapat dilihat bahwa alat Jadwal Sholat Digital Masjid RLA Versi 2.0 (a) menampilkan waktu **12:39:55**, sedangkan alat Jadwal Sholat Digital Masjid RLA Versi 1.0 (b) menampilkan waktu **12:40:04**. Waktu global saat itu adalah pukul **12:40:12**. Gambar 1.4 (c) adalah rincian dokumen pengambilan foto alat jadwal sholat tersebut.



(a) (b) (c)

Gambar 1.4 Perbedaan waktu pada Jadwal Sholat Digital Masjid RLA Versi 1.0 dan Versi 2.0.

Untuk mengatasi permasalahan tersebut, diperlukan sebuah solusi yang dapat mengoptimalkan sistem jadwal sholat digital agar lebih praktis, akurat, dan mudah dikelola. Salah satu pendekatan yang diusulkan adalah dengan mengembangkan sistem berbasis Internet of Things (IoT) menggunakan modul ESP32.[3][4] Melalui integrasi IoT ini, sinkronisasi waktu dapat dilakukan secara otomatis dengan mengakses waktu global melalui jaringan internet, yang kemudian dikendalikan dan dipantau menggunakan *smartphone*.

ESP32, sebagai modul IoT yang dilengkapi konektivitas Wi-Fi dan Bluetooth, akan bertindak sebagai penghubung antara aplikasi *smartphone* dan sistem kendali utama pada alat jadwal sholat. *Smartphone* akan menjalankan aplikasi khusus yang memungkinkan pengguna (admin atau petugas masjid) melakukan sinkronisasi waktu dan pengaturan jadwal sholat secara nirkabel. Waktu global yang diperoleh oleh ESP32 melalui protokol NTP (*Network Time Protocol*).[5] Waktu ini akan dikirimkan ke Arduino Mega2560, yang merupakan otak utama dari sistem jadwal sholat. Pengiriman dilakukan melalui komunikasi serial UART. Arduino Mega2560 kemudian akan memperbarui waktu internal dan menyesuaikan jadwal sholat berdasarkan informasi yang diterima.[6]

Dengan sistem ini, pengelolaan waktu tidak lagi membutuhkan penyesuaian manual, melainkan cukup dilakukan melalui aplikasi di *smartphone*. Selain meningkatkan keakuratan dan konsistensi waktu, pendekatan ini juga mempercepat proses pengaturan serta meminimalkan kesalahan akibat *human error*. Inovasi ini diharapkan dapat meningkatkan keandalan sistem jadwal sholat digital, sekaligus memberikan kenyamanan dan efisiensi bagi pengurus masjid dalam mengelola waktu ibadah secara modern dan terintegrasi.

1.2 Rumusan Masalah

Sistem jadwal sholat digital berbasis IoT di Masjid Rahmatan Lil 'Alamin (RLA) Universitas Putra Indonesia YPTK Padang memiliki beberapa permasalahan teknis utama. Rumusan masalah ini mencakup isu sinkronisasi waktu global, kendala teknis perangkat yang terpasang, kebutuhan pengaturan jeda iqomah per waktu sholat, serta integrasi antarmuka pengaturan melalui *smartphone*. Setiap aspek masalah dirumuskan dengan gaya formal-teknis sesuai standar rekayasa sistem. Sinkronisasi waktu global merupakan aspek krusial agar jadwal sholat dan panggilan adzan berjalan akurat. Dalam konteks IoT, sinkronisasi waktu sangat penting untuk menjaga agar seluruh perangkat beroperasi selaras. Protokol NTP (*Network Time Protocol*) umum digunakan untuk menyinkronkan jam dengan akurasi hingga milidetik, namun ketergantungan pada koneksi jaringan menimbulkan risiko gangguan. Penelitian sebelumnya mencatat bahwa apabila jaringan komunikasi bermasalah, fungsi sistem jadwal sholat digital dapat terhenti [1]. Oleh karena itu, diperlukan mekanisme sinkronisasi waktu yang andal beserta cadangan waktu lokal untuk menjaga keakuratan jadwal. Kendala teknis pada perangkat jadwal sholat digital yang terpasang juga perlu diidentifikasi. Sistem tradisional sering menggunakan MicroSD sebagai media penyimpanan jadwal [1]. Meskipun pendekatan ini dapat menampung data jadwal hingga lebih dari sepuluh tahun [1], hal tersebut membatasi fleksibilitas dalam memperbarui jadwal secara dinamis. Misalnya, perubahan jadwal baru hanya dapat diupdate secara manual jika ada perbedaan data. Selain itu, transfer data jadwal melalui jaringan Internet dilaporkan mengalami jeda waktu dan potensi kegagalan [1], yang menunjukkan perlunya peningkatan keandalan komunikasi dan penyimpanan. Keterbatasan

perangkat keras, seperti akurasi modul RTC, kapasitas tampilan, dan memori juga perlu dievaluasi untuk mendukung keandalan sistem secara keseluruhan. Kebutuhan pengaturan jeda iqomah per waktu sholat merupakan fitur penting yang harus diakomodasi. Iqomah adalah panggilan kedua sebelum sholat, biasanya dilakukan beberapa menit setelah adzan, dan interval waktu ini dapat berbeda pada tiap waktu sholat. Aplikasi jadwal sholat digital modern sudah menyediakan fitur “pengaturan jeda Iqomah per waktu sholat” [7], namun perangkat eksisting mungkin belum menerapkan kemampuan tersebut secara fleksibel. Kekurangan dalam mengatur jeda iqomah sesuai kebutuhan masjid dapat mengganggu kelancaran proses ibadah.

Dalam merancang dan merealisasikan sistem yang optimal, maka rumusan masalah dalam penelitian ini difokuskan untuk menjawab pertanyaan-pertanyaan berikut.

1. Bagaimana mekanisme sinkronisasi waktu global secara otomatis dapat diterapkan pada sistem jadwal sholat digital berbasis IoT di Masjid Rahmatan Lil ‘Alamin (RLA) Universitas Putra Indonesia YPTK Padang ?
2. Apa saja kendala teknis yang terjadi pada alat jadwal sholat digital yang telah dipasang sejak tahun 2016 dan 2019, serta bagaimana solusi teknis yang tepat untuk mengatasinya ?
3. Bagaimana sistem ini dapat menyediakan fitur pengaturan jeda iqomah yang fleksibel dan terpisah untuk setiap dari lima waktu sholat ?
4. Bagaimana perancangan integrasi antara *smartphone* dan sistem jadwal sholat digital dapat memudahkan petugas masjid dalam melakukan konfigurasi dan pemantauan ?

1.3 Batasan Masalah

Agar fokus penelitian tetap terarah dan terukur, maka penelitian ini dibatasi pada ruang lingkup berikut.

1. Penelitian ini hanya dilakukan pada dua unit alat jadwal sholat digital yang terpasang di Masjid Rahmatan Lil ‘Alamin (RLA) Universitas Putra Indonesia YPTK Padang, masing-masing dipasang pada tahun 2016 dan 2019.

2. Sinkronisasi waktu global dilakukan menggunakan modul ESP32 yang terhubung ke internet, dan waktu sinkronisasi diperoleh dari server waktu dunia (*NTP server*).
3. Komunikasi antara ESP32 dan Arduino Mega2560 menggunakan komunikasi serial UART, tanpa menggunakan protokol komunikasi kompleks lain seperti MQTT atau HTTP.
4. Sistem pengaturan jadwal sholat dan jeda iqomah dirancang agar dapat diatur melalui *web browser*.
5. Perancangan sistem hanya mencakup penyesuaian waktu sholat, pengaturan jeda iqomah, dan sinkronisasi waktu global, tidak mencakup pengaturan speaker adzan, integrasi dengan sensor lingkungan, atau fungsi multimedia lainnya.

1.4 Tujuan Penelitian

Penelitian ini bertujuan untuk :

1. Merancang dan mengimplementasikan sistem sinkronisasi waktu global otomatis pada perangkat jadwal sholat digital berbasis IoT.
2. Mengidentifikasi dan memperbaiki permasalahan teknis pada alat jadwal sholat yang telah terpasang sejak tahun 2016 dan 2019.
3. Mengembangkan fitur pengaturan jeda iqomah untuk lima waktu sholat harian, yang dapat diatur melalui aplikasi *smartphone*.
4. Mendesain sistem komunikasi serial antara ESP32 dan Arduino Mega2560 untuk mengirim data waktu dan konfigurasi jadwal.

1.5 Manfaat Penelitian

Berdasarkan hasil penelitian ini, diharapkan dapat memberikan beberapa manfaat sebagai berikut :

1. **Bagi Masjid** : Meningkatkan keakuratan dan efisiensi pengelolaan jadwal sholat digital tanpa perlu melakukan penyetelan manual secara rutin.

2. **Bagi Pengembang Alat** : Menjadi referensi dalam pengembangan sistem jadwal sholat digital berbasis IoT dengan pendekatan modular antara ESP32 dan Arduino Mega2560.
3. **Bagi Dunia Teknik** : Memberikan contoh implementasi rekayasa sistem berbasis mikrokontroler dan IoT yang aplikatif dalam konteks sosial keagamaan.
4. **Bagi laporan Profesi Insinyur** : Menunjukkan kemampuan profesional dalam merancang solusi teknis yang relevan, efisien, dan berorientasi pada kebutuhan pengguna lapangan.

1.6 Sistematika Penulisan

Sistematika penulisan berisi pembahasan apa yang akan ditulis disetiap Bab. Sistematika pada umumnya berupa paragraf yang setiap paragraf mencerminkan bahasan setiap Bab.

BAB I PENDAHULUAN

Bab ini terdiri atas sub-bab Latar Belakang, Rumusan Masalah, Batasan Masalah, Tujuan Penelitian, Manfaat Penelitian, dan Sistematika Penulisan.

BAB II TINJAUAN PUSTAKA

Bab ini memuat teori-teori yang mendukung, studi pustaka terkait sistem jadwal sholat digital, Arduino, ESP32, komunikasi serial.

BAB III METODOLOGI PENELITIAN

Bab ini Menjelaskan metode pengumpulan data, tahapan pengembangan sistem, desain komunikasi perangkat keras dan lunak, serta uji fungsional sistem.

BAB IV ANALISIS HASIL DAN PEMBAHASAN

Pada bab ini disajikan hasil implementasi alat dan aplikasi, serta analisis hasil pengujian sinkronisasi waktu dan pengaturan jadwal sholat.

BAB V PENUTUP

Bab ini menyimpulkan hasil penelitian dan memberikan saran untuk pengembangan sistem lebih lanjut.



BAB II

TINJAUAN PUSTAKA

2.1 Jadwal Sholat Digital

Jadwal sholat digital merupakan sistem elektronik yang dirancang untuk menampilkan waktu-waktu sholat secara otomatis sesuai dengan lokasi geografis tertentu. Sistem ini sangat membantu pengurus masjid dalam mengatur waktu ibadah dan memberikan informasi yang akurat kepada jamaah. Biasanya, jadwal sholat digital dilengkapi dengan fitur waktu adzan, iqomah, dan jam digital *real-time*. Namun, banyak dari perangkat ini masih bersifat statis, dan pengaturannya harus dilakukan secara manual melalui tombol pada perangkat keras.

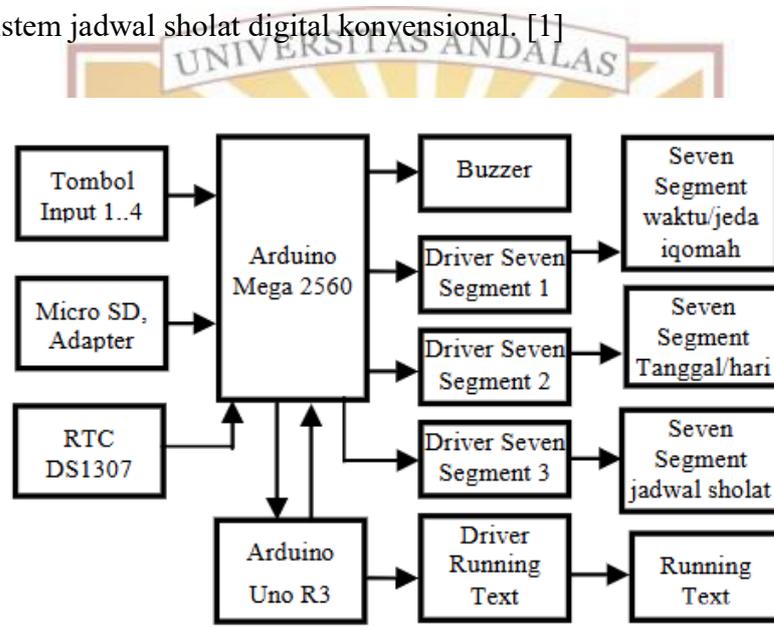


Gambar 2.1 Tampilan Umum Jadwal Sholat Digital di Masjid

Kelebihan dari sistem jadwal sholat digital adalah kemampuannya untuk memberikan informasi waktu ibadah secara visual dan konsisten, sehingga tidak diperlukan lagi penyampaian manual oleh pengurus masjid. Selain itu, tampilannya yang terang dan informatif menjadikannya lebih menarik dibandingkan papan manual atau jadwal cetak. Namun, perangkat ini juga memiliki keterbatasan, terutama dalam hal akurasi waktu jika tidak dilengkapi dengan sistem sinkronisasi otomatis. Deviasi waktu dapat terjadi setelah beberapa bulan penggunaan, terutama jika hanya bergantung pada jam internal (*Real Time Clock*).[8]

Dalam beberapa kasus, sistem ini juga tidak dilengkapi dengan antarmuka pengguna yang memadai untuk melakukan penyesuaian waktu atau jadwal,

sehingga pengguna yang kurang memahami teknis alat menjadi kesulitan dalam pengoperasiannya. Pengembangan teknologi jadwal sholat digital masa kini tidak hanya fokus pada fungsionalitas dasar, melainkan juga harus memperhatikan kemudahan operasional, keandalan waktu, dan fleksibilitas kontrol jarak jauh. Integrasi dengan teknologi mobile serta *Internet of Things* (IoT) menjadi sangat relevan untuk menjawab kebutuhan tersebut. Sistem yang dapat diatur dari perangkat *smartphone* oleh petugas masjid akan meningkatkan efisiensi dan kenyamanan dalam pengelolaan, serta meminimalkan potensi kesalahan akibat pengaturan manual yang tidak presisi. Gambar 2.2 berikut ini merupakan blok diagram sistem jadwal sholat digital konvensional. [1]



Gambar 2.2 Blok Diagram Sistem Jadwal Sholat Digital Konvensional

Lebih lanjut, sistem digital yang adaptif dan berbasis teknologi terkini dapat berkontribusi dalam meningkatkan kualitas pelayanan masjid kepada jamaah. Kehadiran sistem yang handal juga mencerminkan pengelolaan masjid yang modern, profesional, dan mengikuti perkembangan zaman, tanpa mengabaikan nilai-nilai tradisi dan kebutuhan umat.

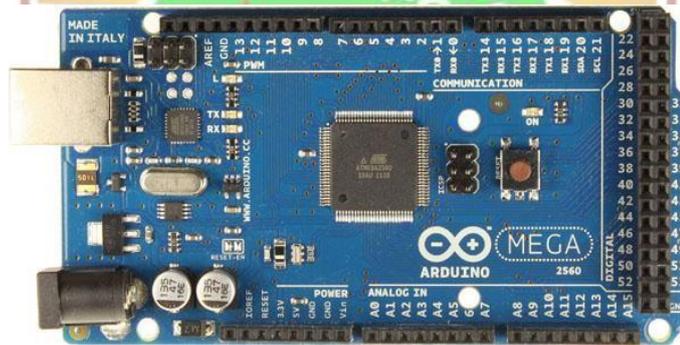
2.2 Arduino Mega2560

Arduino Mega2560 adalah papan pengembangan (*board development*) berbasis ATmega2560 yang memiliki kapasitas lebih besar dibanding varian

Arduino lainnya, baik dari sisi memori maupun jumlah pin input/output. Papan ini menyediakan 54 pin digital I/O, 16 pin input analog, dan 4 port komunikasi serial (UART), menjadikannya sangat ideal untuk proyek dengan banyak perangkat input dan output, seperti sistem jadwal sholat digital. Berikut ini beberapa data berkaitan dengan Arduino Mega2560.

1. Hardware Arduino Mega 2560

Komponen utama di dalam papan arduino adalah mikrokontroler 8 bit dengan merek ATmega yang dibuat oleh Atmel Corporation. Berbagai papan Arduino menggunakan tipe ATmega yang berbeda-beda tergantung dari spesifikasinya. Sebagai contoh Arduino uno menggunakan ATmega 328, sedangkan Arduino Mega 2560 yang lebih canggih menggunakan ATmega 2560. Bentuk dari Arduino Mega 2560 ditunjukkan oleh gambar 2.3 di bawah ini.[9]



Gambar 2.3 Arduino Mega 2560

Adapun data teknis board Arduino Mega2560 dapat dilihat pada tabel 2.1.[10]

Tabel 2.1 Data Teknis Board Arduino Mega 2560

Digital I/O Pins	54 (of which 15 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz

Arduino Mega 2560 versi 1.0 memiliki fitur-fitur baru seperti berikut :

- a. Terdapat penambahan pin SDA dan SCL yang terletak dekat dengan pin AREF. Selain itu, dua pin baru lainnya ditempatkan di dekat pin RESET, salah satunya adalah IOREF. Pin IOREF memungkinkan *shield* untuk menyesuaikan diri dengan tegangan yang digunakan oleh papan (*board*). Di masa depan, *shield* diharapkan kompatibel baik dengan papan yang menggunakan mikrokontroler AVR (beroperasi pada 5 Volt) maupun dengan Arduino Due (beroperasi pada 3.3 Volt). Terdapat juga dua pin tambahan yang saat ini tidak terhubung, dan disediakan untuk keperluan pengembangan di masa mendatang.
 - b. Sirkuit *RESET*
 - c. *Chip* ATmega16U2 menggantikan *chip* ATmega8U2.
2. Sumber daya tegangan Arduino Mega 2560

Arduino Mega2560 dapat diaktifkan melalui koneksi USB atau dengan catu daya eksternal. Sumber daya dipilih secara otomatis. Sumber daya eksternal (non-USB) dapat berasal baik adaptor AC-DC atau baterai. Adaptor dapat dihubungkan dengan mencolokkan steker 2,1 mm yang bagian tengahnya terminal positif ke jack sumber tegangan pada papan. Jika tegangan berasal dari baterai dapat langsung dihubungkan melalui header pin Gnd dan pin Vin dari konektor Power. Papan Arduino mega 2560 dapat beroperasi dengan pasokan daya eksternal 6 volt sampai 20 volt. Jika diberi tegangan kurang dari 7 volt, maka pin 5 volt mungkin akan menghasilkan tegangan kurang dari 5 volt dan ini akan membuat papan menjadi tidak stabil. Jika sumber tegangan menggunakan lebih dari 12 volt, regulator tegangan akan mengalami panas berlebihan dan bisa merusak papan. Rentang sumber tegangan yang dianjurkan adalah 7 volt sampai 12 volt.

Pin tegangan yang tersedia papan Arduino adalah sebagai berikut :

- a. VIN : adalah input tegangan untuk papan Arduino ketika menggunakan sumber daya eksternal. Input tegangan dapat diberikan tegangan melalui pin ini. Jika memasok tegangan untuk papan melalui *jack power*, kita bisa mengakses/mengambil tegangan melalui pin ini.

- b. 5V : sebuah pin yang mengeluarkan tegangan teregulasi 5 volt. Dari pin ini tegangan sudah diatur dari regulator yang tersedia (*built-in*) pada papan. Arduino dapat diaktifkan dengan sumber daya baik berasal dari *jack power* DC (7-12 volt), konektor USB (5 volt), atau pin VIN pada board (7-12 volt). Jangan memberikan tegangan melalui pin 5V atau 3.3V secara langsung tanpa melewati regulator, karena dapat merusak papan Arduino.
- c. 3V3 : sebuah pin yang menghasilkan tegangan 3.3 volt. Tegangan ini dihasilkan oleh regulator yang terdapat pada papan (on-board). Arus maksimum yang dihasilkan adalah 50 mA.
- d. GND : pin ground atau massa.
- e. IOREF : pin ini pada papan Arduino berfungsi untuk memberikan referensi tegangan yang beroperasi pada mikrokontroler.

3. Memori Arduino Mega2560

Arduino ATmega2560 memiliki 256 KB *flash* memori untuk menyimpan kode (yang 8 KB digunakan untuk *bootloader*), 8 KB SRAM dan 4 KB EEPROM (yang dapat dibaca dan ditulis dengan perpustakaan EEPROM).

4. Software IDE Arduino

Arduino diciptakan untuk para pemula bahkan yang tidak memiliki dasar bahasa pemrograman sama sekali. Arduino menggunakan bahasa C++ yang telah dipermudah melalui *library*. *Software* Arduino ini dapat di-*install* di berbagai *operating system* (OS) seperti: LINUX, Mac OS, Windows. Arduino tidak hanya sekedar sebuah alat pengembangan, tetapi kombinasi dari *hardware*, bahasa pemrograman dan *Integrated Development Environment* (IDE) yang canggih. IDE adalah sebuah *software* yang sangat berperan untuk menulis program, meng-*compile* menjadi kode biner dan meng-*upload* ke dalam memory mikrokontroler. Software IDE Arduino terdiri dari 3 (tiga) bagian:

- a. Editor program, untuk menulis dan mengedit program dalam bahasa *processing*. Listing program pada Arduino disebut *sketch*.

- b. *Compiler*, modul yang berfungsi mengubah bahasa *processing* (kode program) ke dalam kode biner karena kode biner adalah satu-satunya bahasa program yang dipahami oleh mikrokontroler.
- c. *Uploader*, modul yang berfungsi memasukkan kode biner kedalam memori mikrokontroler.

5. Konfigurasi pin Arduino Mega2560

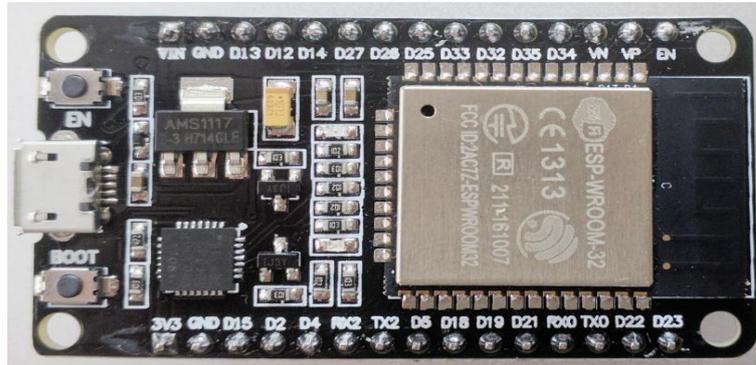
Konfigurasi pin ATmega 2560 adalah sebagai berikut:

- a. VCC merupakan pin yang digunakan sebagai masukan sumber tegangan.
- b. GND merupakan pin untuk Ground.
- c. XTAL1/ XTAL2, XTAL digunakan sebagai pin external clock.
- d. Port A, B, C, D, E, H, dan L merupakan 8 bit port I/O dengan internal *pull-up* resistor. Port G merupakan 6 bit port I/O dengan internal *pull up* resistor.
- e. Port F (PF0:PF7) dan Port K (PK0:PK7) merupakan pin I/O dan merupakan pin masukan ADC.
- f. AVCC adalah pin masukan untuk tegangan ADC.
- g. AREF adalah pin masukan untuk tegangan referensi eksternal.

Dalam penelitian ini, Arduino Mega2560 berperan sebagai unit pengolah utama (*main controller*) yang mengatur logika waktu sholat, waktu jeda iqomah, serta tampilan output ke panel display LED. Arduino Mega2560 ini juga melakukan proses pembacaan dan pengeksekusian perintah dari modul komunikasi eksternal, termasuk data yang dikirim dari modul ESP32.

2.3 Modul ESP32

ESP32 adalah sebuah mikrokontroler berdaya tinggi yang dikembangkan oleh *Espressif Systems*, merupakan penerus dari ESP8266.[11] Mikrokontroler ini dilengkapi dengan Wi-Fi dan Bluetooth secara *onboard*, menjadikannya sangat cocok untuk proyek Internet of Things (IoT). ESP32 memiliki arsitektur dual-core Tensilica Xtensa LX6, dengan kecepatan hingga 240 MHz, serta RAM sebesar 520 KB dan kapasitas penyimpanan flash hingga beberapa megabyte tergantung modulnya.



Gambar 2.4 Modul ESP32 Dev Board

Kemampuannya dalam menangani berbagai protokol komunikasi seperti UART, SPI, I2C, PWM, ADC, DAC, serta integrasi nirkabel Wi-Fi dan Bluetooth menjadikan ESP32 sangat fleksibel dalam pengembangan sistem embedded dan perangkat pintar.[12]

Beberapa fitur utama ESP32 yang menjadi keunggulannya antara lain:

- Dual Core Processor: Memungkinkan multitasking secara efisien.
- Wi-Fi 802.11 b/g/n: Terintegrasi langsung untuk konektivitas internet.
- Bluetooth v4.2 + BLE: Untuk komunikasi jarak dekat seperti pengendalian smartphone.
- GPIO hingga 34 pin: Dapat dikonfigurasi sebagai input/output, PWM, atau fungsi khusus lainnya.
- Analog Input (ADC) hingga 18 channel dan 2 channel DAC.
- Protokol komunikasi: UART, SPI, I2C, CAN, IR, dan PWM.
- Sensor internal: Sensor suhu, hall sensor.
- *Sleep Mode* hemat daya: Cocok untuk aplikasi IoT yang membutuhkan efisiensi energi.
- Kemampuan *Over-The-Air* (OTA): Pemrograman ulang firmware melalui jaringan Wi-Fi.

Berikut beberapa keunggulan ESP32 dibandingkan dengan Arduino Uno, Mega, dan bahkan pendahulunya ESP8266.[13]

Tabel 2.2 Perbandingan Fitur ESP32 dengan Arduino Uno, Mega dan ES8266

Fitur	ESP32	ESP8266	Arduino Uno/Mega
Prosesor	Dual-Core 240 MHz	Single-Core 80 MHz	AVR 16 MHz
Wi-Fi	Ada (integrated)	Ada	Tidak ada
Bluetooth	Ada (Classic + BLE)	Tidak ada	Tidak ada
ADC	Hingga 18 channel	1 channel	6–16 channel
GPIO	>30	±17	±20–50
Memori	520 KB RAM	160 KB RAM	2 KB (Uno), 8 KB (Mega)
Harga	Relatif murah	Murah	Bervariasi

Agar ESP32 dapat diprogram menggunakan Arduino IDE, perlu ditambahkan *board manager* khusus. Berikut adalah langkah-langkah lengkapnya:

Langkah 1: Install Arduino IDE

Pastikan Anda sudah menginstal Arduino IDE versi terbaru. Disarankan minimal versi 1.8.10 ke atas atau menggunakan Arduino IDE 2.0.

Langkah 2: Menambahkan URL Board Manager

1. Buka Arduino IDE.
2. Klik menu File > Preferences.
3. Pada kolom Additional Board Manager URLs, masukkan URL berikut:

https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json

Jika sudah ada URL lain, pisahkan dengan koma.

Langkah 3: Install Board ESP32

1. Klik menu Tools > Board > Boards Manager.
2. Ketik esp32 di kolom pencarian.

3. Pilih esp32 by Espressif Systems, lalu klik Install. Proses ini akan memakan waktu beberapa menit tergantung kecepatan internet Anda.

Langkah 4: Pilih Board dan Port

1. Setelah instalasi selesai, klik Tools > Board, lalu pilih jenis board Anda, misalnya :
 - ESP32 Dev Module
 - NodeMCU-32S
2. Sambungkan ESP32 ke komputer menggunakan kabel USB.
3. Pilih port yang sesuai di menu Tools > Port.

Langkah 5: Upload Sketch

1. Coba upload program sederhana seperti Blink untuk memastikan ESP32 berfungsi.
2. Jika saat upload muncul error Failed to connect, tekan dan tahan tombol BOOT pada ESP32 selama proses upload, lalu lepas setelah muncul "Connecting...".

Dengan langkah ini, ESP32 sudah siap digunakan pada berbagai proyek berbasis Arduino IDE, termasuk sebagai modul Wi-Fi yang berperan dalam sinkronisasi waktu dan kendali berbasis web pada sistem jadwal sholat digital yang sedang diteliti.

2.4 Konsep Internet of Things (IoT)

Internet of Things atau dikenal juga dengan singkatan IoT, merupakan sebuah konsep yang bertujuan untuk memperluas manfaat dari konektivitas internet yang tersambung secara terus-menerus. Adapun kemampuan seperti berbagi data, remote control, dan sebagainya, termasuk juga pada benda di dunia nyata. Contohnya bahan pangan, elektronik, koleksi, peralatan apa saja, termasuk benda hidup yang semuanya tersambung ke jaringan lokal dan global melalui sensor yang tertanam dan selalu aktif.

Makna serupa yang lain, *Internet of Things* (IoT) adalah sebuah konsep/skenario dimana suatu objek yang memiliki kemampuan untuk mentransfer data melalui jaringan tanpa memerlukan interaksi manusia ke manusia atau manusia ke komputer [14]. Istilah IoT (*Internet of Things*) mulai dikenal tahun 1999 yang saat itu disebutkan pertama kalinya dalam sebuah presentasi oleh Kevin Ashton, cofounder and executive director of the Auto-ID Center di MIT.

Dengan semakin berkembangnya infrastruktur internet, maka kita menuju babak berikutnya, di mana bukan hanya smartphone atau komputer saja yang dapat terkoneksi dengan internet. Namun berbagai macam benda nyata akan terkoneksi dengan internet. Sebagai contohnya dapat berupa : mesin produksi, mobil, peralatan elektronik, peralatan yang dapat dikenakan manusia (*wearables*), dan termasuk benda nyata apa saja yang semuanya tersambung ke jaringan lokal dan global menggunakan sensor dan atau aktuator yang tertanam.

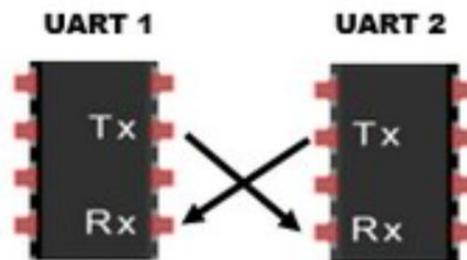
Dengan demikian dapat disimpulkan bahwa konsep Internet of Things (IoT) sendiri mengacu pada kemampuan perangkat fisik untuk saling terhubung melalui jaringan internet, memungkinkan pertukaran data dan kendali dari jarak jauh [15]. Dalam penelitian ini, ESP32 digunakan untuk melakukan sinkronisasi waktu otomatis dari NTP (*Network Time Protocol*) server dan mengirimkan data waktu terkini ke Arduino Mega2560 melalui komunikasi serial UART. Modul ESP32 juga menerima perintah dari *smartphone* petugas masjid, sehingga segala pengaturan bisa dilakukan secara nirkabel tanpa harus menyentuh alat secara langsung.

Dengan penerapan IoT, jadwal sholat digital menjadi sistem yang adaptif, modern, dan minim perawatan. Petugas masjid tidak perlu lagi menyesuaikan waktu secara manual, melainkan cukup melakukan sinkronisasi melalui koneksi Wi-Fi yang terhubung ke server waktu global.

2.5 Komunikasi Serial UART

UART (*Universal Asynchronous Receiver Transmitter*) adalah metode komunikasi serial asinkron yang umum digunakan untuk menghubungkan dua perangkat mikrokontroler [16]. Protokol ini menggunakan dua jalur utama, yaitu

TX (*Transmit*) dan RX (*Receive*), untuk mengirimkan data secara dua arah. UART tidak memerlukan clock eksternal seperti SPI atau I2C, sehingga lebih sederhana dalam implementasi perangkat keras [17].



Gambar 2.5 Diagram Komunikasi Serial UART antara ESP32 dan Arduino

UART merupakan protokol komunikasi serial yang memungkinkan perangkat untuk mengirim dan menerima data per bit secara asinkron. Artinya, data dikirimkan dalam bentuk urutan bit tanpa sinyal clock eksternal. Setiap perangkat memiliki tempat penyimpanan sementara (*buffer*) untuk menampung data yang dikirim dan diterima. Kecepatan komunikasi ditentukan oleh *baud rate* yang mewakili jumlah bit data yang dikirim dalam satu detik.

Dalam sistem ini, UART digunakan sebagai saluran komunikasi antara ESP32 dan Arduino Mega2560. Data hasil sinkronisasi waktu global dari ESP32 akan dikirim ke Arduino melalui jalur ini. Arduino kemudian membaca data tersebut dan memperbarui waktu sistem internalnya. Keandalan komunikasi serial ini sangat penting untuk menjaga akurasi sistem jadwal sholat digital dan menjamin bahwa data yang ditampilkan sesuai dengan waktu aktual.

2.6 Penelitian Terkait

Berbagai penelitian sebelumnya telah mengkaji pengembangan sistem jadwal sholat otomatis dengan beragam pendekatan teknologi. Salah satu yang paling banyak digunakan adalah mikrokontroler Arduino Uno dengan dukungan modul RTC (*Real Time Clock*), seperti yang digunakan oleh Emil Naf'an pada tahun 2019 dan 2021. Sistem tersebut mampu menampilkan jadwal sholat secara otomatis dengan tampilan visual berupa 7-segment display atau running text, yang

dikombinasikan dengan penyimpanan data menggunakan microSD. Akurasi sistem cukup baik dan stabil dalam pengujian jangka panjang. Namun, pengaturan waktu masih dilakukan secara manual dan sistem belum terkoneksi ke jaringan internet. [1] [2] [18]

Tabel 2.3 Perbandingan Fitur dari beberapa Penelitian Terdahulu

No	Penulis & Tahun	Platform	Fitur Utama	Kelebihan	Kelemahan
1	Emil Naf'an (2019) [1]	Arduino Uno, RTC, MicroSD, 7-Segment, Running Text	Menampilkan jadwal sholat dan aktivitas dosen secara otomatis	Akurasi tinggi, pengujian 7 hari non-stop, tampilan jelas	Tidak terhubung ke internet, pengaturan awal manual
2	Emil Naf'an (2021) [2]	Arduino, Seven Segment, Bluetooth	Pengiriman pesan jadwal sholat melalui running text via Bluetooth	Komunikasi nirkabel, mudah dioperasikan	Jangkauan terbatas, tidak terhubung ke internet
3	Yusuf H. Kanoi et al. (2023) [19]	Arduino Uno, RTC DS3231, LED Dot Matrix P10	Menampilkan jadwal sholat lima waktu secara digital	Tampilan jelas dan mudah dibaca, desain sederhana	Tidak terhubung ke internet, pembaruan jadwal manual
4	Helma Widya et al. (2020) [20]	Arduino Uno, LED Display, Bluetooth	Running text jadwal sholat yang dapat diperbarui melalui ponsel Android	Tampilan fleksibel, kontrol mudah	Keterbatasan jarak komunikasi Bluetooth
5	Syaiful Amri et al. (2023) [21]	Arduino, LED Display	Menampilkan jadwal waktu sholat secara otomatis di	Informasi akurat, meningkatkan kenyamanan jamaah	Tidak disebutkan konektivitas internet

			mushalla kampus		
6	Itmi Hidayat Kurniawan et al. (2022) [18]	Mikrokontroler, RTC, Display Digital	Penampil dan pengingat waktu sholat digital dengan waktu iqomat	Real-time, meningkatkan kesadaran waktu sholat	Tidak disebutkan konektivitas internet
7	Safiq Rosad, Anton Yudhana, Abdul Fadlil (2019) [22]	ATmega328, GPS Smartphone, 7-Segment Display	Perhitungan waktu sholat menggunakan metode ephemeris berdasarkan koordinat GPS dari smartphone	Akurasi tinggi sesuai lokasi pengguna, tidak bergantung pada koneksi internet	Tidak memiliki fitur kendali jarak jauh atau integrasi dengan aplikasi mobile
8	Sarifudin, Manshur, Angga Tirtana (2018) [23]	Arduino Uno, RTC DS3231, Dot Matrix P4, Bluetooth HC-06, Aplikasi Android (App Inventor)	Menampilkan jadwal sholat, jam digital, dan tanggal yang dapat diperbarui melalui koneksi Bluetooth dari smartphone Android	Tampilan besar dan jelas, pengaturan waktu fleksibel melalui aplikasi Android, tidak memerlukan koneksi internet	Tidak mendukung kontrol jarak jauh melalui internet, pengaturan awal memerlukan koneksi Bluetooth manual
9	Andhika Wahyu Putra, Rizki Nuryanto, Asni Tafrikhatin (2021) [24]	ESP32, LED Display, Speaker, Web Interface	Jam digital dengan pengingat kegiatan masjid yang dapat dikontrol melalui Wi-Fi menggunakan smartphone	Pengaturan mudah melalui browser, fitur pengingat kegiatan dengan output suara, tampilan LED yang dapat disesuaikan	Jangkauan Wi-Fi terbatas (maksimal 20 meter tanpa halangan), tidak mendukung kontrol jarak jauh melalui internet

10	M. Anton, M.H. Basri, Y. Prasetyo (2020) [25]	Arduino Uno, RTC, Running Text, Speaker USB	Jam istisya otomatis dengan tampilan teks berjalan dan suara azan	Tampilan waktu sholat yang jelas, pengingat audio otomatis, sistem sederhana dan hemat daya	Tidak terhubung ke internet, pengaturan awal manual, tidak mendukung kontrol jarak jauh
----	--	---	---	---	---

Seiring perkembangan teknologi, mulai banyak penelitian yang mengintegrasikan sistem dengan komunikasi nirkabel, seperti *Bluetooth*, yang memudahkan pengaturan dari *smartphone* Android. Helma Widya dan timnya mengembangkan sistem dengan fitur pembaruan jadwal sholat melalui ponsel, memberikan fleksibilitas namun terbatas pada jangkauan komunikasi Bluetooth yang pendek. Hal serupa juga ditemukan pada penelitian Sarifudin dan rekan-rekannya yang menggunakan Bluetooth HC-06 dan aplikasi Android sederhana berbasis App Inventor, namun tanpa dukungan akses jarak jauh melalui internet.

Penelitian lain oleh Yusuf H. Kanoi (2023) menggunakan LED Dot Matrix P10 untuk menampilkan jadwal sholat secara digital, menghasilkan tampilan yang jelas dan informatif, namun masih bergantung pada pembaruan manual. Sementara itu, penelitian oleh Safiq Rosad dan tim menawarkan pendekatan yang lebih akurat dengan menggunakan metode ephemeris berbasis GPS dari *smartphone*, memungkinkan sistem menyesuaikan waktu sholat berdasarkan lokasi pengguna tanpa perlu koneksi internet. Meskipun demikian, sistem ini belum mengintegrasikan kontrol atau pemantauan jarak jauh.

Pendekatan berbasis Internet of Things (IoT) mulai muncul dalam penelitian terbaru, seperti yang dilakukan oleh Andhika Wahyu Putra dan tim pada tahun 2021. Mereka mengembangkan sistem berbasis ESP32 yang dapat dikendalikan melalui web interface menggunakan koneksi Wi-Fi, lengkap dengan fitur pengingat kegiatan masjid berupa output suara melalui speaker. Sistem ini menawarkan fleksibilitas tinggi dan dapat dikontrol melalui browser di *smartphone*, meskipun jangkauan Wi-Fi menjadi batasan utama.

Sistem jam istisya otomatis juga dikembangkan oleh M. Anton dan rekan-rekannya pada tahun 2020. Mereka memanfaatkan kombinasi Arduino, RTC, dan

tampilan running text, serta audio azan melalui speaker USB. Sistem ini sederhana, hemat daya, dan efektif untuk digunakan di masjid, namun tidak mendukung akses kendali jarak jauh ataupun fitur berbasis internet.

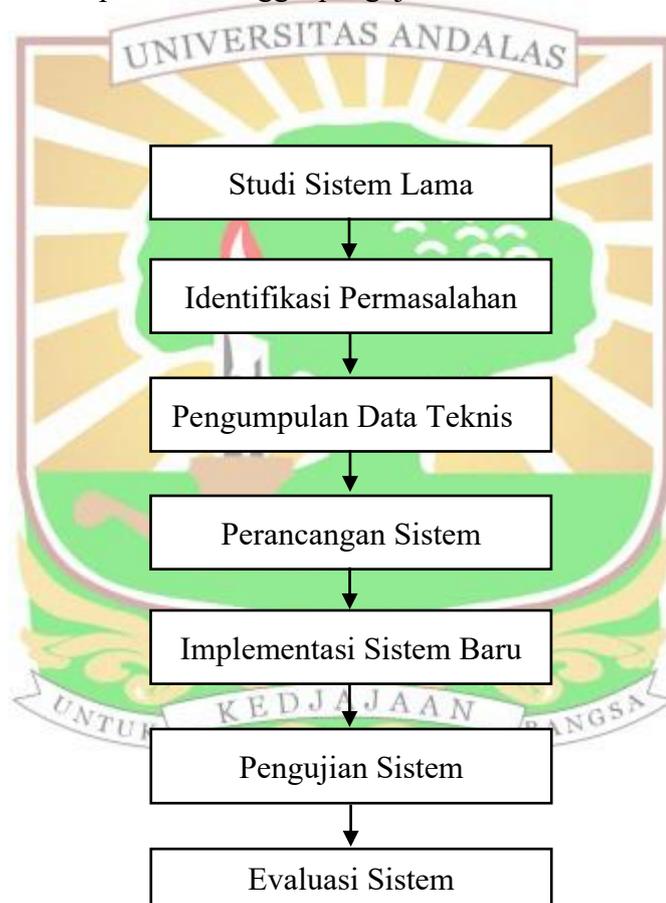
Dari analisis terhadap sepuluh penelitian ini, terlihat bahwa sebagian besar sistem masih bersifat lokal dan belum sepenuhnya memanfaatkan potensi IoT untuk manajemen jarak jauh. Meskipun tampilan dan akurasi waktu terus mengalami peningkatan, pengaturan jadwal dan kontrol sistem masih memerlukan intervensi manual atau koneksi terbatas seperti *Bluetooth*. Oleh karena itu, integrasi ESP32 dengan kemampuan koneksi Wi-Fi dan kontrol melalui aplikasi mobile menjadi peluang yang menjanjikan untuk pengembangan sistem yang lebih fleksibel, efisien, dan sesuai dengan kebutuhan modern pengelolaan masjid. Inovasi ini tidak hanya meningkatkan keakuratan dan kenyamanan, tetapi juga memungkinkan pengurus masjid melakukan pengaturan tanpa perlu berada di lokasi secara langsung. Dalam konteks ini, penelitian yang dilakukan di Masjid Rahmatan Lil ‘Alamin (RLA) Universitas Putra Indonesia YPTK Padang memberikan kontribusi baru dengan menerapkan pendekatan IoT secara menyeluruh. Sistem tidak hanya menyinkronkan waktu secara otomatis melalui internet, tetapi juga memungkinkan pengaturan jeda iqomah dan manajemen akses pengguna langsung dari perangkat *mobile*.



BAB III METODOLOGI PENELITIAN

3.1 Kerangka Penelitian

Kerangka penelitian ini mencakup beberapa aspek penting yang disusun secara sistematis untuk mencapai tujuan optimasi sistem penjadwalan sholat digital. Proses penelitian dilakukan melalui beberapa tahapan mulai dari studi awal sistem eksisting, identifikasi permasalahan, pengumpulan data teknis, perancangan sistem baru, implementasi optimasi, hingga pengujian dan evaluasi sistem yang telah diperbarui.



Gambar 3.1. Kerangka Penelitian

Berikut adalah poin-poin utama dalam kerangka penelitian ini:

1. Tujuan Penelitian

Penelitian ini bertujuan untuk mengoptimasi sistem penjadwalan sholat digital yang telah terpasang di Masjid Rahmatan Lil ‘Alamin (RLA) Universitas Putra

Indonesia YPTK Padang dengan menambahkan fitur berbasis *Internet of Things* (IoT), seperti sinkronisasi waktu global dan pengaturan jeda iqomah melalui *smartphone*.

2. Lokasi Implementasi

Lokasi utama dari penelitian ini adalah Masjid Rahmatan Lil ‘Alamin (RLA) Universitas Putra Indonesia YPTK Padang, tempat di mana dua sistem jadwal sholat telah dipasang sebelumnya (tahun 2016 dan 2019). Lokasi ini menjadi tempat pengujian langsung terhadap sistem hasil optimasi.

3. Tahapan Kegiatan

Proses penelitian terdiri dari beberapa tahapan, yaitu studi sistem lama, identifikasi permasalahan, pengumpulan data teknis, perancangan sistem baru, implementasi, dan pengujian sistem. Setiap tahapan dirancang untuk memberikan umpan balik terhadap keandalan dan efektivitas sistem yang dikembangkan.

4. Metode Evaluasi

Evaluasi dilakukan berdasarkan perbandingan antara sistem lama dan sistem yang telah dioptimasi. Kriteria evaluasi mencakup akurasi sinkronisasi waktu, kemudahan pengaturan jeda iqomah, kestabilan komunikasi antar perangkat, serta keandalan sistem dalam operasional harian masjid.

Dengan kerangka ini, diharapkan hasil penelitian mampu memberikan kontribusi nyata dalam meningkatkan efektivitas pengelolaan jadwal sholat di lingkungan masjid melalui pemanfaatan teknologi terkini. ini dilaksanakan untuk mengoptimasi sistem penjadwalan sholat digital pada Masjid Rahmatan Lil ‘Alamin (RLA) Universitas Putra Indonesia YPTK Padang yang sebelumnya telah dipasang pada tahun 2016 dan 2019. Penelitian ini bertujuan untuk mengidentifikasi kelemahan sistem lama dan menerapkan solusi berbasis IoT, khususnya dengan penambahan modul ESP32 sebagai penghubung sistem dengan perangkat *smartphone*.

3.2 Studi Sistem Lama

Untuk memperjelas perbedaan antara kedua sistem tersebut, berikut adalah fitur utama yang membedakan keduanya.

1. Sistem Tahun 2016:

- Kontrol tampilan menggunakan Arduino Mega2560.
- Menggunakan IC 74HC595 dan ULN2003 sebagai driver 7-segment.
- Menampilkan waktu sholat lokal secara statis.
- Pengaturan jeda iqomah melalui tombol lokal, tidak ada kendali jarak jauh.
- Tidak ada waktu syuruq

2. Sistem Tahun 2019

- Tetap menggunakan Arduino Mega2560 sebagai inti sistem.
- Fitur tambahan waktu syuruq.
- Tidak terintegrasi dengan internet atau waktu global.
- Sudah ada pengaturan tampilan *running text* menggunakan *smartphone*, namun belum mendukung pengaturan jeda iqomah dan waktu global melalui *smartphone*.

Sistem jadwal sholat digital yang dikembangkan pada tahun 2016 menggunakan Arduino Mega2560 sebagai mikrokontroler utama yang bertanggung jawab mengontrol tampilan waktu. Untuk mengendalikan modul 7-segment common anoda, sistem ini memanfaatkan kombinasi IC 74HC595 sebagai shift register dan ULN2003 sebagai driver, yang berfungsi untuk memperkuat arus ke masing-masing segmen LED. Sistem ini hanya mampu menampilkan waktu sholat harian secara statis berdasarkan data yang dimasukkan secara manual tanpa adanya pembaruan otomatis dari jaringan internet. Selain itu, pengaturan jeda iqomah dilakukan secara manual melalui tombol lokal yang terpasang di perangkat, sehingga memerlukan keterlibatan fisik dari pengurus masjid setiap kali ingin melakukan perubahan. Pada sistem ini, waktu syuruq belum ditampilkan, sehingga informasi tersebut tidak tersedia bagi jamaah yang ingin menunda sholat subuh hingga waktu syuruq berlalu. Sistem ini secara keseluruhan belum mendukung kendali jarak jauh dan belum terintegrasi dengan waktu global, sehingga akurasi waktu tergantung pada penyetelan manual.

Pada pengembangan sistem tahun 2019, perangkat keras utama masih menggunakan Arduino Mega2560 sebagai otak dari keseluruhan rangkaian. Pembaruan penting dari versi sebelumnya adalah penambahan fitur waktu syuruq, sehingga sistem mampu memberikan informasi lebih lengkap terkait jadwal sholat. Meski demikian, sistem ini belum terhubung ke internet dan tidak mendukung sinkronisasi waktu global secara otomatis, sehingga jam sistem masih bergantung pada penyesuaian manual yang rentan terhadap kesalahan waktu. Namun, terdapat kemajuan dalam hal antarmuka pengguna, yaitu dengan adanya fitur pengaturan tampilan *running text* melalui *smartphone*, yang memungkinkan pengurus masjid menambahkan pengumuman atau informasi lain yang ditampilkan di layar. Kendati begitu, fitur pengaturan jeda iqomah dan sinkronisasi waktu global melalui *smartphone* belum tersedia, sehingga fleksibilitas sistem masih terbatas. Sistem ini mencerminkan langkah awal menuju integrasi digital yang lebih modern, namun belum sepenuhnya mengadopsi teknologi Internet of Things (IoT).

3.3 Identifikasi Permasalahan

Permasalahan utama yang ditemukan pada sistem jadwal sholat digital yang telah terpasang sebelumnya, baik versi tahun 2016 maupun 2019, dapat diuraikan sebagai berikut :

1. Tidak adanya sinkronisasi waktu otomatis dengan waktu global (internet).

Kedua sistem sebelumnya masih mengandalkan pengaturan waktu secara manual melalui tombol atau potensiometer. Hal ini sangat rawan terhadap kesalahan dalam penyesuaian waktu, terutama ketika terjadi perubahan waktu resmi seperti penyesuaian waktu musim atau ketidakakuratan jam internal. Akibatnya, waktu sholat yang ditampilkan sering kali tidak sinkron dengan waktu resmi dari otoritas seperti BMKG atau Kemenag, dan dapat menyebabkan keterlambatan atau ketidaksesuaian dalam pelaksanaan ibadah di masjid.

2. Tidak adanya pengaturan jeda iqomah melalui *smartphone*.

Sistem tahun 2016 dan sistem tahun 2019, pengaturan jeda iqomah hanya dapat dilakukan melalui tombol lokal yang terpasang di perangkat. Tidak adanya akses jarak jauh membuat pengurus masjid kesulitan untuk melakukan penyesuaian, terutama jika sedang tidak berada di lokasi perangkat. Hal ini

menyulitkan penyesuaian cepat yang sering kali diperlukan dalam situasi khusus seperti adanya ceramah tambahan, hujan deras, atau pergantian imam.

3. Ketergantungan pada pengaturan manual.

Seluruh konfigurasi dan pengaturan waktu, baik waktu sholat maupun jeda iqomah, dilakukan secara manual, baik melalui komputer lokal (upload ulang program) atau tombol fisik. Pendekatan ini tidak hanya menyulitkan proses pemeliharaan sistem, tetapi juga mengurangi efisiensi dalam manajemen waktu ibadah. Di era digital saat ini, pengurus masjid memerlukan sistem yang mudah diakses dan diatur secara fleksibel dari berbagai perangkat.

4. Keterbatasan fitur tampilan dan informasi:

Pada sistem 2016, tidak tersedia informasi tambahan seperti waktu syuruq atau pengumuman penting. Sedangkan sistem 2019 meskipun sudah memiliki fitur *running text* yang dapat dikendalikan melalui *smartphone*, fitur-fitur lainnya seperti jeda iqomah dan pengaturan waktu masih belum dapat diakses secara jarak jauh. Hal ini menunjukkan bahwa sistem belum sepenuhnya memanfaatkan potensi Internet of Things (IoT) secara menyeluruh.

Permasalahan-permasalahan ini menjadi dasar dalam perancangan sistem optimasi yang lebih modern dan terintegrasi. Dengan memanfaatkan modul ESP32 yang mendukung konektivitas Wi-Fi, sistem baru dirancang untuk dapat melakukan sinkronisasi waktu secara otomatis menggunakan protokol NTP (*Network Time Protocol*) dan memungkinkan pengaturan jadwal dan jeda iqomah melalui *smartphone* melalui antarmuka web. Inovasi ini bertujuan meningkatkan akurasi, kenyamanan, dan fleksibilitas dalam pengelolaan jadwal ibadah di lingkungan masjid.

3.4 Perancangan Sistem Optimasi

Sistem optimasi dirancang dengan mengintegrasikan dua buah mikrokontroler, yaitu Arduino Mega2560 dan ESP32, masing-masing dengan fungsi yang spesifik dan saling melengkapi. ESP32 digunakan sebagai penghubung sistem ke jaringan internet dan antarmuka pengguna, sedangkan Arduino Mega2560 bertugas mengelola tampilan waktu dan logika utama sistem.

a. Fungsi Arduino Mega2560 :

- Mengontrol tampilan jadwal sholat digital menggunakan 7-segment *common anoda*.
- Mengatur tampilan waktu sholat, waktu syuruq, jam berjalan, serta countdown iqomah.
- Menerima data dari ESP32 melalui komunikasi serial UART untuk memperbarui parameter seperti waktu sholat dan jeda iqomah.

b. Fungsi ESP32 :

- Menyinkronkan waktu secara otomatis melalui internet menggunakan protokol NTP.
- Menyediakan antarmuka pengguna berbasis web (*web-based UI*) yang dapat diakses melalui *smartphone* atau laptop.
- Mengirimkan data konfigurasi seperti jeda iqomah dan waktu sholat ke Arduino Mega2560 secara *real-time*.

Komunikasi antar mikrokontroler dilakukan melalui protokol UART. Dalam desain ini, ESP32 berfungsi sebagai otak untuk komunikasi dan kendali jarak jauh, sedangkan Arduino Mega2560 tetap berfungsi sebagai otak logika dan pengendali tampilan. Pendekatan ini menjadikan sistem lebih modular dan mudah dikembangkan lebih lanjut di masa depan.

3.5 Implementasi dan Pengujian Sistem

Setelah proses perancangan selesai, tahap implementasi dilakukan di Masjid Rahmatan Lil ‘Alamin (RLA) Universitas Putra Indonesia YPTK Padang. Sistem dirakit dan diintegrasikan ke dalam perangkat eksisting, kemudian dilakukan pengujian menyeluruh untuk mengevaluasi kinerja dan kestabilan sistem. Pengujian dilakukan dalam beberapa aspek penting, yaitu:

1. Uji Sinkronisasi Waktu :

- a. Memastikan ESP32 dapat terhubung ke internet dan mengambil waktu global dari server NTP.
- b. Membandingkan hasil waktu tampilan dengan waktu resmi BMKG.

2. Uji Pengaturan Jeda Iqomah :

- Mengakses antarmuka web melalui *smartphone* untuk mengubah jeda iqomah.
- Memastikan pengaturan tersebut diterima Arduino dan ditampilkan sebagai *countdown*.

3. Uji Komunikasi Serial (UART) :

- Mengirim dan menerima data dari ESP32 ke Arduino dan sebaliknya.
- Mengamati stabilitas koneksi selama penggunaan normal.

4. Uji Keandalan Operasional :

- Menjalankan sistem selama beberapa hari dalam kondisi sebenarnya
- Mengamati respons terhadap gangguan koneksi, pemadaman listrik, serta perubahan pengaturan.

Waktu implementasi dan pengujian sistem dirinci dalam tabel berikut:

Tabel 3.1 Jadwal Pelaksanaan Penelitian

No	Tahapan Kegiatan	Waktu Pelaksanaan (Tahun 2025)											
		Februari				Maret				April			
		1	2	3	4	1	2	3	4	1	2	3	4
1	Studi sistem lama												
2	Perancangan sistem optimasi												
3	Implementasi sistem												
4	Pengujian fungsional awal												
5	Evaluasi dan pengujian lanjutan												
6	Pembuatan Laporan												

Hasil pengujian dibandingkan langsung dengan sistem tahun 2016 dan 2019. Fokus utama evaluasi terletak pada peningkatan fleksibilitas pengaturan, keakuratan waktu, kemudahan penggunaan antarmuka, dan keandalan operasional sistem selama periode pengujian. Dengan sistem baru ini, diharapkan pengelolaan waktu ibadah di masjid menjadi lebih efisien dan responsif terhadap kebutuhan jamaah dan pengurus.

BAB IV HASIL DAN PEMBAHASAN

4.1 Implementasi Sistem

Hasil dari proses perancangan dan implementasi sistem menunjukkan bahwa optimasi sistem jadwal sholat digital berbasis ESP32 dan Arduino Mega2560 dapat berjalan dengan baik. Sistem ini berhasil mengintegrasikan fitur sinkronisasi waktu global serta pengaturan jeda iqomah melalui antarmuka *smartphone*. ESP32 bertugas menangani komunikasi jaringan dan antarmuka pengguna, sedangkan Arduino Mega2560 mengelola tampilan waktu dan countdown iqomah pada 7-segment display. Adapun fitur-fitur yang berhasil diimplementasikan antara lain :

1. Sinkronisasi waktu secara otomatis melalui server NTP.
2. Antarmuka web yang dapat diakses melalui *smartphone* untuk pengaturan jeda iqomah.
3. Komunikasi serial antara ESP32 dan Arduino untuk pertukaran data.
4. Tampilan waktu sholat, jam digital, serta countdown jeda iqomah secara real-time.

Pengoperasian alat dapat dilakukan dengan cukup mudah. Pengurus masjid hanya perlu menyambungkan ESP32 ke jaringan Wi-Fi, kemudian mengakses halaman antarmuka dari *smartphone*. Semua pengaturan dilakukan melalui web interface yang intuitif dan responsif.

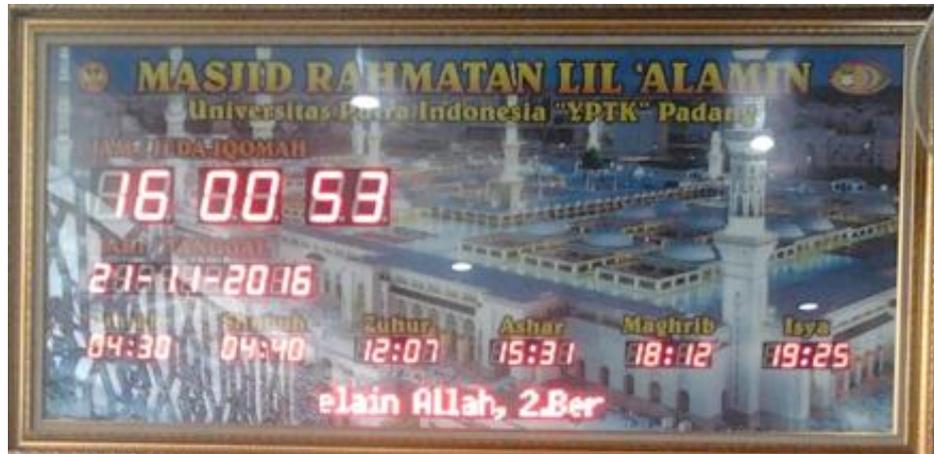
4.2 Dokumentasi Pengujian Alat

Pengujian dilakukan terhadap alat yang telah dirakit dan dipasang di lokasi. Berikut dokumentasi pengujian:

4.2.1 Sistem Tahun 2016

- Mikrokontroler : Arduino Mega2560.
- Driver tampilan : 74HC595 + ULN2003.
- Tampilan : 7-Segment *common anoda*.

- Fitur : Penampilan waktu sholat lokal secara statis.
- Kelemahan : Tidak ada waktu syuruq, dan tidak terkoneksi internet.



Gambar 4.1 Tampilan sistem jadwal sholat tahun 2016

4.2.2 Sistem Tahun 2019

- Mikrokontroler : Arduino Mega2560.
- Tambahan fitur : Penambahan waktu syuruq, pengaturan tampilan melalui *smartphone*.
- Kekurangan : Tidak ada sinkronisasi waktu otomatis dan tidak bisa mengatur jeda iqomah via *smartphone*.



Gambar 4.2 Tampilan sistem jadwal sholat tahun 2019

4.2.3 Sistem Hasil Optimasi Tahun 2025

Perangkat ini menggunakan mikrokontroler Arduino Mega2560 yang dipadukan dengan ESP32 sebagai inti sistemnya. Beberapa fitur utama yang ditawarkan antara lain adalah sinkronisasi waktu secara otomatis melalui NTP

(*Network Time Protocol*), serta kemampuan pengaturan jeda iqomah dan parameter lainnya langsung dari smartphone pengguna. Selain itu, sistem ini juga mampu menampilkan *countdown* iqomah secara otomatis berdasarkan input yang dikirimkan dari *smartphone*. Keunggulan utama dari sistem ini adalah fleksibilitas yang tinggi, responsivitas terhadap kondisi aktual di masjid, serta akurasi yang lebih baik dibandingkan sistem konvensional.

Setelah peralatan dibuat, selanjutnya peralatan diuji untuk mengetahui sejauh mana kinerja dari peralatan yang dirancang. Pengujian dilakukan pada beberapa rangkaian dari blok diagram antara lain ; pengujian rangkaian tombol, *Seven Segment*, *Buzzer*. Disamping itu pengujian juga dilakukan pada sinkronisasi waktu global dan pengujian pengiriman data jeda iqomah.

4.2.3.1 Pengujian Rangkaian Tombol

Pengujian dilakukan dengan cara mengamati nilai tegangan pada saat tombol dalam kondisi ditekan maupun tidak ditekan. Hasil pengukuran ditampilkan pada Tabel 4.1.

Tabel 4.1. Pengujian Tombol

No.	Kondisi Tombol	Tegangan Output (VDC)
1	Ditekan	0.00 V
2	Tidak Ditekan	4.99 V

Berdasarkan data pengujian, diketahui bahwa ketika tombol ditekan, tegangan output sebesar 0,00 VDC, yang dalam sistem digital diartikan sebagai logika '0' (*Low*). Hal ini terjadi karena jalur output tombol terhubung langsung ke ground. Sebaliknya, saat tombol tidak ditekan, tegangan output tercatat sebesar 4,99 VDC. Kondisi ini disebabkan oleh koneksi output ke sumber tegangan melalui resistor sebesar 4,7 K Ω , yang dalam sistem digital dikategorikan sebagai logika '1' (*High*). Dengan demikian, dapat disimpulkan bahwa rangkaian tombol input 1 hingga 4 berfungsi dengan baik dan sesuai untuk dibaca sebagai input digital oleh modul Arduino.

4.2.3.2 Pengujian *Seven Segment* Waktu (jam, menit, detik)

Pengujian dilakukan dengan mengamati secara langsung kinerja *Seven Segment* dalam menampilkan angka 0 hingga 9 pada format waktu (jam, menit, dan detik). Hasil pengujian menunjukkan bahwa *Seven Segment* mampu menampilkan seluruh angka dari 0 sampai 9 dengan baik.



Gambar 4.3. Pengujian Tampilan *Seven Segment* Waktu (jam, menit, detik)

Sebagian hasil dari proses pengujian ditampilkan pada Gambar 4.3. Berdasarkan hasil tersebut, dapat disimpulkan bahwa sistem *Seven Segment* dalam menampilkan waktu (jam, menit, dan detik) bekerja secara optimal.

4.2.3.3 Pengujian *Seven Segment* Tanggal / Hari

Pengujian dilakukan dengan mengamati secara langsung tampilan tanggal dan hari pada *Seven Segment*, lalu membandingkannya dengan data tanggal yang ditampilkan pada laptop (PC). Hasil dari pengujian ini ditampilkan pada Gambar 4.4.



(a)

(b)

(c)

(a) Tampilan Hari

(b) Tampilan Tanggal

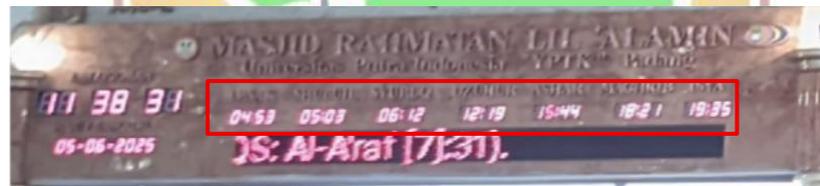
(c) Tampilan Setting waktu Pada Laptop (PC)

Gambar 4.4. Pengujian Tampilan *Seven Segment* Tanggal / Hari

Dalam gambar 4.4 tersebut terlihat bahwa tampilan hari dan tanggal muncul secara bergantian dengan jeda waktu sekitar 3 detik. Interval waktu ini dipilih berdasarkan estimasi rata-rata waktu yang dibutuhkan seseorang untuk membaca informasi hari dan tanggal, yaitu sekitar ± 3 detik. Setelah dilakukan perbandingan antara tampilan pada *Seven Segment* dan pengaturan waktu pada laptop, hasil yang diperoleh sama, yaitu hari Kamis bertepatan dengan tanggal 05 Juni 2025. Berdasarkan hasil tersebut, dapat disimpulkan bahwa rangkaian *Seven Segment* untuk penunjuk waktu (jam, menit, dan detik) berfungsi dengan baik.

4.2.3.4 Pengujian *Seven Segment* Jadwal Sholat

Pengujian dilakukan dengan cara mengamati secara langsung tampilan waktu sholat pada *Seven Segment*, lalu membandingkannya dengan data jadwal sholat yang tersimpan di dalam MicroSD. Hasil dari pengujian ini ditunjukkan pada Gambar 4.5.



(a)

Tgl	Subuh	Syuruq (Terang Sadi)	Dhuha	Zuhur	Ashar	Magrib	Iyaa	Iyaa Kiblat
01	05:03	06:11	06:40	12:18	15:43	18:20	19:34	16:31:28
02	05:03	06:12	06:40	12:18	15:43	18:20	19:34	16:34:28
03	05:03	06:12	06:40	12:18	15:43	18:20	19:35	16:37:25
04	05:03	06:12	06:40	12:18	15:43	18:20	19:35	16:40:18
05	05:03	06:12	06:41	12:19	15:44	18:21	19:35	16:43:06
06	05:03	06:12	06:41	12:19	15:44	18:21	19:35	16:45:49
07	05:03	06:12	06:41	12:19	15:44	18:21	19:35	16:48:32

(b)

File	Edit	View
2025 06 05 04:53	05:03	06:12 12:19 15:44 18:21 19:35
2025 06 06 04:53	05:03	06:12 12:19 15:44 18:21 19:35
2025 06 07 04:53	05:03	06:12 12:19 15:44 18:21 19:36
2025 06 08 04:53	05:03	06:13 12:19 15:44 18:21 19:36
2025 06 09 04:53	05:03	06:13 12:19 15:44 18:21 19:36
2025 06 10 04:54	05:04	06:13 12:20 15:45 18:22 19:36
2025 06 11 04:54	05:04	06:13 12:20 15:45 18:22 19:37

(c)

- (a) Tampilan Jadwal Sholat Pada Alat
- (b) Tampilan Jadwal Sholat Pada Kalender
- (c) Tampilan Jadwal Sholat Pada MicroSD

Gambar 4.5. Pengujian Tampilan *Seven Segment* Jadwal Sholat

Dari gambar 4.5 tersebut, terlihat bahwa waktu sholat untuk tanggal 05 Juni 2025 yang ditampilkan oleh alat sesuai dengan jadwal pada kalender maupun data di MicroSD. Rincian nilai tersebut dapat dilihat pada Tabel 4.2.

Tabel 4.2. Pengujian Data Jadwal Sholat

No.	Nama	Alat	Kalender	MicroSD	Hasil
1	Imsak	04:53	04:53	04:53	Sama
2	Shubuh	05:03	05:03	05:03	Sama
3	Syuruq	06:12	06:12	06:12	Sama
4	Dzuhur	12:19	12:19	12:19	Sama
5	Ashar	15:44	15:44	15:44	Sama
6	Maghrib	19:21	19:21	19:21	Sama
7	Isya	19:35	19:35	19:35	Sama

Dengan demikian dari hasil yang diperoleh dapat dikatakan bahwa rangkaian *Seven Segment* jadwal sholat dapat bekerja dengan baik.

4.2.3.5 Pengujian Rangkaian Buzzer

Pengujian dilakukan dengan memberikan sinyal tegangan berupa logika '0' dan '1' pada input rangkaian *Buzzer*, kemudian mengamati bunyi yang dihasilkan. Hasil dari pengujian ini disajikan dalam Tabel 4.3.

Tabel 4.3. Pengujian *Buzzer*

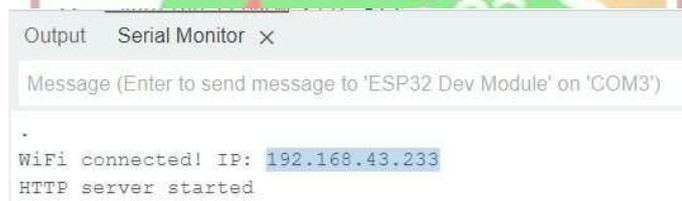
No.	Logika Input	Kondisi <i>Buzzer</i>
1	'0'	Tidak aktif
2	'1'	Aktif / Mengeluarkan Suara / bunyi yang bisa didengar dalam radius ± 30 meter.

Hasil pengujian menunjukkan bahwa saat diberikan logika '0', *Buzzer* berada dalam kondisi nonaktif (tidak menghasilkan suara). Sebaliknya, ketika logika '1' diberikan, *Buzzer* aktif dan menghasilkan bunyi yang dapat terdengar hingga jarak sekitar ± 30 meter. Berdasarkan pengujian ini, dapat disimpulkan bahwa rangkaian

Buzzer berfungsi dengan baik dan memenuhi kriteria untuk digunakan sebagai penanda waktu masuk sholat.

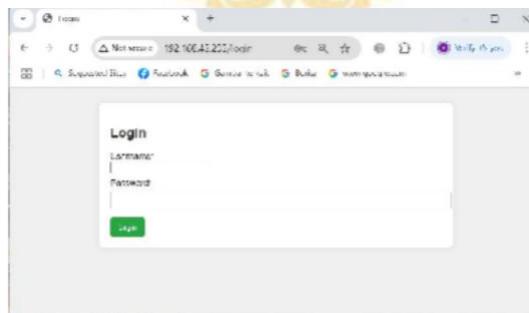
4.2.3.6 Pengujian Sinkronisasi Waktu Global

Pengujian ini bisa dilakukan menggunakan browser pada *smartphone* maupun browser pada laptop. Sistem akan membaca data waktu yang berasal dari server *Network Time Protocol* (NTP) yang terdapat pada *smartphone* maupun laptop. Sebelum pengiriman data waktu tersebut, pastikan modul ESP32 sudah terhubung ke jaringan Wi-Fi. Jika sudah terhubung, selanjutnya lakukan pengaturan sinkronisasi waktu, admin melakukan login pada *Web browser* dengan memasukkan alamat IP : `http://192.168.43.233` . Alamat ini diperoleh dari tampilan pada Serial Monitor Arduino IDE, seperti ditunjukkan pada gambar 4.6 berikut ini.



Gambar 4.6. Tampilan Alamat IP Akses ESP32

Selanjutnya lakukan login sebagai Admin dan masukkan *passwordnya*, seperti tampilan pada gambar 4.7.



Gambar 4.7. Tampilan Login ke Sistem

Setelah berhasil login, Selanjutnya akan tampil *Dashboard* untuk pengaturan Jeda Iqomah dan Sinkronisasi Waktu. Untuk melakukan Sinkronisasi Waktu *Network Time Protocol* (NTP) klik tombol Kirim Waktu ke Arduino, seperti ditunjukkan pada gambar 4.8.



Gambar 4.8. Tampilan *Dashboard* Pengaturan Sinkronisasi Waktu NTP

Gambar 4.9 menunjukkan tampilan hasil pengaturan sinkronisasi Waktu *Network Time Protocol* (NTP) pada alat. Tampilan pada Seven Segment waktu (jam, menit, dan detik) menunjukkan angka : 11:39:45. Masing-masing alat jadwal sholat menunjukkan nilai yang sama. Ini menunjukkan bahwa pengaturan sinkronisasi waktu NTP dapat bekerja dengan baik.

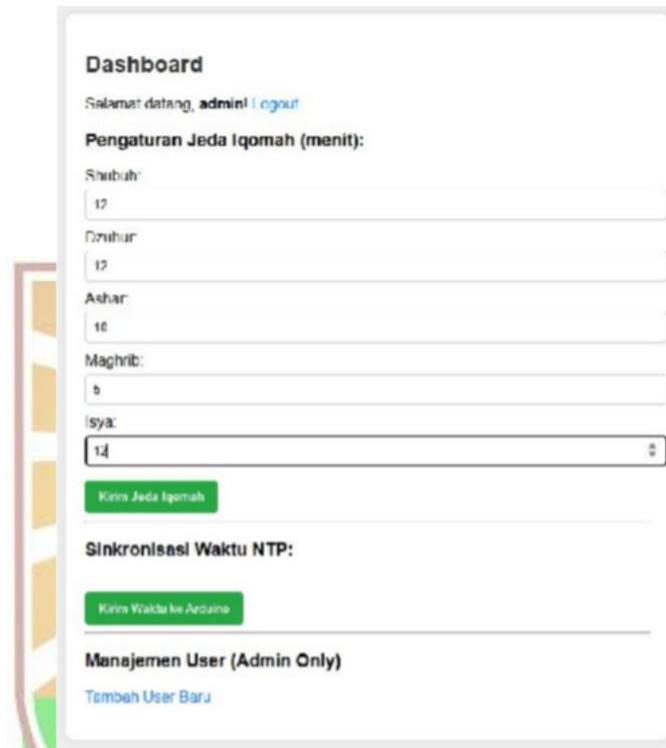


Gambar 4.9. Tampilan Hasil Pengaturan Sinkronisasi Waktu NTP Pada Alat

4.2.3.7 Pengujian Waktu Jeda Iqomah

Pengujian pengaturan waktu jeda Iqomah dilakukan dengan mengirimkan data waktu jeda Iqomah ke Arduino Mega2560 melalui ESP32. Pengaturan jeda

Iqomah dilakukan secara bersamaan untuk kesemua waktu sholat, yaitu waktu Shubuh, Dzuhur, Ashar, Maghrib dan Isya. Perubahan nilai waktu jeda Iqomah sangat mudah dilakukan, cukup dengan menekan tombol panah atas maupun bawah disisi kanan *Dashboard*, seperti ditunjukkan pada gambar 4.10.



The screenshot shows a web dashboard titled "Dashboard" with a user greeting "Selamat datang, admin! Logout". The main section is "Pengaturan Jeda Iqomah (menit):" and contains five input fields for different prayer times: Shubuh (12), Dzuhur (12), Ashar (10), Maghrib (5), and Isya (14). Each field has a small arrow on the right side for navigation. Below these fields are two green buttons: "Kirim Jeda Iqomah" and "Sinkronisasi Waktu NTP:". The bottom section is "Manajemen User (Admin Only)" with a link "Tambah User Baru".

Gambar 4.10. Tampilan *Dashboard* Pengaturan Sinkronisasi Waktu Jeda Iqomah

Setelah tombol Kirim Jeda Iqomah diklik, maka nilai waktu Shubuh, Dzuhur, Ashar, Maghrib dan Isya akan dikirimkan ke Arduino Mega2560. Setelah dilakukan pengujian, maka diperoleh hasil bahwa nilai jeda Iqomah yang dikirimkan ke Arduino Mega2560 sama dengan nilai yang ditampilkan oleh alat Jadwal sholat Digital. Dengan demikian pengaturan waktu jeda Iqomah melalui ESP32 yang dikirim pada smartphone dalam bekerja dengan baik.

4.3 Analisis Perbandingan Sistem

Analisis perbandingan dilakukan terhadap tiga versi sistem jadwal sholat digital yang telah diterapkan di Masjid Rahmatan Lil 'Alamin (RLA) Universitas Putra Indonesia YPTK Padang, yaitu sistem tahun 2016, 2019, dan sistem hasil

optimasi tahun 2025. Setiap sistem memiliki karakteristik dan fitur yang mencerminkan perkembangan teknologi serta kebutuhan operasional masjid yang semakin kompleks.

Sistem tahun 2016 merupakan sistem paling dasar dengan fungsi utama menampilkan waktu sholat lokal secara statis menggunakan tampilan 7-segment yang dikendalikan oleh Arduino Mega2560. Namun, sistem ini tidak memiliki fitur jeda iqomah, tidak mencantumkan waktu syuruq, dan tidak terintegrasi dengan jaringan internet. Pengaturan waktu dilakukan secara manual sehingga berpotensi terjadi ketidaksesuaian dengan waktu sholat yang resmi.

Kemudian, sistem tahun 2019 memperlihatkan adanya pengembangan dengan penambahan fitur waktu syuruq dan kemampuan untuk mengatur tampilan running text melalui smartphone. Meskipun demikian, sistem ini masih belum mendukung pengaturan jeda iqomah secara fleksibel dan belum terhubung ke internet, sehingga sinkronisasi waktu masih dilakukan secara manual seperti pada sistem sebelumnya.

Adapun sistem hasil optimasi tahun 2025 mengintegrasikan teknologi Internet of Things (IoT) dengan memanfaatkan modul ESP32 sebagai penghubung ke jaringan internet. Sistem ini dapat melakukan sinkronisasi waktu otomatis melalui protokol NTP dan menyediakan antarmuka web yang dapat diakses melalui smartphone. Hal ini memungkinkan pengurus masjid mengatur jeda iqomah serta parameter lainnya secara jarak jauh dan lebih fleksibel. Selain itu, fitur *countdown* iqomah ditampilkan secara otomatis, sehingga lebih membantu dalam mengatur waktu pelaksanaan sholat berjamaah.

Untuk memperjelas perbedaan antar sistem, berikut disajikan tabel perbandingan fitur utama.

Tabel 4.4 Perbandingan Sistem Penjadwalan Sholat

Parameter	Sistem 2016	Sistem 2019	Sistem 2025 (Optimasi)
Sinkronisasi waktu global	Tidak	Tidak	Ya
Pengaturan jeda iqomah via HP	Tidak	Tidak	Ya
Waktu syuruq	Tidak	Ya	Ya
Tampilan via 7-segment	Ya	Ya	Ya
Pengaturan via tombol lokal	Tidak	Ya	Opsional
Antarmuka Web	Tidak	Sebagian	Ya

Berdasarkan tabel di atas, dapat disimpulkan bahwa sistem hasil optimasi tahun 2025 menawarkan solusi yang lebih lengkap dan adaptif terhadap kebutuhan pengurus masjid. Fitur-fitur modern yang diterapkan berhasil mengatasi keterbatasan sistem sebelumnya, serta memberikan kemudahan operasional dan peningkatan efisiensi dalam manajemen jadwal sholat.

4.4 Evaluasi Kinerja Sistem

Evaluasi kinerja dilakukan untuk menilai sejauh mana sistem hasil optimasi mampu menjawab permasalahan pada sistem sebelumnya serta memenuhi kebutuhan operasional di Masjid Rahmatan Lil 'Alamin (RLA) Universitas Putra Indonesia YPTK Padang. Evaluasi ini mencakup beberapa aspek utama, yaitu keakuratan waktu, kehandalan komunikasi, kemudahan pengaturan, serta ketahanan sistem terhadap kondisi operasional harian.

Pertama, dari sisi keakuratan waktu, sistem yang telah dioptimasi mampu melakukan sinkronisasi otomatis dengan waktu global melalui koneksi internet menggunakan protokol NTP. Hasil pengujian menunjukkan bahwa waktu yang ditampilkan pada tampilan 7-segment sangat presisi dan konsisten dengan waktu resmi dari BMKG maupun server NTP lainnya. Hal ini mengatasi kendala pada sistem lama yang masih bergantung pada pengaturan waktu manual, yang rentan mengalami pergeseran waktu akibat kesalahan manusia atau gangguan teknis.

Kedua, keandalan komunikasi serial antara ESP32 dan Arduino Mega2560 dinilai cukup stabil. Pengujian dilakukan dengan mengirimkan data pengaturan

waktu dan jeda iqomah secara berkala dari antarmuka web ke kontroler, dan tidak ditemukan keterlambatan atau kehilangan data. Ini menunjukkan bahwa protokol komunikasi UART yang digunakan telah diimplementasikan dengan baik.

Dari segi kemudahan pengaturan, penggunaan antarmuka berbasis web memberikan fleksibilitas tinggi bagi pengurus masjid. Pengaturan jeda iqomah dan waktu dapat dilakukan dari *smartphone* tanpa harus berada di dekat perangkat. Hal ini menjadi nilai tambah signifikan dibandingkan sistem sebelumnya yang hanya bisa diatur secara lokal.

Selain itu, uji ketahanan sistem juga dilakukan dengan menjalankan perangkat selama lebih dari seminggu dalam kondisi operasional harian masjid. Sistem terbukti mampu bertahan dalam menghadapi gangguan seperti pemadaman listrik maupun pemutusan koneksi internet sementara. Setelah koneksi internet kembali normal, ESP32 secara otomatis melakukan resinkronisasi waktu tanpa perlu campur tangan pengguna.

Secara keseluruhan, sistem hasil optimasi menunjukkan peningkatan performa yang signifikan dari segi fungsionalitas, akurasi, efisiensi, dan kenyamanan operasional. Evaluasi ini memperkuat kesimpulan bahwa teknologi IoT yang diterapkan berhasil mengatasi keterbatasan sistem sebelumnya dan mampu mendukung manajemen waktu ibadah yang lebih baik di lingkungan masjid.

4.5 Pembahasan Hasil

Hasil implementasi sistem penjadwalan shalat digital berbasis ESP32 dan Arduino Mega2560 menunjukkan bahwa integrasi antara perangkat keras dan perangkat lunak dapat memberikan solusi yang efisien dan modern terhadap kebutuhan pengelolaan jadwal ibadah di masjid. Dengan adanya kemampuan konektivitas Wi-Fi pada ESP32, sistem dapat melakukan sinkronisasi waktu global secara otomatis, sehingga waktu shalat yang ditampilkan selalu akurat dan sesuai dengan standar waktu resmi.

Selain itu, pengaturan jeda iqomah melalui antarmuka web yang dapat diakses oleh *smartphone* mempermudah pengurus masjid dalam melakukan penyesuaian jadwal secara fleksibel, terutama saat terjadi perubahan jadwal

kegiatan seperti pengajian atau ceramah tambahan. Ini merupakan kemajuan signifikan dibandingkan sistem sebelumnya yang hanya mengandalkan tombol fisik, yang terbatas dari segi aksesibilitas dan efisiensi.

Dari hasil pengujian, sistem juga menunjukkan respons yang cepat dan stabil ketika menerima perintah melalui *web interface*. Hal ini menunjukkan bahwa komunikasi serial antara ESP32 dan Arduino berjalan lancar, serta pengolahan data pada sisi mikrokontroler dapat dilakukan dengan efektif. Tidak ditemukan delay signifikan antara input pengguna melalui *smartphone* dengan perubahan tampilan di perangkat, yang menandakan sistem ini dapat diandalkan untuk penggunaan harian.

Lebih jauh, dari segi operasional, sistem telah diuji dalam kondisi nyata selama lebih dari satu minggu dan tidak menunjukkan tanda-tanda penurunan performa, baik dalam aspek tampilan waktu, *countdown* iqomah, maupun kestabilan jaringan. Bahkan dalam situasi di mana koneksi internet sempat terputus, sistem tetap mampu mempertahankan waktu terakhir yang valid, dan melakukan resinkronisasi otomatis saat koneksi kembali tersedia.

Adanya fitur tambahan seperti penambahan waktu syuruq juga menjadi poin plus, karena memberikan informasi waktu ibadah yang lebih lengkap. Hal ini menunjukkan bahwa sistem hasil pengembangan ini tidak hanya memperbaiki kekurangan sistem lama, tetapi juga memperluas fungsionalitasnya untuk mendukung kenyamanan jamaah dan pengurus masjid.

Dengan hasil-hasil tersebut, sistem ini membuktikan bahwa penerapan teknologi IoT pada manajemen jadwal ibadah memiliki potensi besar untuk diadopsi lebih luas di masjid-masjid lain. Tingkat keterjangkauan perangkat dan kemudahan implementasinya juga memungkinkan sistem ini direplikasi secara luas dengan sedikit penyesuaian sesuai kebutuhan masing-masing masjid.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian dan pengujian yang telah dilakukan terhadap sistem optimasi penjadwalan sholat digital di Masjid Rahmatan Lil ‘Alamin (RLA) Universitas Putra Indonesia YPTK Padang, dapat ditarik beberapa kesimpulan utama sebagai berikut:

1. Sistem optimasi dengan integrasi modul ESP32 berhasil meningkatkan keakuratan waktu penjadwalan sholat melalui sinkronisasi otomatis dengan server NTP, sehingga menghilangkan kesalahan waktu yang sebelumnya sering terjadi pada sistem lama.
2. Penggunaan ESP32 yang terhubung ke jaringan Wi-Fi memungkinkan pengaturan jeda iqomah dan parameter jadwal sholat lainnya secara fleksibel melalui antarmuka web berbasis smartphone, memberikan kemudahan dan efisiensi bagi pengurus masjid.
3. Komunikasi serial UART antara Arduino Mega2560 dan ESP32 berjalan dengan stabil dan handal dalam mentransfer data pengaturan jadwal dan waktu, meskipun dalam kondisi gangguan sinyal masih mampu mempertahankan koneksi dalam batas toleransi.
4. Pengujian operasional di lingkungan masjid menunjukkan sistem dapat berfungsi dengan baik dalam kondisi nyata, dengan ketahanan yang memadai terhadap gangguan koneksi internet dan pemadaman listrik sementara.

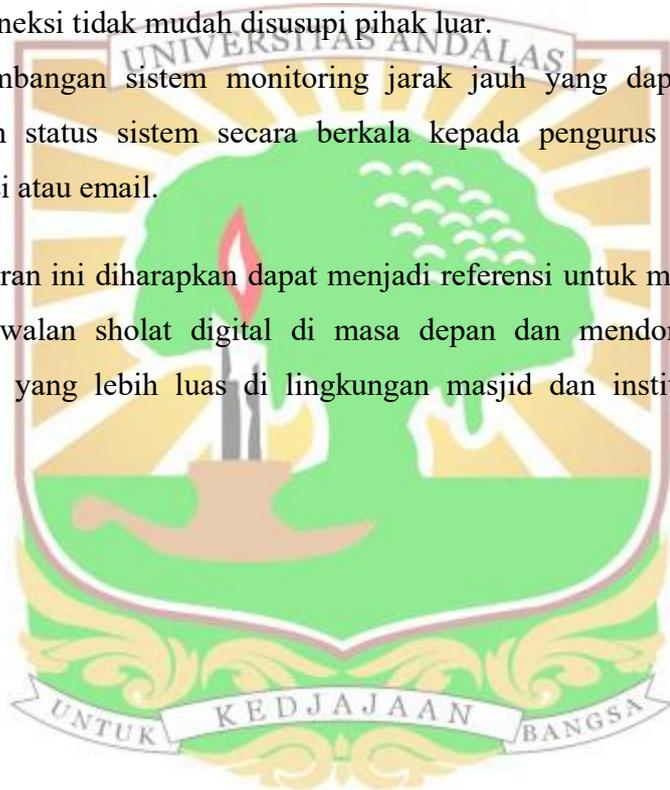
Secara keseluruhan, sistem yang diusulkan berhasil mengatasi kelemahan sistem lama dan memberikan solusi berbasis IoT yang modern dan praktis untuk penjadwalan sholat digital di masjid.

5.2 Saran

Berdasarkan pengalaman pengembangan dan pengujian sistem, terdapat beberapa saran yang dapat menjadi pertimbangan untuk pengembangan dan penelitian lanjutan :

1. Pengembangan antarmuka pengguna berbasis web agar lebih interaktif dan *user-friendly*, misalnya dengan aplikasi mobile native atau pemberitahuan notifikasi saat terjadi gangguan sinkronisasi waktu.
2. Penambahan fitur backup data dan konfigurasi agar pengaturan tidak hilang saat terjadi pemadaman listrik atau reset sistem.
3. Integrasi sensor lingkungan seperti sensor cahaya untuk otomatisasi kecerahan tampilan dan sensor suara untuk pengaturan suara iqomah secara otomatis.
4. Evaluasi lebih lanjut terhadap keamanan jaringan IoT agar data pengaturan dan koneksi tidak mudah disusupi pihak luar.
5. Pengembangan sistem monitoring jarak jauh yang dapat memberikan laporan status sistem secara berkala kepada pengurus masjid melalui aplikasi atau email.

Saran-saran ini diharapkan dapat menjadi referensi untuk menyempurnakan sistem penjadwalan sholat digital di masa depan dan mendorong penerapan teknologi IoT yang lebih luas di lingkungan masjid dan institusi keagamaan lainnya.



DAFTAR PUSTAKA

- [1] E. Naf'an, "Akurasi Sistem Penjadwalan Sholat Digital Menggunakan Arduino Sebagai Pengendali," *J. Sistim Inf. dan Teknol.*, vol. 1, no. 4, pp. 77–84, 2019.
- [2] Emil Naf'an, "PENGEMBANGAN SISTEM PENJADWALAN SHOLAT DIGITAL DENGAN PENGIRIMAN PESAN PADA RUNNING TEXT MELALUI KOMUNIKASI BLUETOOTH," *J. Teknoif Tek. Inform. Inst. Teknol. Padang*, vol. 9, no. 1, pp. 47–54, Apr. 2021.
- [3] O. Barybin, E. Zaitseva, and V. Brazhnyi, "Testing the Security ESP32 Internet of Things Devices," in *2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T)*, 2019, pp. 143–146.
- [4] M. Anusha, P. B. Kumar, V. Akhil, M. Gouthami, M. . Chinnaiah, and S. Shaik, "Internet of Things (IOT) based energy monitoring with ESP 32 and using Thingspeak," in *2024 10th International Conference on Communication and Signal Processing (ICCSP)*, 2024, pp. 1383–1387.
- [5] T.-C. Hou, L.-H. Liu, Y.-K. Lan, Y.-T. Chen, and Y.-S. Chu, "An Improved Network Time Protocol for Industrial Internet of Things," *Sensors*, vol. 22, no. 13, p. 5021, Jul. 2022.
- [6] T.Mangayarkarasi, Rajakumar.P, Sivabalan.M, and N. Gokul.R, "Arduino Based Smart Metering System with Smart 2560," in *2024 International Conference on Power, Energy, Control and Transmission Systems (ICPECTS)*, 2024, pp. 1–4.
- [7] L. A. S. I. Akbar, M. S. Iqbal, D. F. Budiman, A. S. Rachman, G. W. Wiriasto, and . S., "PEMANFAATAN RUNNING TEXT SEBAGAI ALAT BANTU INFORMASI WAKTU SHOLAT DI MASJID YAYASAN DARUL HIKMAH LOMBOK TENGAH," *J. Bakti Nusa*, vol. 2, no. 1, pp. 9–14, Feb. 2021.

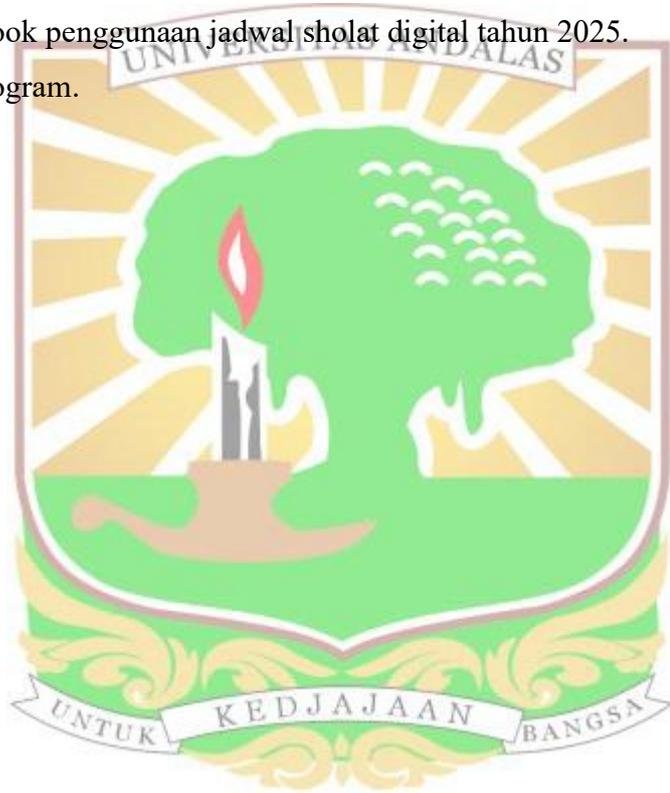
- [8] I. H. Kurniawan, L. Hayat, and D. N. K. Hardani, "Rancang Bangun Teknologi Penampil dan Pengingat Waktu Sholat Digital Di Lingkungan Pimpinan Cabang Muhammadiyah Baturaden," *J. Pengabd. Tek. dan Sains*, vol. 2, no. 01, Feb. 2022.
- [9] A. Drymonitis, "Introduction to Arduino," 2024, pp. 67–134.
- [10] Arduino, "ARDUINO MEGA 2560 REV3," 2019.
- [11] U. N. Malang, "Pengembangan Modul Pembelajaran Antarmuka dan Komunikasi Data Menggunakan Protokol MQTT dan ESP32," vol. 24, no. 3, pp. 392–407, 2024.
- [12] F. A. Aryatama and S. Samsugi, "Sistem Keamanan Kendaraan Bermotor Dengan ESP32 Menggunakan Kontrol Android," *SMATIKA J.*, vol. 14, no. 01, pp. 167–181, Jul. 2024.
- [13] Z. Didi and I. El Azami, "IoT, Comparative Study Between the Use of Arduino Uno, Esp32, and Raspberry pi in Greenhouses," 2022, pp. 718–726.
- [14] M. Sarosa *et al.*, "Air Cleaning System Based On The Internet Of Things (IoT)," in *2023 International Conference on Electrical and Information Technology (IEIT)*, 2023, pp. 367–371.
- [15] P. Bellini, P. Nesi, and G. Pantaleo, "IoT-Enabled Smart Cities: A Review of Concepts, Frameworks and Key Technologies," *Appl. Sci.*, vol. 12, no. 3, p. 1607, Feb. 2022.
- [16] R. Sonwane, A. Deshmukh, and S. Choudhary, "UART (Universal Asynchronous Receiver Transmitter) for Serial Data Communication: Design and Implementation on FPGA Platform," in *2023 2nd International Conference on Futuristic Technologies (INCOFT)*, 2023, pp. 1–4.
- [17] R. B. Chithra, S. A R, N. E. Mujassim, M. Gupta, and P. K. N, "Design and Implementation of UART With Effective Serial Communication," in *2024 Asia Pacific Conference on Innovation in Technology (APCIT)*, 2024, pp. 1–5.

- [18] I. H. Kurniawan, L. Hayat, and A. Fauzan, "Implementasi Teknologi Jadwal Waktu Sholat dan Media Informasi Digital Berbasis Mikrokontroler di Wilayah Pimpinan Ranting Muhammadiyah Klahang, Kecamatan Sokaraja, Kabupaten Banyumas," *J. Pengabd. Tek. dan Sains*, vol. 5, no. 1, p. 25, Jan. 2025.
- [19] Y. H. Kanoi, S. Abdussamad, and S. W. Dali, "Perancangan Jam Digital Waktu Sholat Menggunakan Arduino Uno," *Jambura J. Electr. Electron. Eng.*, vol. 1, no. 2, pp. 32–39, Oct. 2019.
- [20] W. Helma, H. Alam, J. W. Syafrawali, and R. . Bangun, "Rancang Bangun Running Text Led Display Jadwal Waktu Sholat Berbasis Arduino Uno Sebagai Media Informasi," *J. Electr. Technol.*, vol. 5, no. 2, pp. 2502–3624, 2020.
- [21] S. Amri, W. M. Faizal, A. Azizul, P. Almubarak, and N. Azima, "Implementasi Jadwal Shalat Digital Dengan Menggunakan Running Text Di Mushalla Kampus Politeknik Negeri Bengkalis," *Tanjak J. Pengabd. Kpd. Masy.*, vol. 5, no. 1, Jun. 2024.
- [22] S. Rosad, A. Yudhana, and A. Fadlil, "Jadwal Sholat Digital Menggunakan Metode Ephemeris Berdasarkan Titik Koordinat Smartphone," *IT J. Res. Dev.*, vol. 3, no. 2, pp. 30–43, Jan. 2019.
- [23] S. Sarifudin, M. Manshur, and A. Tirtana, "Penggunaan Komunikasi Bluetooth Pada Smartphone Android Untuk Pengiriman Data Pada Jam Digital Berbasis Arduino," *J. ELTIKOM*, vol. 1, no. 2, pp. 102–112, Jan. 2018.
- [24] A. W. Putra, R. Nuryanto, and A. Tafrikhatin, "Fitur Pengingat Kegiatan Masjid Dengan Kontrol Wi-Fi Berbasis ESP-32 Pada Jam Digital," *J. Pendidik. Tambusai*, vol. 5, no. 3, pp. 6177–6187, 2021.
- [25] M. Anton and M. H. Basri, "Perancangan+Jam+Istiwa+Otomatis+Menggunakan+Running+Text+dan+Speaker+Sebagai+Alat+Bantu+Waktu+Sholat+Di+Masjid+Nurul+Hidayah+Al-Taqwa," vol. 5, no. 2, pp. 43–48, 2020.

LAMPIRAN

Lampiran Penelitian ini terdiri dari:

1. Lampiran Bukti pelaksanaan pembuatan jadwal sholat digital tahun 2016.
2. Manual Book penggunaan jadwal sholat digital tahun 2016.
3. Bukti pelaksanaan pembuatan jadwal sholat digital tahun 2019.
4. Manual Book penggunaan jadwal sholat digital tahun 2019.
5. Bukti pelaksanaan pembuatan jadwal sholat digital tahun 2025.
6. Manual Book penggunaan jadwal sholat digital tahun 2025.
7. Listing Program.



1 1

Laporan Penelitian Insinyur - Emil Naf'an, Ph.D. - 2441612132 update 11 Juni 2025a.docx

 check 1 -- No Repository 032 check 1 Fuerza Aerea del Peru

Document Details

Submission ID

trn:oid::1:3324727333

Submission Date

Aug 30, 2025, 12:16 AM GMT-5

Download Date

Aug 30, 2025, 12:25 AM GMT-5

File Name

Laporan_Penelitian_Insinyur_-_Emil_Naf_an_Ph.D._-_2441612132_update_11_Juni_2025a.docx

File Size

7.3 MB

63 Pages**11,313 Words****70,666 Characters**

24% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography
-

Top Sources

- 23%  Internet sources
 - 7%  Publications
 - 14%  Submitted works (Student Papers)
-

Top Sources

- 23% Internet sources
- 7% Publications
- 14% Submitted works (Student Papers)

Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	Internet	
jsisfotek.org		4%
2	Internet	
docplayer.info		3%
3	Internet	
123dok.com		2%
4	Internet	
www.neliti.com		2%
5	Internet	
repository.its.ac.id		2%
6	Student papers	
Universitas Andalas		2%
7	Internet	
eprints.polsri.ac.id		1%
8	Internet	
www.scribd.com		<1%
9	Internet	
scholar.unand.ac.id		<1%
10	Internet	
www.arduinoindonesia.id		<1%
11	Internet	
www.jsisfotek.org		<1%

12	Internet	repository.ub.ac.id	<1%
13	Internet	etd.umy.ac.id	<1%
14	Internet	repository.binadarma.ac.id	<1%
15	Student papers	Universitas Muhammadiyah Sumatera Utara	<1%
16	Internet	teknoif.itp.ac.id	<1%
17	Student papers	Fakultas Teknik	<1%
18	Internet	repository.stiegici.ac.id	<1%
19	Internet	core.ac.uk	<1%
20	Internet	eprints.utdi.ac.id	<1%
21	Student papers	University of Balamand	<1%
22	Internet	repo.usni.ac.id	<1%
23	Internet	id.123dok.com	<1%
24	Internet	repository.uksw.edu	<1%
25	Internet	repository.unsri.ac.id	<1%

26	Publication	Ahmad Qashid Husaini, Herliyani Hasanah, Nurchim. "Pengendalian lampu dan ...	<1%
27	Internet	jurnal.polibatam.ac.id	<1%
28	Internet	jurnalnasional.ump.ac.id	<1%
29	Internet	repo.bunghatta.ac.id	<1%
30	Internet	repository.unisba.ac.id	<1%
31	Student papers	Telkom University	<1%
32	Student papers	Universitas Islam Riau	<1%
33	Student papers	Universitas Putera Indonesia YPTK Padang	<1%
34	Internet	eprints.umk.ac.id	<1%
35	Internet	guidevalpeline.com	<1%
36	Internet	repository.upi.edu	<1%
37	Internet	fliphtml5.com	<1%
38	Internet	repository.upbatam.ac.id	<1%
39	Internet	repository.usu.ac.id	<1%

40	Internet	vdocuments.site	<1%
41	Internet	eprints.undip.ac.id	<1%
42	Internet	fkip.upiypk.ac.id	<1%
43	Internet	www.arduino.biz.id	<1%
44	Internet	www.us.st.com	<1%
45	Internet	digilib.unila.ac.id	<1%
46	Internet	eclipse.itbu.ac.id	<1%
47	Internet	repository.polman-babel.ac.id	<1%
48	Internet	sir.stikom.edu	<1%
49	Internet	www.deutscher-apotheker-verlag.de	<1%
50	Internet	www.semanticscholar.org	<1%
51	Internet	www.tabengan.com	<1%
52	Internet	chip.co.id	<1%
53	Internet	dspace.uii.ac.id	<1%

54	Internet	dspace.ups.edu.ec	<1%
55	Internet	repository.poliupg.ac.id	<1%
56	Internet	repository.uinsu.ac.id	<1%
57	Internet	repository.usd.ac.id	<1%
58	Internet	wynnia.blogspot.com	<1%
59	Internet	eprints.umpo.ac.id	<1%

**OPTIMASI SISTEM PENJADWALAN SHOLAT DIGITAL BERBASIS
INTERNET OF THINGS (IoT) PADA MASJID RAHMATAN
LIL 'ALAMIN (RLA) UNIVERSITAS PUTRA
INDONESIA YPTK PADANG**

LAPORAN PENELITIAN

**EMIL NAF'AN
2441612132**

PEMBIMBING :

Prof. Ir. Nilda Tri Putri, M.T., Ph.D., IPU, ASEAN Eng.



**PROGRAM STUDI PENDIDIKAN PROFESI INSINYUR
SEKOLAH PASCASARJANA
UNIVERSITAS ANDALAS
2025**

**OPTIMASI SISTEM PENJADWALAN SHOLAT DIGITAL BERBASIS
INTERNET OF THINGS (IoT) PADA MASJID RAHMATAN
LIL 'ALAMIN (RLA) UNIVERSITAS PUTRA
INDONESIA YPTK PADANG**

**Emil Naf'an
2441612132**

LAPORAN PENELITIAN

**Sebagai Salah Satu Syarat untuk Memperoleh
Gelar Insinyur pada
Sekolah Pascasarjana
Universitas Andalas**

**PROGRAM STUDI PENDIDIKAN PROFESI INSINYUR
SEKOLAH PASCASARJANA
UNIVERSITAS ANDALAS
2025**

HALAMAN PERSETUJUAN

Judul Laporan Akhir : **OPTIMASI SISTEM PENJADWALAN SHOLAT DIGITAL BERBASIS *INTERNET OF THINGS* (IoT) PADA MASJID RAHMATAN LIL 'ALAMIN (RLA) UNIVERSITAS PUTRA INDONESIA YPTK PADANG**

Nama Mahasiswa : **Emil Naf'an**

Nomor Induk Mahasiswa : **2441612132**

Program Studi : **PENDIDIKAN PROFESI INSINYUR**

Laporan Penelitian telah diuji dan dipertahankan di depan sidang panitia ujian Profesi Insinyur pada Sekolah Pascasarjana Universitas Andalas dan dinyatakan lulus pada tanggal Juni 2025.

Menyetujui,

1. Pembimbing

2. Koordinator Program Studi

Prof. Ir. Nilda Tri Putri, M.T.,
Ph.D., IPU, ASEAN Eng.
NIP. 197707162003122003

Ir. Benny Dwika Leonanda, M.T., IPM,
ASEAN Eng.
NIP. 196608061994121000

3. Direktur Sekolah Pascasarjana
Universitas Andalas

Prof. apt. Henny Lucida, Ph.D.
NIP. 196701151991032002

HALAMAN PERSETUJUAN PEMBIMBING SEBAGAI PERSYARATAN UJIAN INSINYUR

Judul Laporan Penelitian : **OPTIMASI SISTEM PENJADWALAN
SHOLAT DIGITAL BERBASIS
INTERNET OF THINGS (IoT) PADA
MASJID RAHMATAN LIL 'ALAMIN
(RLA) UNIVERSITAS PUTRA
INDONESIA YPTK PADANG**

Nama Mahasiswa : Emil Naf'an

Nomor Induk Mahasiswa : 2441612132

Program Studi : Pendidikan Profesi Insinyur

Laporan Penelitian ini telah diperiksa dan dinyatakan telah memenuhi untuk mengikuti Ujian Profesi Insinyur pada Program Studi Pendidikan Profesi Insinyur, Sekolah Pascasarjana, Universitas Andalas.

Padang, Juni 2025

Dosen Pembimbing,

Prof. Ir. Nilda Tri Putri, M.T.,
Ph.D., IPU, ASEAN Eng.
NIP. 197707162003122003

*) Pilih salah satu

SURAT PERNYATAAN KEASLIAN TUGAS LAPORAN PENELITIAN

Saya yang bertanda tangan di bawah ini:

Nama : Emil Naf'an

NIM : 2441612132

Tempat Tgl Lahir : Air Bangis, 17 Desember 1974

Alamat : Komplek Fadilla Mandiri Gg. Sarai No. 25 RT.02/RW.12,
Kel. Banuaran Nan XX, Kec. Lubuk Begalung, Kota
Padang.

Dengan ini menyatakan bahwa Tugas Laporan Penelitian dengan judul 'OPTIMASI SSTEM PENJADWALAN SHOLAT DIGITAL BERBASIS *INTERNET OF THINGS* (IoT) PADA MASJID RAHMATAN LIL 'ALAMIN (RLA) UNIVERSITAS PUTRA INDONESIA YPTK PADANG' adalah hasil pekerjaan saya; dan seluruh ide, pendapat, atau materi dari sumber lain telah dikutip dengan cara penulisan referensi yang sesuai.

Pernyataan ini saya buat dengan sebenar-benarnya dan jika pernyataan ini tidak sesuai dengan kenyataan, maka saya bersedia menanggung sanksi yang akan dikenakan kepada saya termasuk pencabutan gelar Profesi Insinyur yang nanti saya dapatkan.

Padang, 06 Juni 2025

Yang Menyatakan



Emil Naf'an

ABSTRAK

Penelitian ini bertujuan untuk melakukan optimasi terhadap sistem jadwal sholat digital yang telah terpasang di Masjid Rahmatan Lil 'Alamin Universitas Putra Indonesia YPTK Padang. Sistem sebelumnya terdiri dari dua unit yang dikembangkan pada tahun 2016 dan 2019, masing-masing menggunakan Arduino Mega2560, RTC (DS1307), dan panel 7-segment common anoda yang dikendalikan melalui shift register 74HC595 dan driver ULN2003. Meskipun sistem ini telah berjalan dengan baik, terdapat dua kelemahan utama yang menjadi dasar perlunya optimasi, yaitu tidak adanya sinkronisasi waktu secara otomatis terhadap waktu global, serta pengaturan jeda iqomah yang masih bersifat manual dan kurang fleksibel.

Optimasi dilakukan dengan menambahkan modul ESP-32 sebagai pengendali berbasis *Internet of Things (IoT)* yang memungkinkan sistem terhubung ke jaringan WiFi. Modul ini bertugas menyinkronkan waktu secara otomatis melalui server NTP (*Network Time Protocol*), serta menyediakan antarmuka web sederhana yang dapat diakses melalui *smartphone* oleh pengurus masjid untuk mengatur jeda iqomah secara mudah dan cepat. Komunikasi antara ESP-32 dan Arduino Mega2560 dilakukan melalui protokol UART.

Hasil pengujian menunjukkan bahwa sistem baru mampu menjaga ketepatan waktu dengan deviasi kurang dari satu detik selama periode pengujian. Pengaturan jeda iqomah melalui antarmuka web dapat dilakukan dalam waktu kurang dari dua detik dengan tingkat keberhasilan 100%. Selain itu, pengurus masjid memberikan umpan balik yang positif terkait kemudahan penggunaan dan efisiensi waktu operasional. Dengan optimasi ini, sistem jadwal sholat digital menjadi lebih andal, fleksibel, dan sesuai dengan kebutuhan masjid modern yang mengutamakan efisiensi dan ketepatan waktu.

Kata Kunci: Jadwal Sholat Digital, IoT, ESP-32, Arduino, Sinkronisasi Waktu, Iqomah.

ABSTRACT

This research aims to optimize the digital prayer schedule system already installed at Masjid Rahmatan Lil 'Alamin Universitas Putra Indonesia YPTK Padang. The existing system consists of two units developed in 2016 and 2019, respectively. Both systems utilize Arduino Mega2560, RTC modules (DS1307), and 6-digit common anode 7-segment displays driven via 74HC595 shift registers and ULN2003 transistor arrays. While the system functioned reliably, it had two primary limitations: the lack of automatic synchronization with global time, and the manual, inflexible configuration of iqomah delay settings.

The optimization was achieved by integrating an ESP-32 module as an Internet of Things (IoT) controller, enabling the system to connect to local WiFi networks. The ESP-32 module performs automatic time synchronization via an NTP (Network Time Protocol) server and provides a simple web-based interface accessible via smartphone. Through this interface, mosque administrators can configure iqomah delays efficiently without the need for physical interaction with the hardware. Communication between the ESP-32 and Arduino Mega2560 is established through UART serial protocol.

Testing results indicate that the optimized system maintains time accuracy with a deviation of less than one second over the testing period. Iqomah delay adjustments via the web interface are completed in under two seconds with 100% success. Furthermore, feedback from mosque staff highlights improved usability and operational efficiency. With this optimization, the digital prayer schedule system becomes more reliable, flexible, and aligned with the modern needs of mosques that prioritize punctuality and ease of operation.

Keywords: Digital Prayer Schedule, IoT, ESP-32, Arduino, Time Synchronization, Iqomah.

59

DAFTAR ISI

34

14

HALAMAN PERSETUJUAN.....	iii
HALAMAN PERSETUJUAN PEMBIMBING SEBAGAI PERSYARATAN UJIAN INSINYUR.....	iv
SURAT PERNYATAAN KEASLIAN.....	v
ABSTRAK.....	vi
ABSTRACT.....	vii
DAFTAR ISI.....	viii
DAFTAR TABEL.....	x
DAFTAR GAMBAR.....	xi
DAFTAR SINGKATAN.....	xii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	5
1.3 Batasan Masalah.....	6
1.4 Tujuan Penelitian.....	7
1.5 Manfaat Penelitian.....	7
1.6 Sistematika Penulisan.....	8
BAB II TINJAUAN PUSTAKA.....	10
2.1 Jadwal Sholat Digital.....	10
2.2 Arduino Mega2560.....	11
2.3 Modul ESP32.....	15
2.4 Konsep Internet of Things (IoT).....	18
2.5 Komunikasi Serial UART.....	19
2.6 Penelitian Terkait.....	20

BAB III METODE PENELITIAN	25
3.1 Metode Penelitian	25
3.2 Studi Sistem Lama.....	27
3.3 Identifikasi Permasalahan.....	28
3.4 Perancangan Sistem Optimasi	29
3.5 Implementasi dan Pengujian Sistem.....	30
BAB IV HASIL DAN PEMBAHASAN	32
4.1 Implementasi Sistem.....	32
4.2 Dokumentasi Pengujian Alat	32
4.3 Analisis Perbandingan Sistem	40
4.4 Evaluasi Kinerja Sistem.....	42
4.5 Pembahasan Hasil.....	43
BAB V KESIMPULAN DAN SARAN	45
5.1 Kesimpulan	45
5.2 Saran	45
DAFTAR PUSTAKA	47
LAMPIRAN	50

DAFTAR TABEL

Tabel 2.1 Data Teknis Board Arduino Mega 2560	10
Tabel 2.2 Perbandingan Fitur ESP32 dengan Arduino Uno, Mega dan ES8266	15
Tabel 2.3 Perbandingan Fitur dari beberapa Penelitian Terdahulu.....	19
Tabel 3.1 Jadwal Pelaksanaan Penelitian.....	29
Tabel 4.1 Perbandingan Sistem Penjadwalan Sholat	34
Tabel 4.2 Pengujian Data Jadwal Sholat	37
Tabel 4.3 Pengujian <i>Buzzer</i>	37
Tabel 4.4 Perbandingan Sistem Penjadwalan Sholat	42

DAFTAR GAMBAR

Gambar 1.1 Jadwal Sholat Digital Masjid RLA Versi 1.0.....	1
Gambar 1.2 Jadwal Sholat Digital Masjid RLA Versi 2.0.....	2
Gambar 1.3 Jadwal Sholat Masjid RLA Versi 1.0 dan Versi 2.0	3
Gambar 1.4 Perbedaan waktu pada Jadwal Sholat Digital Masjid RLA Versi 1.0 dan Versi 2.0	4
Gambar 2.1 Tampilan Umum Jadwal Sholat Digital di Masjid.....	8
Gambar 2.2 Blok Diagram Sistem Jadwal Sholat Digital Konvensional.....	9
Gambar 2.3 Arduino Mega 2560	10
Gambar 2.4 Modul ESP32 Dev Board.....	14
Gambar 2.5 Diagram Komunikasi Serial UART antara ESP32 dan Arduino	18
Gambar 4.1 Tampilan sistem jadwal sholat tahun 2016	33
Gambar 4.2 Tampilan sistem jadwal sholat tahun 2019	33
Gambar 4.3 Pengujian Tampilan <i>Seven Segment</i> Waktu (jam, menit, detik)	35
Gambar 4.4 Pengujian Tampilan <i>Seven Segment</i> Tanggal / Hari	35
Gambar 4.5 Pengujian Tampilan <i>Seven Segment</i> Jadwal Sholat	36
Gambar 4.6 Tampilan Alamat IP Akses ESP32	38
Gambar 4.7 Tampilan Login ke Sistem	38
Gambar 4.8 Tampilan Dashboard Pengaturan Sinkronisasi Waktu NTP	39
Gambar 4.9 Tampilan Hasil Pengaturan Sinkronisasi Waktu NTP Pada Alat	39
Gambar 4.10 Tampilan Dashboard Pengaturan Sinkron Waktu Jeda Iqomah	40

DAFTAR SINGKATAN

GPS	<i>Global Positioning System</i>
IoT	<i>Internet of Things</i>
NTP	<i>Network Time Protocol</i>
RLA	Rahmaan Lil 'Alamin
RTC	<i>Real Time Clock</i>
UART	<i>Universal Asynchronous Receiver Transmitter</i>
USB	<i>Universal Serial Bus</i>

2

BAB I PENDAHULUAN

1.1 Latar Belakang

1 Sistem jadwal sholat digital telah menjadi perangkat penting dalam mendukung aktivitas ibadah di masjid. Salah satu fungsinya adalah menampilkan waktu sholat dengan akurat dan konsisten berdasarkan lokasi tertentu. Di Masjid Rahmatan Lil 'Alamin (RLA) Universitas Putra Indonesia YPTK Padang, telah terpasang dua unit alat jadwal sholat digital, masing-masing dibuat dan dipasang pada tahun 2016 dan 2019. Kedua alat tersebut telah beroperasi selama bertahun-tahun dan secara umum telah memberikan manfaat yang signifikan bagi jamaah masjid. Gambar 1.1 merupakan dokumentasi alat jadwal sholat yang pertama dipasang (Versi 1.0), yaitu pada tahun 2016. [1]



Gambar 1.1 Jadwal Sholat Digital Masjid RLA Versi 1.0

Waktu pemasangan alat ini dapat dilihat pada tanggal yang tertera pada alat jadwal sholat digital yang tertera pada alat jadwal sholat digital tersebut, yaitu tanggal 21 November 2016. Pada tahun 2016 tersebut masjid masih berukuran relatif kecil, sehingga hanya memerlukan satu buah jadwal sholat saja. Masjid ini terletak di dalam lokasi kampus bagian belakang, berdekatan dengan gedung A

Kampus UPI YPTK Padang. Masjid ini hanya dimanfaatkan oleh kalangan civitas akademika saja dan jarang dimanfaatkan oleh masyarakat umum. Ini disebabkan letaknya yang relatif jauh dari jalan raya, sehingga relatif sulit untuk diakses oleh masyarakat umum.

Seiring dengan perkembangan kampus Universitas Putra Indonesia, pada tahun 2019, didirikan Masjid yang baru dengan ukuran yang lebih besar dari sebelumnya. Masjid yang lama berubah fungsi menjadi tempat mahasiswa belajar mata kuliah agama. Masjid baru ini masih menggunakan nama sebelumnya. Masjid ini terletak masih di bagian depan kampus, di Jalan Raya Lubuk Begalung. Ini tentu saja memudahkan akses masyarakat luas, ditambah dengan sarana parkir yang memadai.

Sejalan dengan pemindahan masjid tersebut, maka jadwal sholat juga ikut berpindah ke lokasi yang baru. Posisi peletakan berada di sisi kiri mihrab masjid. Pihak Yayasan Perguruan Tinggi Komputer Padang (YPTK) minta dibuatkan lagi Jadwal sholat yang baru yang dilengkapi dengan *Running Text* dalam ukuran yang relatif besar, yaitu 1.3 x 3.5 meter. Gambar 1.2 merupakan dokumentasi alat jadwal sholat yang kedua (Versi 2.0), yaitu pada tahun 2019 [2].



Gambar 1.2 Jadwal Sholat Digital Masjid RLA Versi 2.0

Namun, berdasarkan pengalaman langsung dalam pengelolaan alat tersebut, terdapat beberapa kendala yang masih terjadi hingga saat ini. Salah satu kendala utama adalah proses penyesuaian waktu global yang masih dilakukan secara

manual, yaitu dengan menekan tombol-tombol fisik pada alat. Penyesuaian ini rutin dilakukan sekitar tiga bulan sekali, karena dalam periode tersebut alat mengalami deviasi waktu yang cukup signifikan akibat tidak tersinkron dengan waktu dunia secara otomatis.

Selain itu, hasil pengamatan teknis menunjukkan bahwa tingkat akurasi waktu pada masing-masing alat berbeda, meskipun keduanya berada di lokasi yang sama. Ketidakesesuaian ini berpotensi menimbulkan kebingungan di kalangan jamaah, terutama ketika terjadi perbedaan waktu azan beberapa detik antara dua alat dalam satu area masjid. Gambar 1.3 menunjukkan tampilan kedua alat jadwal sholat digital di Masjid Rahmatan Lil ‘Alamin (RLA) UPI YPTK Padang ini.



Gambar 1.3 Jadwal Sholat Masjid RLA Versi 1.0 dan Versi 2.0

Jika dilihat dari dekat, maka terdapat perbedaan waktu (dalam detik) antara alat jadwal sholat digital Versi 1.0 dan Versi 2.0. Perbedaan ini dapat dilihat pada gambar 1.4. Pada gambar 1.4 dapat dilihat bahwa alat Jadwal Sholat Digital Masjid RLA Versi 2.0 (a) menampilkan waktu **12:39:55**, sedangkan alat Jadwal Sholat Digital Masjid RLA Versi 1.0 (b) menampilkan waktu **12:40:04**. Waktu global saat itu adalah pukul **12:40:12**. Gambar 1.4 (c) adalah rincian dokumen pengambilan foto alat jadwal sholat tersebut.



Gambar 1.4 Perbedaan waktu pada Jadwal Sholat Digital Masjid RLA Versi 1.0 dan Versi 2.0.

Untuk mengatasi permasalahan tersebut, diperlukan sebuah solusi yang dapat mengoptimalkan sistem jadwal sholat digital agar lebih praktis, akurat, dan mudah dikelola. Salah satu pendekatan yang diusulkan adalah dengan mengembangkan sistem berbasis Internet of Things (IoT) menggunakan modul ESP32.[3][4] Melalui integrasi IoT ini, sinkronisasi waktu dapat dilakukan secara otomatis dengan mengakses waktu global melalui jaringan internet, yang kemudian dikendalikan dan dipantau menggunakan *smartphone*.

ESP32, sebagai modul IoT yang dilengkapi konektivitas Wi-Fi dan Bluetooth, akan bertindak sebagai penghubung antara aplikasi *smartphone* dan sistem kendali utama pada alat jadwal sholat. *Smartphone* akan menjalankan aplikasi khusus yang memungkinkan pengguna (admin atau petugas masjid) melakukan sinkronisasi waktu dan pengaturan jadwal sholat secara nirkabel. Waktu global yang diperoleh oleh ESP32 melalui protokol NTP (*Network Time Protocol*).[5] Waktu ini akan dikirimkan ke Arduino Mega2560, yang merupakan otak utama dari sistem jadwal sholat. Pengiriman dilakukan melalui komunikasi serial UART. Arduino Mega2560 kemudian akan memperbarui waktu internal dan menyesuaikan jadwal sholat berdasarkan informasi yang diterima.[6]

Dengan sistem ini, pengelolaan waktu tidak lagi membutuhkan penyesuaian manual, melainkan cukup dilakukan melalui aplikasi di *smartphone*. Selain meningkatkan keakuratan dan konsistensi waktu, pendekatan ini juga mempercepat proses pengaturan serta meminimalkan kesalahan akibat *human error*. Inovasi ini diharapkan dapat meningkatkan keandalan sistem jadwal sholat digital, sekaligus memberikan kenyamanan dan efisiensi bagi pengurus masjid dalam mengelola waktu ibadah secara modern dan terintegrasi.

1.2 Rumusan Masalah

1 Sistem jadwal sholat digital berbasis IoT di Masjid Rahmatan Lil 'Alamin (RLA) Universitas Putra Indonesia YPTK Padang memiliki beberapa permasalahan teknis utama. Rumusan masalah ini mencakup isu sinkronisasi waktu global, kendala teknis perangkat yang terpasang, kebutuhan pengaturan jeda iqomah per waktu sholat, serta integrasi antarmuka pengaturan melalui *smartphone*. Setiap aspek masalah dirumuskan dengan gaya formal-teknis sesuai standar rekayasa sistem. Sinkronisasi waktu global merupakan aspek krusial agar jadwal sholat dan panggilan adzan berjalan akurat. Dalam konteks IoT, sinkronisasi waktu sangat penting untuk menjaga agar seluruh perangkat beroperasi selaras. Protokol NTP (*Network Time Protocol*) umum digunakan untuk menyinkronkan jam dengan akurasi hingga milidetik, namun ketergantungan pada koneksi jaringan menimbulkan risiko gangguan. Penelitian sebelumnya mencatat bahwa apabila jaringan komunikasi bermasalah, fungsi sistem jadwal sholat digital dapat terhenti [1]. Oleh karena itu, diperlukan mekanisme sinkronisasi waktu yang andal beserta cadangan waktu lokal untuk menjaga keakuratan jadwal. Kendala teknis pada perangkat jadwal sholat digital yang terpasang juga perlu diidentifikasi. Sistem tradisional sering menggunakan MicroSD sebagai media penyimpanan jadwal [1]. Meskipun pendekatan ini dapat menampung data jadwal hingga lebih dari sepuluh tahun [1], hal tersebut membatasi fleksibilitas dalam memperbarui jadwal secara dinamis. Misalnya, perubahan jadwal baru hanya dapat diupdate secara manual jika ada perbedaan data. Selain itu, transfer data jadwal melalui jaringan Internet dilaporkan mengalami jeda waktu dan potensi kegagalan [1], yang menunjukkan perlunya peningkatan keandalan komunikasi dan penyimpanan. Keterbatasan

perangkat keras, seperti akurasi modul RTC, kapasitas tampilan, dan memori juga perlu dievaluasi untuk mendukung keandalan sistem secara keseluruhan. Kebutuhan pengaturan jeda iqomah per waktu sholat merupakan fitur penting yang harus diakomodasi. Iqomah adalah panggilan kedua sebelum sholat, biasanya dilakukan beberapa menit setelah adzan, dan interval waktu ini dapat berbeda pada tiap waktu sholat. Aplikasi jadwal sholat digital modern sudah menyediakan fitur “pengaturan jeda Iqomah per waktu sholat” [7], namun perangkat eksisting mungkin belum menerapkan kemampuan tersebut secara fleksibel. Kekurangan dalam mengatur jeda iqomah sesuai kebutuhan masjid dapat mengganggu kelancaran proses ibadah.

Dalam merancang dan merealisasikan sistem yang optimal, maka rumusan masalah dalam penelitian ini difokuskan untuk menjawab pertanyaan-pertanyaan berikut.

1. Bagaimana mekanisme sinkronisasi waktu global secara otomatis dapat diterapkan pada sistem jadwal sholat digital berbasis IoT di Masjid Rahmatan Lil ‘Alamin (RLA) Universitas Putra Indonesia YPTK Padang ?
2. Apa saja kendala teknis yang terjadi pada alat jadwal sholat digital yang telah dipasang sejak tahun 2016 dan 2019, serta bagaimana solusi teknis yang tepat untuk mengatasinya ?
3. Bagaimana sistem ini dapat menyediakan fitur pengaturan jeda iqomah yang fleksibel dan terpisah untuk setiap dari lima waktu sholat ?
4. Bagaimana perancangan integrasi antara *smartphone* dan sistem jadwal sholat digital dapat memudahkan petugas masjid dalam melakukan konfigurasi dan pemantauan ?

1.3 Batasan Masalah

Agar fokus penelitian tetap terarah dan terukur, maka penelitian ini dibatasi pada ruang lingkup berikut.

1. Penelitian ini hanya dilakukan pada dua unit alat jadwal sholat digital yang terpasang di Masjid Rahmatan Lil ‘Alamin (RLA) Universitas Putra Indonesia YPTK Padang, masing-masing dipasang pada tahun 2016 dan 2019.

2. Sinkronisasi waktu global dilakukan menggunakan modul ESP32 yang terhubung ke internet, dan waktu sinkronisasi diperoleh dari server waktu dunia (*NTP server*).
3. Komunikasi antara ESP32 dan Arduino Mega2560 menggunakan komunikasi serial UART, tanpa menggunakan protokol komunikasi kompleks lain seperti MQTT atau HTTP.
4. Sistem pengaturan jadwal sholat dan jeda iqomah dirancang agar dapat diatur melalui *web browser*.
5. Perancangan sistem hanya mencakup penyesuaian waktu sholat, pengaturan jeda iqomah, dan sinkronisasi waktu global, tidak mencakup pengaturan speaker adzan, integrasi dengan sensor lingkungan, atau fungsi multimedia lainnya.

1.4 Tujuan Penelitian

Penelitian ini bertujuan untuk :

1. Merancang dan mengimplementasikan sistem sinkronisasi waktu global otomatis pada perangkat jadwal sholat digital berbasis IoT.
2. Mengidentifikasi dan memperbaiki permasalahan teknis pada alat jadwal sholat yang telah terpasang sejak tahun 2016 dan 2019.
3. Mengembangkan fitur pengaturan jeda iqomah untuk lima waktu sholat harian, yang dapat diatur melalui aplikasi *smartphone*.
4. Mendesain sistem komunikasi serial antara ESP32 dan Arduino Mega2560 untuk mengirim data waktu dan konfigurasi jadwal.

1.5 Manfaat Penelitian

Berdasarkan hasil penelitian ini, diharapkan dapat memberikan beberapa manfaat sebagai berikut :

1. **Bagi Masjid** : Meningkatkan keakuratan dan efisiensi pengelolaan jadwal sholat digital tanpa perlu melakukan penyetelan manual secara rutin.

2. **Bagi Pengembang Alat** : Menjadi referensi dalam pengembangan sistem jadwal sholat digital berbasis IoT dengan pendekatan modular antara ESP32 dan Arduino Mega2560.
3. **Bagi Dunia Teknik** : Memberikan contoh implementasi rekayasa sistem berbasis mikrokontroler dan IoT yang aplikatif dalam konteks sosial keagamaan.
4. **Bagi laporan Profesi Insinyur** : Menunjukkan kemampuan profesional dalam merancang solusi teknis yang relevan, efisien, dan berorientasi pada kebutuhan pengguna lapangan.

1.6 Sistematika Penulisan

Sistematika penulisan berisi pembahasan apa yang akan ditulis disetiap Bab. Sistematika pada umumnya berupa paragraf yang setiap paragraf mencerminkan bahasan setiap Bab.

BAB I PENDAHULUAN

Bab ini terdiri atas sub-bab Latar Belakang, Rumusan Masalah, Batasan Masalah, Tujuan Penelitian, Manfaat Penelitian, dan Sistematika Penulisan.

BAB I TINJAUAN PUSTAKA

Bab ini memuat teori-teori yang mendukung, studi pustaka terkait sistem jadwal sholat digital, Arduino, ESP32, komunikasi serial.

BAB III METODOLOGI PENELITIAN

Bab ini Menjelaskan metode pengumpulan data, tahapan pengembangan sistem, desain komunikasi perangkat keras dan lunak, serta uji fungsional sistem.

BAB IV ANALISIS HASIL DAN PEMBAHASAN

Pada bab ini disajikan hasil implementasi alat dan aplikasi, serta analisis hasil pengujian sinkronisasi waktu dan pengaturan jadwal sholat.

BAB V PENUTUP

Bab ini menyimpulkan hasil penelitian dan memberikan saran untuk pengembangan sistem lebih lanjut.

BAB II

TINJAUAN PUSTAKA

2.1 Jadwal Sholat Digital

Jadwal sholat digital merupakan sistem elektronik yang dirancang untuk menampilkan waktu-waktu sholat secara otomatis sesuai dengan lokasi geografis tertentu. Sistem ini sangat membantu pengurus masjid dalam mengatur waktu ibadah dan memberikan informasi yang akurat kepada jamaah. Biasanya, jadwal sholat digital dilengkapi dengan fitur waktu adzan, iqomah, dan jam digital *real-time*. Namun, banyak dari perangkat ini masih bersifat statis, dan pengaturannya harus dilakukan secara manual melalui tombol pada perangkat keras.

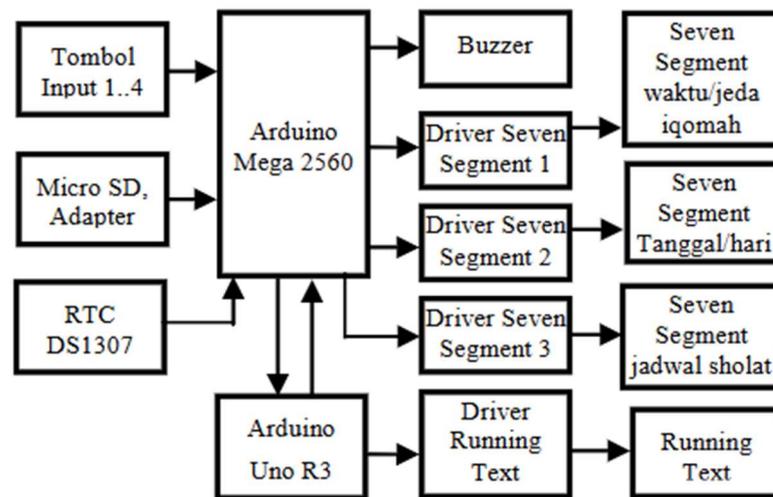


Gambar 2.1 Tampilan Umum Jadwal Sholat Digital di Masjid

Kelebihan dari sistem jadwal sholat digital adalah kemampuannya untuk memberikan informasi waktu ibadah secara visual dan konsisten, sehingga tidak diperlukan lagi penyampaian manual oleh pengurus masjid. Selain itu, tampilannya yang terang dan informatif menjadikannya lebih menarik dibandingkan papan manual atau jadwal cetak. Namun, perangkat ini juga memiliki keterbatasan, terutama dalam hal akurasi waktu jika tidak dilengkapi dengan sistem sinkronisasi otomatis. Deviasi waktu dapat terjadi setelah beberapa bulan penggunaan, terutama jika hanya bergantung pada jam internal (*Real Time Clock*).[8]

Dalam beberapa kasus, sistem ini juga tidak dilengkapi dengan antarmuka pengguna yang memadai untuk melakukan penyesuaian waktu atau jadwal,

sehingga pengguna yang kurang memahami teknis alat menjadi kesulitan dalam pengoperasiannya. Pengembangan teknologi jadwal sholat digital masa kini tidak hanya fokus pada fungsionalitas dasar, melainkan juga harus memperhatikan kemudahan operasional, keandalan waktu, dan fleksibilitas kontrol jarak jauh. Integrasi dengan teknologi mobile serta *Internet of Things* (IoT) menjadi sangat relevan untuk menjawab kebutuhan tersebut. Sistem yang dapat diatur dari perangkat *smartphone* oleh petugas masjid akan meningkatkan efisiensi dan kenyamanan dalam pengelolaan, serta meminimalkan potensi kesalahan akibat pengaturan manual yang tidak presisi. Gambar 2.2 berikut ini merupakan blok diagram sistem jadwal sholat digital konvensional. [1]



Gambar 2.2 Blok Diagram Sistem Jadwal Sholat Digital Konvensional

Lebih lanjut, sistem digital yang adaptif dan berbasis teknologi terkini dapat berkontribusi dalam meningkatkan kualitas pelayanan masjid kepada jamaah. Kehadiran sistem yang handal juga mencerminkan pengelolaan masjid yang modern, profesional, dan mengikuti perkembangan zaman, tanpa mengabaikan nilai-nilai tradisi dan kebutuhan umat.

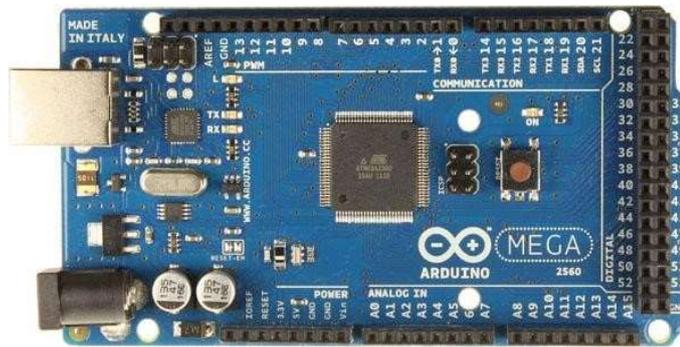
2.2 Arduino Mega2560

Arduino Mega2560 adalah papan pengembangan (*board development*) berbasis ATmega2560 yang memiliki kapasitas lebih besar dibanding varian

Arduino lainnya, baik dari sisi memori maupun jumlah pin input/output. Papan ini menyediakan 54 pin digital I/O, 16 pin input analog, dan 4 port komunikasi serial (UART), menjadikannya sangat ideal untuk proyek dengan banyak perangkat input dan output, seperti sistem jadwal sholat digital. Berikut ini beberapa data berkaitan dengan Arduino Mega2560.

1. Hardware Arduino Mega 2560

Komponen utama di dalam papan arduino adalah mikrokontroler 8 bit dengan merek ATmega yang dibuat oleh Atmel Corporation. Berbagai papan Arduino menggunakan tipe ATmega yang berbeda-beda tergantung dari spesifikasinya. Sebagai contoh Arduino uno menggunakan ATmega 328, sedangkan Arduino Mega 2560 yang lebih canggih menggunakan ATmega 2560. Bentuk dari Arduino Mega 2560 ditunjukkan oleh gambar 2.3 di bawah ini.[9]



Gambar 2.3 Arduino Mega 2560

Adapun data teknis board Arduino Mega2560 dapat dilihat pada tabel 2.1.[10]

Tabel 2.1 Data Teknis Board Arduino Mega 2560

Digital I/O Pins	54 (of which 15 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz

- 4
- 2
- Arduino Mega 2560 versi 1.0 memiliki fitur-fitur baru seperti berikut :
- a. Terdapat penambahan pin SDA dan SCL yang terletak dekat dengan pin AREF. Selain itu, dua pin baru lainnya ditempatkan di dekat pin RESET, salah satunya adalah IOREF. Pin IOREF memungkinkan *shield* untuk menyesuaikan diri dengan tegangan yang digunakan oleh papan (*board*). Di masa depan, *shield* diharapkan kompatibel baik dengan papan yang menggunakan mikrokontroler AVR (beroperasi pada 5 Volt) maupun dengan Arduino Due (beroperasi pada 3.3 Volt). Terdapat juga dua pin tambahan yang saat ini tidak terhubung, dan disediakan untuk keperluan pengembangan di masa mendatang.
 - b. Sirkuit *RESET*
 - c. *Chip* ATmega16U2 menggantikan *chip* ATmega8U2.

2. Sumber daya tegangan Arduino Mega 2560

Arduino Mega2560 dapat diaktifkan melalui koneksi USB atau dengan catu daya eksternal. Sumber daya dipilih secara otomatis. Sumber daya eksternal (non-USB) dapat berasal baik adaptor AC-DC atau baterai. Adaptor dapat dihubungkan dengan mencolokkan steker 2,1 mm yang bagian tengahnya terminal positif ke jack sumber tegangan pada papan. Jika tegangan berasal dari baterai dapat langsung dihubungkan melalui header pin Gnd dan pin Vin dari konektor Power. Papan Arduino mega 2560 dapat beroperasi dengan pasokan daya eksternal 6 volt sampai 20 volt. Jika diberi tegangan kurang dari 7 volt, maka pin 5 volt mungkin akan menghasilkan tegangan kurang dari 5 volt dan ini akan membuat papan menjadi tidak stabil. Jika sumber tegangan menggunakan lebih dari 12 volt, regulator tegangan akan mengalami panas berlebihan dan bisa merusak papan. Rentang sumber tegangan yang dianjurkan adalah 7 volt sampai 12 volt.

Pin tegangan yang tersedia papan Arduino adalah sebagai berikut :

- 4
- a. VIN : adalah input tegangan untuk papan Arduino ketika menggunakan sumber daya eksternal. Input tegangan dapat diberikan tegangan melalui pin ini. Jika memasok tegangan untuk papan melalui *jack power*, kita bisa mengakses/mengambil tegangan melalui pin ini.

- b. 5V : sebuah pin yang mengeluarkan tegangan teregulasi 5 volt. Dari pin ini tegangan sudah diatur dari regulator yang tersedia (*built-in*) pada papan. Arduino dapat diaktifkan dengan sumber daya baik berasal dari *jack power* DC (7-12 volt), konektor USB (5 volt), atau pin VIN pada board (7-12 volt). Jangan memberikan tegangan melalui pin 5V atau 3.3V secara langsung tanpa melewati regulator, karena dapat merusak papan Arduino.
- c. 3V3 : sebuah pin yang menghasilkan tegangan 3.3 volt. Tegangan ini dihasilkan oleh regulator yang terdapat pada papan (on-board). Arus maksimum yang dihasilkan adalah 50 mA.
- d. GND : pin ground atau massa.
- e. IOREF : pin ini pada papan Arduino berfungsi untuk memberikan referensi tegangan yang beroperasi pada mikrokontroler.

3. Memori Arduino Mega2560

Arduino ATmega2560 memiliki 256 KB *flash* memori untuk menyimpan kode (yang 8 KB digunakan untuk *bootloader*), 8 KB SRAM dan 4 KB EEPROM (yang dapat dibaca dan ditulis dengan perpustakaan EEPROM).

4. Software IDE Arduino

Arduino diciptakan untuk para pemula bahkan yang tidak memiliki dasar bahasa pemrograman sama sekali. Arduino menggunakan bahasa C++ yang telah dipermudah melalui *library*. *Software* Arduino ini dapat di-*install* di berbagai *operating system* (OS) seperti: LINUX, Mac OS, Windows. Arduino tidak hanya sekedar sebuah alat pengembangan, tetapi kombinasi dari *hardware*, bahasa pemrograman dan *Integrated Development Environment* (IDE) yang canggih. IDE adalah sebuah *software* yang sangat berperan untuk menulis program, meng-*compile* menjadi kode biner dan meng-*upload* ke dalam memory mikrokontroler. *Software IDE* Arduino terdiri dari 3 (tiga) bagian:

- a. Editor program, untuk menulis dan mengedit program dalam bahasa *processing*. Listing program pada Arduino disebut *sketch*.

- b. *Compiler*, modul yang berfungsi mengubah bahasa *processing* (kode program) ke dalam kode biner karena kode biner adalah satu-satunya bahasa program yang dipahami oleh mikrokontroler.
- c. *Uploader*, modul yang berfungsi memasukkan kode biner kedalam memori mikrokontroler.

5. Konfigurasi pin Arduino Mega2560

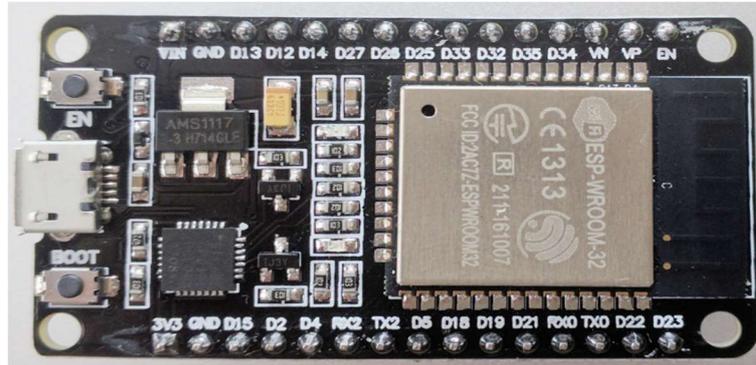
Konfigurasi pin ATmega 2560 adalah sebagai berikut:

- a. VCC merupakan pin yang digunakan sebagai masukan sumber tegangan.
- b. GND merupakan pin untuk Ground.
- c. XTAL1/ XTAL2, XTAL digunakan sebagai pin external clock.
- d. Port A, B, C, D, E, H, dan L merupakan 8 bit port I/O dengan internal *pull-up* resistor. Port G merupakan 6 bit port I/O dengan internal *pull up* resistor.
- e. Port F (PF0:PF7) dan Port K (PK0:PK7) merupakan pin I/O dan merupakan pin masukan ADC.
- f. AVCC adalah pin masukan untuk tegangan ADC.
- g. AREF adalah pin masukan untuk tegangan referensi eksternal.

Dalam penelitian ini, Arduino Mega2560 berperan sebagai unit pengolah utama (*main controller*) yang mengatur logika waktu sholat, waktu jeda iqomah, serta tampilan output ke panel display LED. Arduino Mega2560 ini juga melakukan proses pembacaan dan pengeksekusian perintah dari modul komunikasi eksternal, termasuk data yang dikirim dari modul ESP32.

2.3 Modul ESP32

ESP32 adalah sebuah mikrokontroler berdaya tinggi yang dikembangkan oleh *Espressif Systems*, merupakan penerus dari ESP8266.[11] Mikrokontroler ini dilengkapi dengan Wi-Fi dan Bluetooth secara *onboard*, menjadikannya sangat cocok untuk proyek Internet of Things (IoT). ESP32 memiliki arsitektur dual-core Tensilica Xtensa LX6, dengan kecepatan hingga 240 MHz, serta RAM sebesar 520 KB dan kapasitas penyimpanan flash hingga beberapa megabyte tergantung modulnya.



Gambar 2.4 Modul ESP32 Dev Board

Kemampuannya dalam menangani berbagai protokol komunikasi seperti UART, SPI, I2C, PWM, ADC, DAC, serta integrasi nirkabel Wi-Fi dan Bluetooth menjadikan ESP32 sangat fleksibel dalam pengembangan sistem embedded dan perangkat pintar.[12]

Beberapa fitur utama ESP32 yang menjadi keunggulannya antara lain:

- Dual Core Processor: Memungkinkan multitasking secara efisien.
- Wi-Fi 802.11 b/g/n: Terintegrasi langsung untuk konektivitas internet.
- Bluetooth v4.2 + BLE: Untuk komunikasi jarak dekat seperti pengendalian smartphone.
- GPIO hingga 34 pin: Dapat dikonfigurasi sebagai input/output, PWM, atau fungsi khusus lainnya.
- Analog Input (ADC) hingga 18 channel dan 2 channel DAC.
- Protokol komunikasi: UART, SPI, I2C, CAN, IR, dan PWM.
- Sensor internal: Sensor suhu, hall sensor.
- *Sleep Mode* hemat daya: Cocok untuk aplikasi IoT yang membutuhkan efisiensi energi.
- Kemampuan *Over-The-Air* (OTA): Pemrograman ulang firmware melalui jaringan Wi-Fi.

Berikut beberapa keunggulan ESP32 dibandingkan dengan Arduino Uno, Mega, dan bahkan pendahulunya ESP8266.[13]

Tabel 2.2 Perbandingan Fitur ESP32 dengan Arduino Uno, Mega dan ES8266

Fitur	ESP32	ESP8266	Arduino Uno/Mega
Prosesor	Dual-Core 240 MHz	Single-Core 80 MHz	AVR 16 MHz
Wi-Fi	Ada (integrated)	Ada	Tidak ada
Bluetooth	Ada (Classic + BLE)	Tidak ada	Tidak ada
ADC	Hingga 18 channel	1 channel	6–16 channel
GPIO	>30	±17	±20–50
Memori	520 KB RAM	160 KB RAM	2 KB (Uno), 8 KB (Mega)
Harga	Relatif murah	Murah	Bervariasi

Agar ESP32 dapat diprogram menggunakan Arduino IDE, perlu ditambahkan *board manager* khusus. Berikut adalah langkah-langkah lengkapnya:

Langkah 1: Install Arduino IDE

Pastikan Anda sudah menginstal Arduino IDE versi terbaru.

Disarankan minimal versi 1.8.10 ke atas atau menggunakan Arduino IDE 2.0.

Langkah 2: Menambahkan URL Board Manager

1. Buka Arduino IDE.
2. Klik menu File > Preferences.
3. Pada kolom Additional Board Manager URLs, masukkan URL berikut:

https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json

Jika sudah ada URL lain, pisahkan dengan koma.

Langkah 3: Install Board ESP32

1. Klik menu Tools > Board > Boards Manager.
2. Ketik esp32 di kolom pencarian.

35

3. Pilih `esp32` by Espressif Systems, lalu klik Install. Proses ini akan memakan waktu beberapa menit tergantung kecepatan internet Anda.

Langkah 4: Pilih Board dan Port

10

1. Setelah instalasi selesai, klik Tools > Board, lalu pilih jenis board Anda, misalnya :
 - ESP32 Dev Module
 - NodeMCU-32S
2. Sambungkan ESP32 ke komputer menggunakan kabel USB.
3. Pilih port yang sesuai di menu Tools > Port.

Langkah 5: Upload Sketch

1. Coba upload program sederhana seperti Blink untuk memastikan ESP32 berfungsi.
2. Jika saat upload muncul error Failed to connect, tekan dan tahan tombol BOOT pada ESP32 selama proses upload, lalu lepas setelah muncul "Connecting...".

Dengan langkah ini, ESP32 sudah siap digunakan pada berbagai proyek berbasis Arduino IDE, termasuk sebagai modul Wi-Fi yang berperan dalam sinkronisasi waktu dan kendali berbasis web pada sistem jadwal sholat digital yang sedang diteliti.

3

2.4 Konsep Internet of Things (IoT)

Internet of Things atau dikenal juga dengan singkatan IoT, merupakan sebuah konsep yang bertujuan untuk memperluas manfaat dari konektivitas internet yang tersambung secara terus-menerus. Adapun kemampuan seperti berbagi data, remote control, dan sebagainya, termasuk juga pada benda di dunia nyata. Contohnya bahan pangan, elektronik, koleksi, peralatan apa saja, termasuk benda hidup yang semuanya tersambung ke jaringan lokal dan global melalui sensor yang tertanam dan selalu aktif.

Makna serupa yang lain, *Internet of Things* (IoT) adalah sebuah konsep/skenario dimana suatu objek yang memiliki kemampuan untuk mentransfer data melalui jaringan tanpa memerlukan interaksi manusia ke manusia atau manusia ke komputer [14]. Istilah IoT (*Internet of Things*) mulai dikenal tahun 1999 yang saat itu disebutkan pertama kalinya dalam sebuah presentasi oleh Kevin Ashton, cofounder and executive director of the Auto-ID Center di MIT.

Dengan semakin berkembangnya infrastruktur internet, maka kita menuju babak berikutnya, di mana bukan hanya smartphone atau komputer saja yang dapat terkoneksi dengan internet. Namun berbagai macam benda nyata akan terkoneksi dengan internet. Sebagai contohnya dapat berupa : mesin produksi, mobil, peralatan elektronik, peralatan yang dapat dikenakan manusia (*wearables*), dan termasuk benda nyata apa saja yang semuanya tersambung ke jaringan lokal dan global menggunakan sensor dan atau aktuator yang tertanam.

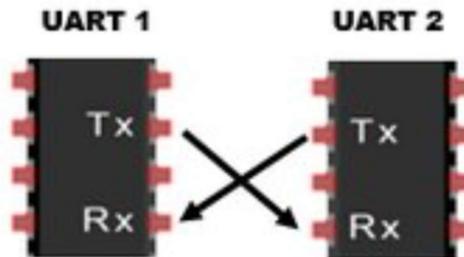
31 Dengan demikian dapat disimpulkan bahwa konsep *Internet of Things* (IoT) sendiri mengacu pada kemampuan perangkat fisik untuk saling terhubung melalui jaringan internet, memungkinkan pertukaran data dan kendali dari jarak jauh [15]. Dalam penelitian ini, ESP32 digunakan untuk melakukan sinkronisasi waktu otomatis dari NTP (*Network Time Protocol*) server dan mengirimkan data waktu terkini ke Arduino Mega2560 melalui komunikasi serial UART. Modul ESP32 juga menerima perintah dari *smartphone* petugas masjid, sehingga segala pengaturan bisa dilakukan secara nirkabel tanpa harus menyentuh alat secara langsung.

Dengan penerapan IoT, jadwal sholat digital menjadi sistem yang adaptif, modern, dan minim perawatan. Petugas masjid tidak perlu lagi menyesuaikan waktu secara manual, melainkan cukup melakukan sinkronisasi melalui koneksi Wi-Fi yang terhubung ke server waktu global.

2.5 Komunikasi Serial UART

48 UART (*Universal Asynchronous Receiver Transmitter*) adalah metode komunikasi serial asinkron yang umum digunakan untuk menghubungkan dua perangkat mikrokontroler [16]. Protokol ini menggunakan dua jalur utama, yaitu

TX (*Transmit*) dan RX (*Receive*), untuk mengirimkan data secara dua arah. UART tidak memerlukan clock eksternal seperti SPI atau I2C, sehingga lebih sederhana dalam implementasi perangkat keras [17].



Gambar 2.5 Diagram Komunikasi Serial UART antara ESP32 dan Arduino

10 UART merupakan protokol komunikasi serial yang memungkinkan perangkat untuk mengirim dan menerima data per bit secara asinkron. Artinya, data dikirimkan dalam bentuk urutan bit tanpa sinyal clock eksternal. Setiap perangkat memiliki tempat penyimpanan sementara (*buffer*) untuk menampung data yang dikirim dan diterima. Kecepatan komunikasi ditentukan oleh *baud rate* yang mewakili jumlah bit data yang dikirim dalam satu detik.

Dalam sistem ini, UART digunakan sebagai saluran komunikasi antara ESP32 dan Arduino Mega2560. Data hasil sinkronisasi waktu global dari ESP32 akan dikirim ke Arduino melalui jalur ini. Arduino kemudian membaca data tersebut dan memperbarui waktu sistem internalnya. Keandalan komunikasi serial ini sangat penting untuk menjaga akurasi sistem jadwal sholat digital dan menjamin bahwa data yang ditampilkan sesuai dengan waktu aktual.

2.6 Penelitian Terkait

12 Berbagai penelitian sebelumnya telah mengkaji pengembangan sistem jadwal sholat otomatis dengan beragam pendekatan teknologi. Salah satu yang paling banyak digunakan adalah mikrokontroler Arduino Uno dengan dukungan modul RTC (*Real Time Clock*), seperti yang digunakan oleh Emil Naf'an pada tahun 2019 dan 2021. Sistem tersebut mampu menampilkan jadwal sholat secara otomatis dengan tampilan visual berupa 7-segment display atau running text, yang

dikombinasikan dengan penyimpanan data menggunakan microSD. Akurasi sistem cukup baik dan stabil dalam pengujian jangka panjang. Namun, pengaturan waktu masih dilakukan secara manual dan sistem belum terkoneksi ke jaringan internet. [1] [2] [18]

Tabel 2.3 Perbandingan Fitur dari beberapa Penelitian Terdahulu

No	Penulis & Tahun	Platform	Fitur Utama	Kelebihan	Kelemahan
1	Emil Naf'an (2019) [1]	Arduino Uno, RTC, MicroSD, 7-Segment, Running Text	Menampilkan jadwal sholat dan aktivitas dosen secara otomatis	Akurasi tinggi, pengujian 7 hari non-stop, tampilan jelas	Tidak terhubung ke internet, pengaturan awal manual
2	Emil Naf'an (2021) [2]	Arduino, <i>Seven Segment</i> , Bluetooth	Pengiriman pesan jadwal sholat melalui running text via Bluetooth	Komunikasi nirkabel, mudah dioperasikan	Jangkauan terbatas, tidak terhubung ke internet
3	Yusuf H. Kanoi et al. (2023) [19]	Arduino Uno, RTC DS3231, LED Dot Matrix P10	Menampilkan jadwal sholat lima waktu secara digital	Tampilan jelas dan mudah dibaca, desain sederhana	Tidak terhubung ke internet, pembaruan jadwal manual
4	Helma Widya et al. (2020) [20]	Arduino Uno, LED Display, Bluetooth	Running text jadwal sholat yang dapat diperbarui melalui ponsel Android	Tampilan fleksibel, kontrol mudah	Keterbatasan jarak komunikasi Bluetooth
5	Syaiful Amri et al. (2023) [21]	Arduino, LED Display	Menampilkan jadwal waktu sholat secara otomatis di	Informasi akurat, meningkatkan kenyamanan jamaah	Tidak disebutkan konektivitas internet

			mushalla kampus		
6	Itmi Hidayat Kurniawan et al. (2022) [18]	Mikrokontroler, RTC, Display Digital	Penampil dan pengingat waktu sholat digital dengan waktu iqomat	Real-time, meningkatkan kesadaran waktu sholat	Tidak disebutkan konektivitas internet
7	Safiq Rosad, Anton Yudhana, Abdul Fadlil (2019) [22]	ATmega328, GPS Smartphone, 7-Segment Display	Perhitungan waktu sholat menggunakan metode ephemeris berdasarkan koordinat GPS dari smartphone	Akurasi tinggi sesuai lokasi pengguna, tidak bergantung pada koneksi internet	Tidak memiliki fitur kendali jarak jauh atau integrasi dengan aplikasi mobile
8	Sarifudin, Manshur, Angga Tirtana (2018) [23]	Arduino Uno, RTC DS3231, Dot Matrix P4, Bluetooth HC-06, Aplikasi Android (App Inventor)	Menampilkan jadwal sholat, jam digital, dan tanggal yang dapat diperbarui melalui koneksi Bluetooth dari smartphone Android	Tampilan besar dan jelas, pengaturan waktu fleksibel melalui aplikasi Android, tidak memerlukan koneksi internet	Tidak mendukung kontrol jarak jauh melalui internet, pengaturan awal memerlukan koneksi Bluetooth manual
9	Andhika Wahyu Putra, Rizki Nuryanto, Asni Tafrikhatin (2021) [24]	ESP32, LED Display, Speaker, Web Interface	Jam digital dengan pengingat kegiatan masjid yang dapat dikontrol melalui Wi-Fi menggunakan smartphone	Pengaturan mudah melalui browser, fitur pengingat kegiatan dengan output suara, tampilan LED yang dapat disesuaikan	Jangkauan Wi-Fi terbatas (maksimal 20 meter tanpa halangan), tidak mendukung kontrol jarak jauh melalui internet

10	M. Anton, M.H. Basri, Y. Prasetyo (2020) [25]	Arduino Uno, RTC, Running Text, Speaker USB	Jam istisya otomatis dengan tampilan teks berjalan dan suara azan	Tampilan waktu sholat yang jelas, pengingat audio otomatis, sistem sederhana dan hemat daya	Tidak terhubung ke internet, pengaturan awal manual, tidak mendukung kontrol jarak jauh
----	---	---	---	---	---

Seiring perkembangan teknologi, mulai banyak penelitian yang mengintegrasikan sistem dengan komunikasi nirkabel, seperti *Bluetooth*, yang memudahkan pengaturan dari *smartphone* Android. Helma Widya dan timnya mengembangkan sistem dengan fitur pembaruan jadwal sholat melalui ponsel, memberikan fleksibilitas namun terbatas pada jangkauan komunikasi Bluetooth yang pendek. Hal serupa juga ditemukan pada penelitian Sarifudin dan rekan-rekannya yang menggunakan Bluetooth HC-06 dan aplikasi Android sederhana berbasis App Inventor, namun tanpa dukungan akses jarak jauh melalui internet.

Penelitian lain oleh Yusuf H. Kanoi (2023) menggunakan LED Dot Matrix P10 untuk menampilkan jadwal sholat secara digital, menghasilkan tampilan yang jelas dan informatif, namun masih bergantung pada pembaruan manual. Sementara itu, penelitian oleh Safiq Rosad dan tim menawarkan pendekatan yang lebih akurat dengan menggunakan metode ephemeris berbasis GPS dari *smartphone*, memungkinkan sistem menyesuaikan waktu sholat berdasarkan lokasi pengguna tanpa perlu koneksi internet. Meskipun demikian, sistem ini belum mengintegrasikan kontrol atau pemantauan jarak jauh.

Pendekatan berbasis Internet of Things (IoT) mulai muncul dalam penelitian terbaru, seperti yang dilakukan oleh Andhika Wahyu Putra dan tim pada tahun 2021. Mereka mengembangkan sistem berbasis ESP32 yang dapat dikendalikan melalui web interface menggunakan koneksi Wi-Fi, lengkap dengan fitur pengingat kegiatan masjid berupa output suara melalui speaker. Sistem ini menawarkan fleksibilitas tinggi dan dapat dikontrol melalui browser di *smartphone*, meskipun jangkauan Wi-Fi menjadi batasan utama.

Sistem jam istisya otomatis juga dikembangkan oleh M. Anton dan rekan-rekannya pada tahun 2020. Mereka memanfaatkan kombinasi Arduino, RTC, dan

tampilan running text, serta audio azan melalui speaker USB. Sistem ini sederhana, hemat daya, dan efektif untuk digunakan di masjid, namun tidak mendukung akses kendali jarak jauh ataupun fitur berbasis internet.

Dari analisis terhadap sepuluh penelitian ini, terlihat bahwa sebagian besar sistem masih bersifat lokal dan belum sepenuhnya memanfaatkan potensi IoT untuk manajemen jarak jauh. Meskipun tampilan dan akurasi waktu terus mengalami peningkatan, pengaturan jadwal dan kontrol sistem masih memerlukan intervensi manual atau koneksi terbatas seperti *Bluetooth*. Oleh karena itu, integrasi ESP32 dengan kemampuan koneksi Wi-Fi dan kontrol melalui aplikasi mobile menjadi peluang yang menjanjikan untuk pengembangan sistem yang lebih fleksibel, efisien, dan sesuai dengan kebutuhan modern pengelolaan masjid. Inovasi ini tidak hanya meningkatkan keakuratan dan kenyamanan, tetapi juga memungkinkan pengurus masjid melakukan pengaturan tanpa perlu berada di lokasi secara langsung. Dalam konteks ini, penelitian yang dilakukan di Masjid Rahmatan Lil ‘Alamin (RLA) Universitas Putra Indonesia YPTK Padang memberikan kontribusi baru dengan menerapkan pendekatan IoT secara menyeluruh. Sistem tidak hanya menyinkronkan waktu secara otomatis melalui internet, tetapi juga memungkinkan pengaturan jeda iqomah dan manajemen akses pengguna langsung dari perangkat *mobile*.

2

BAB III METODOLOGI PENELITIAN

3.1 Kerangka Penelitian

Kerangka penelitian ini mencakup beberapa aspek penting yang disusun secara sistematis untuk mencapai tujuan optimasi sistem penjadwalan sholat digital. Proses penelitian dilakukan melalui beberapa tahapan mulai dari studi awal sistem eksisting, identifikasi permasalahan, pengumpulan data teknis, perancangan sistem baru, implementasi optimasi, hingga pengujian dan evaluasi sistem yang telah diperbarui.

12



12

Gambar 3.1. Kerangka Penelitian

2

Berikut adalah poin-poin utama dalam kerangka penelitian ini:

1. Tujuan Penelitian

Penelitian ini bertujuan untuk mengoptimasi sistem penjadwalan sholat digital yang telah terpasang di Masjid Rahmatan Lil 'Alamin (RLA) Universitas Putra

1

Indonesia YPTK Padang dengan menambahkan fitur berbasis *Internet of Things* (IoT), seperti sinkronisasi waktu global dan pengaturan jeda iqomah melalui *smartphone*.

2. Lokasi Implementasi

Lokasi utama dari penelitian ini adalah Masjid Rahmatan Lil ‘Alamin (RLA) Universitas Putra Indonesia YPTK Padang, tempat di mana dua sistem jadwal sholat telah dipasang sebelumnya (tahun 2016 dan 2019). Lokasi ini menjadi tempat pengujian langsung terhadap sistem hasil optimasi.

3. Tahapan Kegiatan

Proses penelitian terdiri dari beberapa tahapan, yaitu studi sistem lama, identifikasi permasalahan, pengumpulan data teknis, perancangan sistem baru, implementasi, dan pengujian sistem. Setiap tahapan dirancang untuk memberikan umpan balik terhadap keandalan dan efektivitas sistem yang dikembangkan.

4. Metode Evaluasi

Evaluasi dilakukan berdasarkan perbandingan antara sistem lama dan sistem yang telah dioptimasi. Kriteria evaluasi mencakup akurasi sinkronisasi waktu, kemudahan pengaturan jeda iqomah, kestabilan komunikasi antar perangkat, serta keandalan sistem dalam operasional harian masjid.

Dengan kerangka ini, diharapkan hasil penelitian mampu memberikan kontribusi nyata dalam meningkatkan efektivitas pengelolaan jadwal sholat di lingkungan masjid melalui pemanfaatan teknologi terkini. ini dilaksanakan untuk mengoptimasi sistem penjadwalan sholat digital pada Masjid Rahmatan Lil ‘Alamin (RLA) Universitas Putra Indonesia YPTK Padang yang sebelumnya telah dipasang pada tahun 2016 dan 2019. Penelitian ini bertujuan untuk mengidentifikasi kelemahan sistem lama dan menerapkan solusi berbasis IoT, khususnya dengan penambahan modul ESP32 sebagai penghubung sistem dengan perangkat *smartphone*.

3.2 Studi Sistem Lama

Untuk memperjelas perbedaan antara kedua sistem tersebut, berikut adalah fitur utama yang membedakan keduanya.

1. Sistem Tahun 2016:

- Kontrol tampilan menggunakan Arduino Mega2560.
- Menggunakan IC 74HC595 dan ULN2003 sebagai driver 7-segment.
- Menampilkan waktu sholat lokal secara statis.
- Pengaturan jeda iqomah melalui tombol lokal, tidak ada kendali jarak jauh.
- Tidak ada waktu syuruq

2. Sistem Tahun 2019

- Tetap menggunakan Arduino Mega2560 sebagai inti sistem.
- Fitur tambahan waktu syuruq.
- Tidak terintegrasi dengan internet atau waktu global.
- Sudah ada pengaturan tampilan *running text* menggunakan *smartphone*, namun belum mendukung pengaturan jeda iqomah dan waktu global melalui *smartphone*.

Sistem jadwal sholat digital yang dikembangkan pada tahun 2016 menggunakan Arduino Mega2560 sebagai mikrokontroler utama yang bertanggung jawab mengontrol tampilan waktu. Untuk mengendalikan modul 7-segment common anoda, sistem ini memanfaatkan kombinasi IC 74HC595 sebagai shift register dan ULN2003 sebagai driver, yang berfungsi untuk memperkuat arus ke masing-masing segmen LED. Sistem ini hanya mampu menampilkan waktu sholat harian secara statis berdasarkan data yang dimasukkan secara manual tanpa adanya pembaruan otomatis dari jaringan internet. Selain itu, pengaturan jeda iqomah dilakukan secara manual melalui tombol lokal yang terpasang di perangkat, sehingga memerlukan keterlibatan fisik dari pengurus masjid setiap kali ingin melakukan perubahan. Pada sistem ini, waktu syuruq belum ditampilkan, sehingga informasi tersebut tidak tersedia bagi jamaah yang ingin menunda sholat subuh hingga waktu syuruq berlalu. Sistem ini secara keseluruhan belum mendukung kendali jarak jauh dan belum terintegrasi dengan waktu global, sehingga akurasi waktu tergantung pada penyetelan manual.

Pada pengembangan sistem tahun 2019, perangkat keras utama masih menggunakan Arduino Mega2560 sebagai otak dari keseluruhan rangkaian. Pembaruan penting dari versi sebelumnya adalah penambahan fitur waktu syuruq, sehingga sistem mampu memberikan informasi lebih lengkap terkait jadwal sholat. Meski demikian, sistem ini belum terhubung ke internet dan tidak mendukung sinkronisasi waktu global secara otomatis, sehingga jam sistem masih bergantung pada penyesuaian manual yang rentan terhadap kesalahan waktu. Namun, terdapat kemajuan dalam hal antarmuka pengguna, yaitu dengan adanya fitur pengaturan tampilan *running text* melalui *smartphone*, yang memungkinkan pengurus masjid menambahkan pengumuman atau informasi lain yang ditampilkan di layar. Kendati begitu, fitur pengaturan jeda iqomah dan sinkronisasi waktu global melalui *smartphone* belum tersedia, sehingga fleksibilitas sistem masih terbatas. Sistem ini mencerminkan langkah awal menuju integrasi digital yang lebih modern, namun belum sepenuhnya mengadopsi teknologi Internet of Things (IoT).

3.3 Identifikasi Permasalahan

Permasalahan utama yang ditemukan pada sistem jadwal sholat digital yang telah terpasang sebelumnya, baik versi tahun 2016 maupun 2019, dapat diuraikan sebagai berikut :

1. Tidak adanya sinkronisasi waktu otomatis dengan waktu global (internet).

Kedua sistem sebelumnya masih mengandalkan pengaturan waktu secara manual melalui tombol atau potensiometer. Hal ini sangat rawan terhadap kesalahan dalam penyesuaian waktu, terutama ketika terjadi perubahan waktu resmi seperti penyesuaian waktu musim atau ketidakakuratan jam internal. Akibatnya, waktu sholat yang ditampilkan sering kali tidak sinkron dengan waktu resmi dari otoritas seperti BMKG atau Kemenag, dan dapat menyebabkan keterlambatan atau ketidaksesuaian dalam pelaksanaan ibadah di masjid.

2. Tidak adanya pengaturan jeda iqomah melalui *smartphone*.

Sistem tahun 2016 dan sistem tahun 2019, pengaturan jeda iqomah hanya dapat dilakukan melalui tombol lokal yang terpasang di perangkat. Tidak adanya akses jarak jauh membuat pengurus masjid kesulitan untuk melakukan penyesuaian, terutama jika sedang tidak berada di lokasi perangkat. Hal ini

menyulitkan penyesuaian cepat yang sering kali diperlukan dalam situasi khusus seperti adanya ceramah tambahan, hujan deras, atau pergantian imam.

3. Ketergantungan pada pengaturan manual.

Seluruh konfigurasi dan pengaturan waktu, baik waktu sholat maupun jeda iqomah, dilakukan secara manual, baik melalui komputer lokal (upload ulang program) atau tombol fisik. Pendekatan ini tidak hanya menyulitkan proses pemeliharaan sistem, tetapi juga mengurangi efisiensi dalam manajemen waktu ibadah. Di era digital saat ini, pengurus masjid memerlukan sistem yang mudah diakses dan diatur secara fleksibel dari berbagai perangkat.

4. Keterbatasan fitur tampilan dan informasi:

Pada sistem 2016, tidak tersedia informasi tambahan seperti waktu syuruq atau pengumuman penting. Sedangkan sistem 2019 meskipun sudah memiliki fitur *running text* yang dapat dikendalikan melalui *smartphone*, fitur-fitur lainnya seperti jeda iqomah dan pengaturan waktu masih belum dapat diakses secara jarak jauh. Hal ini menunjukkan bahwa sistem belum sepenuhnya memanfaatkan potensi Internet of Things (IoT) secara menyeluruh.

Permasalahan-permasalahan ini menjadi dasar dalam perancangan sistem optimasi yang lebih modern dan terintegrasi. Dengan memanfaatkan modul ESP32 yang mendukung konektivitas Wi-Fi, sistem baru dirancang untuk dapat melakukan sinkronisasi waktu secara otomatis menggunakan protokol NTP (*Network Time Protocol*) dan memungkinkan pengaturan jadwal dan jeda iqomah melalui *smartphone* melalui antarmuka web. Inovasi ini bertujuan meningkatkan akurasi, kenyamanan, dan fleksibilitas dalam pengelolaan jadwal ibadah di lingkungan masjid.

3.4 Perancangan Sistem Optimasi

Sistem optimasi dirancang dengan mengintegrasikan dua buah mikrokontroler, yaitu Arduino Mega2560 dan ESP32, masing-masing dengan fungsi yang spesifik dan saling melengkapi. ESP32 digunakan sebagai penghubung sistem ke jaringan internet dan antarmuka pengguna, sedangkan Arduino Mega2560 bertugas mengelola tampilan waktu dan logika utama sistem.

a. Fungsi Arduino Mega2560 :

- Mengontrol tampilan jadwal sholat digital menggunakan 7-segment *common anoda*.
- Mengatur tampilan waktu sholat, waktu syuruq, jam berjalan, serta countdown iqomah.
- Menerima data dari ESP32 melalui komunikasi serial UART untuk memperbarui parameter seperti waktu sholat dan jeda iqomah.

b. Fungsi ESP32 :

- Menyinkronkan waktu secara otomatis melalui internet menggunakan protokol NTP.
- Menyediakan antarmuka pengguna berbasis web (*web-based UI*) yang dapat diakses melalui *smartphone* atau laptop.
- Mengirimkan data konfigurasi seperti jeda iqomah dan waktu sholat ke Arduino Mega2560 secara *real-time*.

Komunikasi antar mikrokontroler dilakukan melalui protokol UART. Dalam desain ini, ESP32 berfungsi sebagai otak untuk komunikasi dan kendali jarak jauh, sedangkan Arduino Mega2560 tetap berfungsi sebagai otak logika dan pengendali tampilan. Pendekatan ini menjadikan sistem lebih modular dan mudah dikembangkan lebih lanjut di masa depan.

3.5 Implementasi dan Pengujian Sistem

1 Setelah proses perancangan selesai, tahap implementasi dilakukan di Masjid Rahmatan Lil 'Alamin (RLA) Universitas Putra Indonesia YPTK Padang. Sistem dirakit dan diintegrasikan ke dalam perangkat eksisting, kemudian dilakukan pengujian menyeluruh untuk mengevaluasi kinerja dan kestabilan sistem. Pengujian dilakukan dalam beberapa aspek penting, yaitu:

1. Uji Sinkronisasi Waktu :

- a. Memastikan ESP32 dapat terhubung ke internet dan mengambil waktu global dari server NTP.
- b. Membandingkan hasil waktu tampilan dengan waktu resmi BMKG.

2. Uji Pengaturan Jeda Iqomah :

- a. Mengakses antarmuka web melalui *smartphone* untuk mengubah jeda iqomah.
- b. Memastikan pengaturan tersebut diterima Arduino dan ditampilkan sebagai *countdown*.

3. Uji Komunikasi Serial (UART) :

- a. Mengirim dan menerima data dari ESP32 ke Arduino dan sebaliknya.
- b. Mengamati stabilitas koneksi selama penggunaan normal.

4. Uji Keandalan Operasional :

- a. Menjalankan sistem selama beberapa hari dalam kondisi sebenarnya
- b. Mengamati respons terhadap gangguan koneksi, pemadaman listrik, serta perubahan pengaturan.

Waktu implementasi dan pengujian sistem dirinci dalam tabel berikut:

Tabel 3.1 Jadwal Pelaksanaan Penelitian

No	Tahapan Kegiatan	Waktu Pelaksanaan (Tahun 2025)											
		Februari				Maret				April			
		1	2	3	4	1	2	3	4	1	2	3	4
1	Studi sistem lama	■	■										
2	Perancangan sistem optimasi			■	■								
3	Implementasi sistem					■	■	■					
4	Pengujian fungsional awal							■	■	■			
5	Evaluasi dan pengujian lanjutan									■	■	■	■
6	Pembuatan Laporan			■	■	■	■	■	■	■	■	■	■

Hasil pengujian dibandingkan langsung dengan sistem tahun 2016 dan 2019. Fokus utama evaluasi terletak pada peningkatan fleksibilitas pengaturan, keakuratan waktu, kemudahan penggunaan antarmuka, dan keandalan operasional sistem selama periode pengujian. Dengan sistem baru ini, diharapkan pengelolaan waktu ibadah di masjid menjadi lebih efisien dan responsif terhadap kebutuhan jamaah dan pengurus.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Implementasi Sistem

Hasil dari proses perancangan dan implementasi sistem menunjukkan bahwa optimasi sistem jadwal sholat digital berbasis ESP32 dan Arduino Mega2560 dapat berjalan dengan baik. Sistem ini berhasil mengintegrasikan fitur sinkronisasi waktu global serta pengaturan jeda iqomah melalui antarmuka smartphone. ESP32 bertugas menangani komunikasi jaringan dan antarmuka pengguna, sedangkan Arduino Mega2560 mengelola tampilan waktu dan countdown iqomah pada 7-segment display. Adapun fitur-fitur yang berhasil diimplementasikan antara lain :

1. Sinkronisasi waktu secara otomatis melalui server NTP.
2. Antarmuka web yang dapat diakses melalui smartphone untuk pengaturan jeda iqomah.
3. Komunikasi serial antara ESP32 dan Arduino untuk pertukaran data.
4. Tampilan waktu sholat, jam digital, serta countdown jeda iqomah secara real-time.

Pengoperasian alat dapat dilakukan dengan cukup mudah. Pengurus masjid hanya perlu menyambungkan ESP32 ke jaringan Wi-Fi, kemudian mengakses halaman antarmuka dari *smartphone*. Semua pengaturan dilakukan melalui web interface yang intuitif dan responsif.

4.2 Dokumentasi Pengujian Alat

Pengujian dilakukan terhadap alat yang telah dirakit dan dipasang di lokasi. Berikut dokumentasi pengujian:

4.2.1 Sistem Tahun 2016

- Mikrokontroler : Arduino Mega2560.
- Driver tampilan : 74HC595 + ULN2003.
- Tampilan : 7-Segment *common anoda*.

- Fitur : Penampilan waktu sholat lokal secara statis.
- Kelemahan : Tidak ada waktu syuruq, dan tidak terkoneksi internet.



Gambar 4.1 Tampilan sistem jadwal sholat tahun 2016

4.2.2 Sistem Tahun 2019

- Mikrokontroler : Arduino Mega2560.
- Tambahan fitur : Penambahan waktu syuruq, pengaturan tampilan melalui *smartphone*.
- Kekurangan : Tidak ada sinkronisasi waktu otomatis dan tidak bisa mengatur jeda iqomah via *smartphone*.



Gambar 4.2 Tampilan sistem jadwal sholat tahun 2019

4.2.3 Sistem Hasil Optimasi Tahun 2025

Perangkat ini menggunakan mikrokontroler Arduino Mega2560 yang dipadukan dengan ESP32 sebagai inti sistemnya. Beberapa fitur utama yang ditawarkan antara lain adalah sinkronisasi waktu secara otomatis melalui NTP

(*Network Time Protocol*), serta kemampuan pengaturan jeda iqomah dan parameter lainnya langsung dari smartphone pengguna. Selain itu, sistem ini juga mampu menampilkan *countdown* iqomah secara otomatis berdasarkan input yang dikirimkan dari *smartphone*. Keunggulan utama dari sistem ini adalah fleksibilitas yang tinggi, responsivitas terhadap kondisi aktual di masjid, serta akurasi yang lebih baik dibandingkan sistem konvensional.

Setelah peralatan dibuat, selanjutnya peralatan diuji untuk mengetahui sejauh mana kinerja dari peralatan yang dirancang. Pengujian dilakukan pada beberapa rangkaian dari blok diagram antara lain ; pengujian rangkaian tombol, *Seven Segment*, *Buzzer*. Disamping itu pengujian juga dilakukan pada sinronisasi waktu global dan pengujian pengiriman data jeda iqomah.

4.2.3.1 Pengujian Rangkaian Tombol

Pengujian dilakukan dengan cara mengamati nilai tegangan pada saat tombol dalam kondisi ditekan maupun tidak ditekan. Hasil pengukuran ditampilkan pada Tabel 4.1.

Tabel 4.1. Pengujian Tombol

No.	Kondisi Tombol	Tegangan Output (VDC)
1	Ditekan	0.00 V
2	Tidak Ditekan	4.99 V

Berdasarkan data pengujian, diketahui bahwa ketika tombol ditekan, tegangan output sebesar 0,00 VDC, yang dalam sistem digital diartikan sebagai logika '0' (*Low*). Hal ini terjadi karena jalur output tombol terhubung langsung ke ground. Sebaliknya, saat tombol tidak ditekan, tegangan output tercatat sebesar 4,99 VDC. Kondisi ini disebabkan oleh koneksi output ke sumber tegangan melalui resistor sebesar 4,7 K Ω , yang dalam sistem digital dikategorikan sebagai logika '1' (*High*). Dengan demikian, dapat disimpulkan bahwa rangkaian tombol input 1 hingga 4 berfungsi dengan baik dan sesuai untuk dibaca sebagai input digital oleh modul Arduino.

4.2.3.2 Pengujian *Seven Segment Waktu* (jam, menit, detik)

Pengujian dilakukan dengan mengamati secara langsung kinerja *Seven Segment* dalam menampilkan angka 0 hingga 9 pada format waktu (jam, menit, dan detik). Hasil pengujian menunjukkan bahwa *Seven Segment* mampu menampilkan seluruh angka dari 0 sampai 9 dengan baik.



Gambar 4.3. Pengujian Tampilan *Seven Segment Waktu* (jam, menit, detik)

Sebagian hasil dari proses pengujian ditampilkan pada Gambar 4.3. Berdasarkan hasil tersebut, dapat disimpulkan bahwa sistem *Seven Segment* dalam menampilkan waktu (jam, menit, dan detik) bekerja secara optimal.

4.2.3.3 Pengujian *Seven Segment Tanggal / Hari*

Pengujian dilakukan dengan mengamati secara langsung tampilan tanggal dan hari pada *Seven Segment*, lalu membandingkannya dengan data tanggal yang ditampilkan pada laptop (PC). Hasil dari pengujian ini ditampilkan pada Gambar 4.4.



(a)

(b)

(c)

(a) Tampilan Hari

(b) Tampilan Tanggal

(c) Tampilan Setting waktu Pada Laptop (PC)

Gambar 4.4. Pengujian Tampilan *Seven Segment Tanggal / Hari*

1 Dalam gambar 4.4 tersebut terlihat bahwa tampilan hari dan tanggal muncul secara bergantian dengan jeda waktu sekitar 3 detik. Interval waktu ini dipilih berdasarkan estimasi rata-rata waktu yang dibutuhkan seseorang untuk membaca informasi hari dan tanggal, yaitu sekitar ± 3 detik. Setelah dilakukan perbandingan antara tampilan pada *Seven Segment* dan pengaturan waktu pada laptop, hasil yang diperoleh sama, yaitu hari Kamis bertepatan dengan tanggal 05 Juni 2025. Berdasarkan hasil tersebut, dapat disimpulkan bahwa rangkaian *Seven Segment* untuk penunjuk waktu (jam, menit, dan detik) berfungsi dengan baik.

1 **4.2.3.4 Pengujian *Seven Segment* Jadwal Sholat**

1 Pengujian dilakukan dengan cara mengamati secara langsung tampilan waktu sholat pada *Seven Segment*, lalu membandingkannya dengan data jadwal sholat yang tersimpan di dalam MicroSD. Hasil dari pengujian ini ditunjukkan pada Gambar 4.5.



(a)

Jadwal Waktu Shalat Juni

Tgl	Subuh	Syuruq (Setengah Subuh)	Dhuha	Zuhur	Ashar	Magrib	Iyaa	Iyaa Kiblat
01	05:03	06:11	06:40	12:18	15:43	18:20	19:34	16:31:28
02	05:03	06:12	06:40	12:18	15:43	18:20	19:34	16:34:28
03	05:03	06:12	06:40	12:18	15:43	18:20	19:35	16:37:25
04	05:03	06:12	06:40	12:18	15:43	18:20	19:35	16:40:18
05	05:03	06:12	06:41	12:19	15:44	18:21	19:35	16:43:06
06	05:03	06:12	06:41	12:19	15:44	18:21	19:35	16:45:49
07	05:03	06:12	06:41	12:19	15:44	18:21	19:35	16:48:32

(b)

(c)

- (a) Tampilan Jadwal Sholat Pada Alat
- (b) Tampilan Jadwal Sholat Pada Kalender
- (c) Tampilan Jadwal Sholat Pada MicroSD

Gambar 4.5. Pengujian Tampilan *Seven Segment* Jadwal Sholat

Dari gambar 4.5 tersebut, terlihat bahwa waktu sholat untuk tanggal 05 Juni 2025 yang ditampilkan oleh alat sesuai dengan jadwal pada kalender maupun data di MicroSD. Rincian nilai tersebut dapat dilihat pada Tabel 4.2.

Tabel 4.2. Pengujian Data Jadwal Sholat

No.	Nama	Alat	Kalender	MicroSD	Hasil
1	Imsak	04:53	04:53	04:53	Sama
2	Shubuh	05:03	05:03	05:03	Sama
3	Syuruq	06:12	06:12	06:12	Sama
4	Dzuhur	12:19	12:19	12:19	Sama
5	Ashar	15:44	15:44	15:44	Sama
6	Maghrib	19:21	19:21	19:21	Sama
7	Isya	19:35	19:35	19:35	Sama

Dengan demikian dari hasil yang diperoleh dapat dikatakan bahwa rangkaian *Seven Segment* jadwal sholat dapat bekerja dengan baik.

4.2.3.5 Pengujian Rangkaian Buzzer

Pengujian dilakukan dengan memberikan sinyal tegangan berupa logika '0' dan '1' pada input rangkaian *Buzzer*, kemudian mengamati bunyi yang dihasilkan. Hasil dari pengujian ini disajikan dalam Tabel 4.3.

Tabel 4.3. Pengujian Buzzer

No.	Logika Input	Kondisi Buzzer
1	'0'	Tidak aktif
2	'1'	Aktif / Mengeluarkan Suara / bunyi yang bisa didengar dalam radius ± 30 meter.

Hasil pengujian menunjukkan bahwa saat diberikan logika '0', *Buzzer* berada dalam kondisi nonaktif (tidak menghasilkan suara). Sebaliknya, ketika logika '1' diberikan, *Buzzer* aktif dan menghasilkan bunyi yang dapat terdengar hingga jarak sekitar ± 30 meter. Berdasarkan pengujian ini, dapat disimpulkan bahwa rangkaian

Buzzer berfungsi dengan baik dan memenuhi kriteria untuk digunakan sebagai penanda waktu masuk sholat.

4.2.3.6 Pengujian Sinkronisasi Waktu Global

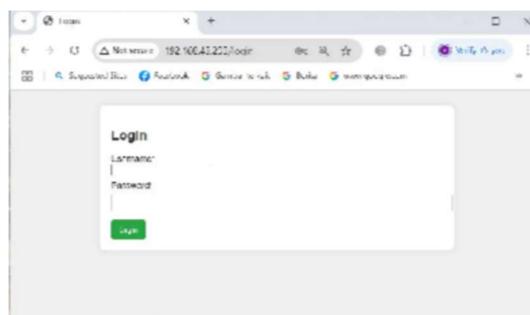
Pengujian ini bisa dilakukan menggunakan browser pada *smartphone* maupun browser pada laptop. Sistem akan membaca data waktu yang berasal dari server *Network Time Protocol* (NTP) yang terdapat pada *smartphone* maupun laptop. Sebelum pengiriman data waktu tersebut, pastikan modul ESP32 sudah terhubung ke jaringan Wi-Fi. Jika sudah terhubung, selanjutnya lakukan pengaturan sinkronisasi waktu, admin melakukan login pada *Web browser* dengan memasukkan alamat IP : `http://192.168.43.233` . Alamat ini diperoleh dari tampilan pada Serial Monitor Arduino IDE, seperti ditunjukkan pada gambar 4.6 berikut ini.



```
Output Serial Monitor X
Message (Enter to send message to 'ESP32 Dev Module' on 'COM3')
-
WiFi connected! IP: 192.168.43.233
HTTP server started
```

Gambar 4.6. Tampilan Alamat IP Akses ESP32

Selanjutnya lakukan login sebagai Admin dan masukkan *passwordnya*, seperti tampilan pada gambar 4.7.



Gambar 4.7. Tampilan Login ke Sistem

3 Setelah berhasil login, Selanjutnya akan tampil *Dashboard* untuk pengaturan Jeda Iqomah dan Sinkronisasi Waktu. Untuk melakukan Sinkronisasi Waktu *Network Time Protocol* (NTP) klik tombol Kirim Waktu ke Arduino, seperti ditunjukkan pada gambar 4.8.



Gambar 4.8. Tampilan *Dashboard* Pengaturan Sinkronisasi Waktu NTP

1 Gambar 4.9 menunjukkan tampilan hasil pengaturan sinkronisasi Waktu *Network Time Protocol* (NTP) pada alat. Tampilan pada *Seven Segment* waktu (jam, menit, dan detik) menunjukkan angka : 11:39:45. Masing-masing alat jadwal sholat menunjukkan nilai yang sama. Ini menunjukkan bahwa pengaturan sinkronisasi waktu NTP dapat bekerja dengan baik.

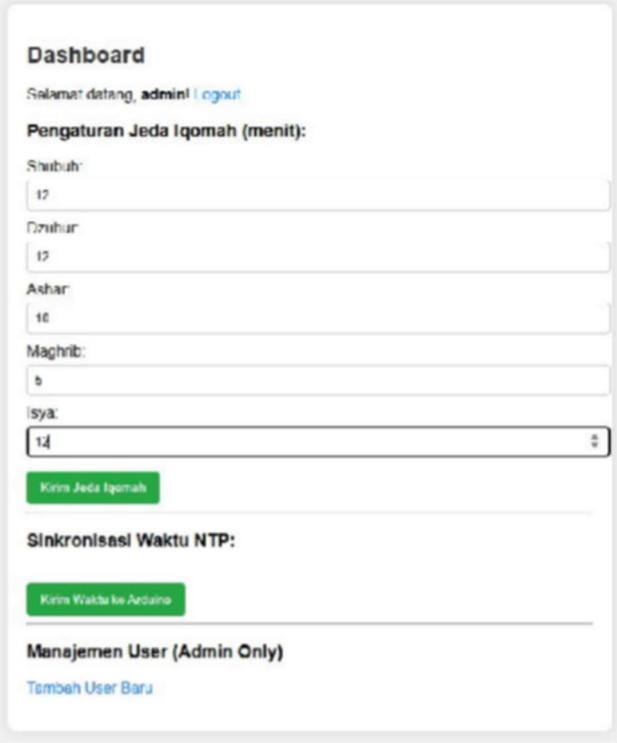


Gambar 4.9. Tampilan Hasil Pengaturan Sinkronisasi Waktu NTP Pada Alat

4.2.3.7 Pengujian Waktu Jeda Iqomah

Pengujian pengaturan waktu jeda Iqomah dilakukan dengan mengirimkan data waktu jeda Iqomah ke Arduino Mega2560 melalui ESP32. Pengaturan jeda

Iqomah dilakukan secara bersamaan untuk kesemua waktu sholat, yaitu waktu Shubuh, Dzuhur, Ashar, Maghrib dan Isya. Perubahan nilai waktu jeda Iqomah sangat mudah dilakukan, cukup dengan menekan tombol panah atas maupun bawah disisi kanan *Dashboard*, seperti ditunjukkan pada gambar 4.10.



The screenshot shows a web dashboard titled "Dashboard". At the top, it says "Selamat datang, admin! Logout". Below this is the section "Pengaturan Jeda Iqomah (menit):". It contains five input fields for different prayer times: Shubuh (12), Dzuhur (12), Ashar (10), Maghrib (5), and Isya (14). Each field has a small arrow on the right side, indicating it is a dropdown menu. Below the input fields is a green button labeled "Kirim Jeda Iqomah". Underneath is the section "Sinkronisasi Waktu NTP:" with a green button labeled "Kirim Waktu ke Arduino". At the bottom, there is a section "Manajemen User (Admin Only)" with a link "Tambah User Baru".

Gambar 4.10. Tampilan *Dashboard* Pengaturan Sinkronisasi Waktu Jeda Iqomah

Setelah tombol Kirim Jeda Iqomah diklik, maka nilai waktu Shubuh, Dzuhur, Ashar, Maghrib dan Isya akan dikirimkan ke Arduino Mega2560. Setelah dilakukan pengujian, maka diperoleh hasil bahwa nilai jeda Iqomah yang dikirimkan ke Arduino Mega2560 sama dengan nilai yang ditampilkan oleh alat Jadwal sholat Digital. Dengan demikian pengaturan waktu jeda Iqomah melalui ESP32 yang dikirim pada smartphone dalam bekerja dengan baik.

4.3 Analisis Perbandingan Sistem

Analisis perbandingan dilakukan terhadap tiga versi sistem jadwal sholat digital yang telah diterapkan di Masjid Rahmatan Lil 'Alamin (RLA) Universitas Putra Indonesia YPTK Padang, yaitu sistem tahun 2016, 2019, dan sistem hasil

optimasi tahun 2025. Setiap sistem memiliki karakteristik dan fitur yang mencerminkan perkembangan teknologi serta kebutuhan operasional masjid yang semakin kompleks.

Sistem tahun 2016 merupakan sistem paling dasar dengan fungsi utama menampilkan waktu sholat lokal secara statis menggunakan tampilan 7-segment yang dikendalikan oleh Arduino Mega2560. Namun, sistem ini tidak memiliki fitur jeda iqomah, tidak mencantumkan waktu syuruq, dan tidak terintegrasi dengan jaringan internet. Pengaturan waktu dilakukan secara manual sehingga berpotensi terjadi ketidaksesuaian dengan waktu sholat yang resmi.

Kemudian, sistem tahun 2019 memperlihatkan adanya pengembangan dengan penambahan fitur waktu syuruq dan kemampuan untuk mengatur tampilan running text melalui smartphone. Meskipun demikian, sistem ini masih belum mendukung pengaturan jeda iqomah secara fleksibel dan belum terhubung ke internet, sehingga sinkronisasi waktu masih dilakukan secara manual seperti pada sistem sebelumnya.

Adapun sistem hasil optimasi tahun 2025 mengintegrasikan teknologi Internet of Things (IoT) dengan memanfaatkan modul ESP32 sebagai penghubung ke jaringan internet. Sistem ini dapat melakukan sinkronisasi waktu otomatis melalui protokol NTP dan menyediakan antarmuka web yang dapat diakses melalui smartphone. Hal ini memungkinkan pengurus masjid mengatur jeda iqomah serta parameter lainnya secara jarak jauh dan lebih fleksibel. Selain itu, fitur *countdown* iqomah ditampilkan secara otomatis, sehingga lebih membantu dalam mengatur waktu pelaksanaan sholat berjamaah.

Untuk memperjelas perbedaan antar sistem, berikut disajikan tabel perbandingan fitur utama.

Tabel 4.4 Perbandingan Sistem Penjadwalan Sholat

Parameter	Sistem 2016	Sistem 2019	Sistem 2025 (Optimasi)
Sinkronisasi waktu global	Tidak	Tidak	Ya
Pengaturan jeda iqomah via HP	Tidak	Tidak	Ya
Waktu syuruq	Tidak	Ya	Ya
Tampilan via 7-segment	Ya	Ya	Ya
Pengaturan via tombol lokal	Tidak	Ya	Opsional
Antarmuka Web	Tidak	Sebagian	Ya

56 Berdasarkan tabel di atas, dapat disimpulkan bahwa sistem hasil optimasi tahun 2025 menawarkan solusi yang lebih lengkap dan adaptif terhadap kebutuhan pengurus masjid. Fitur-fitur modern yang diterapkan berhasil mengatasi keterbatasan sistem sebelumnya, serta memberikan kemudahan operasional dan peningkatan efisiensi dalam manajemen jadwal sholat.

4.4 Evaluasi Kinerja Sistem

42 Evaluasi kinerja dilakukan untuk menilai sejauh mana sistem hasil optimasi mampu menjawab permasalahan pada sistem sebelumnya serta memenuhi kebutuhan operasional di Masjid Rahmatan Lil 'Alamin (RLA) Universitas Putra Indonesia YPTK Padang. Evaluasi ini mencakup beberapa aspek utama, yaitu keakuratan waktu, kehandalan komunikasi, kemudahan pengaturan, serta ketahanan sistem terhadap kondisi operasional harian.

Pertama, dari sisi keakuratan waktu, sistem yang telah dioptimasi mampu melakukan sinkronisasi otomatis dengan waktu global melalui koneksi internet menggunakan protokol NTP. Hasil pengujian menunjukkan bahwa waktu yang ditampilkan pada tampilan 7-segment sangat presisi dan konsisten dengan waktu resmi dari BMKG maupun server NTP lainnya. Hal ini mengatasi kendala pada sistem lama yang masih bergantung pada pengaturan waktu manual, yang rentan mengalami pergeseran waktu akibat kesalahan manusia atau gangguan teknis.

Kedua, keandalan komunikasi serial antara ESP32 dan Arduino Mega2560 dinilai cukup stabil. Pengujian dilakukan dengan mengirimkan data pengaturan

waktu dan jeda iqomah secara berkala dari antarmuka web ke kontroler, dan tidak ditemukan keterlambatan atau kehilangan data. Ini menunjukkan bahwa protokol komunikasi UART yang digunakan telah diimplementasikan dengan baik.

Dari segi kemudahan pengaturan, penggunaan antarmuka berbasis web memberikan fleksibilitas tinggi bagi pengurus masjid. Pengaturan jeda iqomah dan waktu dapat dilakukan dari *smartphone* tanpa harus berada di dekat perangkat. Hal ini menjadi nilai tambah signifikan dibandingkan sistem sebelumnya yang hanya bisa diatur secara lokal.

Selain itu, uji ketahanan sistem juga dilakukan dengan menjalankan perangkat selama lebih dari seminggu dalam kondisi operasional harian masjid. Sistem terbukti mampu bertahan dalam menghadapi gangguan seperti pemadaman listrik maupun pemutusan koneksi internet sementara. Setelah koneksi internet kembali normal, ESP32 secara otomatis melakukan resinkronisasi waktu tanpa perlu campur tangan pengguna.

Secara keseluruhan, sistem hasil optimasi menunjukkan peningkatan performa yang signifikan dari segi fungsionalitas, akurasi, efisiensi, dan kenyamanan operasional. Evaluasi ini memperkuat kesimpulan bahwa teknologi IoT yang diterapkan berhasil mengatasi keterbatasan sistem sebelumnya dan mampu mendukung manajemen waktu ibadah yang lebih baik di lingkungan masjid.

4.5 Pembahasan Hasil

5 Hasil implementasi sistem penjadwalan sholat digital berbasis ESP32 dan Arduino Mega2560 menunjukkan bahwa integrasi antara perangkat keras dan perangkat lunak dapat memberikan solusi yang efisien dan modern terhadap kebutuhan pengelolaan jadwal ibadah di masjid. Dengan adanya kemampuan konektivitas Wi-Fi pada ESP32, sistem dapat melakukan sinkronisasi waktu global secara otomatis, sehingga waktu sholat yang ditampilkan selalu akurat dan sesuai dengan standar waktu resmi.

Selain itu, pengaturan jeda iqomah melalui antarmuka web yang dapat diakses oleh *smartphone* mempermudah pengurus masjid dalam melakukan penyesuaian jadwal secara fleksibel, terutama saat terjadi perubahan jadwal

kegiatan seperti pengajian atau ceramah tambahan. Ini merupakan kemajuan signifikan dibandingkan sistem sebelumnya yang hanya mengandalkan tombol fisik, yang terbatas dari segi aksesibilitas dan efisiensi.

Dari hasil pengujian, sistem juga menunjukkan respons yang cepat dan stabil ketika menerima perintah melalui *web interface*. Hal ini menunjukkan bahwa komunikasi serial antara ESP32 dan Arduino berjalan lancar, serta pengolahan data pada sisi mikrokontroler dapat dilakukan dengan efektif. Tidak ditemukan delay signifikan antara input pengguna melalui *smartphone* dengan perubahan tampilan di perangkat, yang menandakan sistem ini dapat diandalkan untuk penggunaan harian.

Lebih jauh, dari segi operasional, sistem telah diuji dalam kondisi nyata selama lebih dari satu minggu dan tidak menunjukkan tanda-tanda penurunan performa, baik dalam aspek tampilan waktu, *countdown* iqomah, maupun kestabilan jaringan. Bahkan dalam situasi di mana koneksi internet sempat terputus, sistem tetap mampu mempertahankan waktu terakhir yang valid, dan melakukan resinkronisasi otomatis saat koneksi kembali tersedia.

Adanya fitur tambahan seperti penambahan waktu syuruq juga menjadi poin plus, karena memberikan informasi waktu ibadah yang lebih lengkap. Hal ini menunjukkan bahwa sistem hasil pengembangan ini tidak hanya memperbaiki kekurangan sistem lama, tetapi juga memperluas fungsionalitasnya untuk mendukung kenyamanan jamaah dan pengurus masjid.

Dengan hasil-hasil tersebut, sistem ini membuktikan bahwa penerapan teknologi IoT pada manajemen jadwal ibadah memiliki potensi besar untuk diadopsi lebih luas di masjid-masjid lain. Tingkat keterjangkauan perangkat dan kemudahan implementasinya juga memungkinkan sistem ini direplikasi secara luas dengan sedikit penyesuaian sesuai kebutuhan masing-masing masjid.

22

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian dan pengujian yang telah dilakukan terhadap sistem optimasi penjadwalan sholat digital di Masjid Rahmatan Lil 'Alamin (RLA) Universitas Putra Indonesia YPTK Padang, dapat ditarik beberapa kesimpulan utama sebagai berikut:

1. Sistem optimasi dengan integrasi modul ESP32 berhasil meningkatkan keakuratan waktu penjadwalan sholat melalui sinkronisasi otomatis dengan server NTP, sehingga menghilangkan kesalahan waktu yang sebelumnya sering terjadi pada sistem lama.
2. Penggunaan ESP32 yang terhubung ke jaringan Wi-Fi memungkinkan pengaturan jeda iqomah dan parameter jadwal sholat lainnya secara fleksibel melalui antarmuka web berbasis smartphone, memberikan kemudahan dan efisiensi bagi pengurus masjid.
3. Komunikasi serial UART antara Arduino Mega2560 dan ESP32 berjalan dengan stabil dan handal dalam mentransfer data pengaturan jadwal dan waktu, meskipun dalam kondisi gangguan sinyal masih mampu mempertahankan koneksi dalam batas toleransi.
4. Pengujian operasional di lingkungan masjid menunjukkan sistem dapat berfungsi dengan baik dalam kondisi nyata, dengan ketahanan yang memadai terhadap gangguan koneksi internet dan pemadaman listrik sementara.

Secara keseluruhan, sistem yang diusulkan berhasil mengatasi kelemahan sistem lama dan memberikan solusi berbasis IoT yang modern dan praktis untuk penjadwalan sholat digital di masjid.

5.2 Saran

Berdasarkan pengalaman pengembangan dan pengujian sistem, terdapat beberapa saran yang dapat menjadi pertimbangan untuk pengembangan dan penelitian lanjutan :

23

1. Pengembangan antarmuka pengguna berbasis web agar lebih interaktif dan *user-friendly*, misalnya dengan aplikasi mobile native atau pemberitahuan notifikasi saat terjadi gangguan sinkronisasi waktu.
2. Penambahan fitur backup data dan konfigurasi agar pengaturan tidak hilang saat terjadi pemadaman listrik atau reset sistem.
3. Integrasi sensor lingkungan seperti sensor cahaya untuk otomatisasi kecerahan tampilan dan sensor suara untuk pengaturan suara iqomah secara otomatis.
4. Evaluasi lebih lanjut terhadap keamanan jaringan IoT agar data pengaturan dan koneksi tidak mudah disusupi pihak luar.
5. Pengembangan sistem monitoring jarak jauh yang dapat memberikan laporan status sistem secara berkala kepada pengurus masjid melalui aplikasi atau email.

Saran-saran ini diharapkan dapat menjadi referensi untuk menyempurnakan sistem penjadwalan sholat digital di masa depan dan mendorong penerapan teknologi IoT yang lebih luas di lingkungan masjid dan institusi keagamaan lainnya.

DAFTAR PUSTAKA

- [1] E. Naf'an, "Akurasi Sistem Penjadwalan Sholat Digital Menggunakan Arduino Sebagai Pengendali," *J. Sistim Inf. dan Teknol.*, vol. 1, no. 4, pp. 77–84, 2019.
- [2] Emil Naf'an, "PENGEMBANGAN SISTEM PENJADWALAN SHOLAT DIGITAL DENGAN PENGIRIMAN PESAN PADA RUNNING TEXT MELALUI KOMUNIKASI BLUETOOTH," *J. Teknoif Tek. Inform. Inst. Teknol. Padang*, vol. 9, no. 1, pp. 47–54, Apr. 2021.
- [3] O. Barybin, E. Zaitseva, and V. Brazhnyi, "Testing the Security ESP32 Internet of Things Devices," in *2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T)*, 2019, pp. 143–146.
- [4] M. Anusha, P. B. Kumar, V. Akhil, M. Gouthami, M. . Chinnaaiah, and S. Shaik, "Internet of Things (IOT) based energy monitoring with ESP 32 and using Thingspeak," in *2024 10th International Conference on Communication and Signal Processing (ICCSP)*, 2024, pp. 1383–1387.
- [5] T.-C. Hou, L.-H. Liu, Y.-K. Lan, Y.-T. Chen, and Y.-S. Chu, "An Improved Network Time Protocol for Industrial Internet of Things," *Sensors*, vol. 22, no. 13, p. 5021, Jul. 2022.
- [6] T.Mangayarkarasi, Rajakumar.P, Sivabalan.M, and N. Gokul.R, "Arduino Based Smart Metering System with Smart 2560," in *2024 International Conference on Power, Energy, Control and Transmission Systems (ICPECTS)*, 2024, pp. 1–4.
- [7] L. A. S. I. Akbar, M. S. Iqbal, D. F. Budiman, A. S. Rachman, G. W. Wiriasto, and . S., "PEMANFAATAN RUNNING TEXT SEBAGAI ALAT BANTU INFORMASI WAKTU SHOLAT DI MASJID YAYASAN DARUL HIKMAH LOMBOK TENGAH," *J. Bakti Nusa*, vol. 2, no. 1, pp. 9–14, Feb. 2021.

- [8] I. H. Kurniawan, L. Hayat, and D. N. K. Hardani, "Rancang Bangun Teknologi Penampil dan Pengingat Waktu Sholat Digital Di Lingkungan Pimpinan Cabang Muhammadiyah Baturaden," *J. Pengabd. Tek. dan Sains*, vol. 2, no. 01, Feb. 2022.
- [9] A. Drymonitis, "Introduction to Arduino," 2024, pp. 67–134.
- [10] Arduino, "ARDUINO MEGA 2560 REV3," 2019.
- [11] U. N. Malang, "Pengembangan Modul Pembelajaran Antarmuka dan Komunikasi Data Menggunakan Protokol MQTT dan ESP32," vol. 24, no. 3, pp. 392–407, 2024.
- [12] F. A. Aryatama and S. Samsugi, "Sistem Keamanan Kendaraan Bermotor Dengan ESP32 Menggunakan Kontrol Android," *SMATIKA J.*, vol. 14, no. 01, pp. 167–181, Jul. 2024.
- [13] Z. Didi and I. El Azami, "IoT, Comparative Study Between the Use of Arduino Uno, Esp32, and Raspberry pi in Greenhouses," 2022, pp. 718–726.
- [14] M. Sarosa *et al.*, "Air Cleaning System Based On The Internet Of Things (IoT)," in *2023 International Conference on Electrical and Information Technology (IEIT)*, 2023, pp. 367–371.
- [15] P. Bellini, P. Nesi, and G. Pantaleo, "IoT-Enabled Smart Cities: A Review of Concepts, Frameworks and Key Technologies," *Appl. Sci.*, vol. 12, no. 3, p. 1607, Feb. 2022.
- [16] R. Sonwane, A. Deshmukh, and S. Choudhary, "UART (Universal Asynchronous Receiver Transmitter) for Serial Data Communication: Design and Implementation on FPGA Platform," in *2023 2nd International Conference on Futuristic Technologies (INCOFT)*, 2023, pp. 1–4.
- [17] R. B. Chithra, S. A R, N. E. Mujassim, M. Gupta, and P. K. N, "Design and Implementation of UART With Effective Serial Communication," in *2024 Asia Pacific Conference on Innovation in Technology (APCIT)*, 2024, pp. 1–5.

- [18] I. H. Kurniawan, L. Hayat, and A. Fauzan, "Implementasi Teknologi Jadwal Waktu Sholat dan Media Informasi Digital Berbasis Mikrokontroler di Wilayah Pimpinan Ranting Muhammadiyah Klahang, Kecamatan Sokaraja, Kabupaten Banyumas," *J. Pengabd. Tek. dan Sains*, vol. 5, no. 1, p. 25, Jan. 2025.
- [19] Y. H. Kanoi, S. Abdussamad, and S. W. Dali, "Perancangan Jam Digital Waktu Sholat Menggunakan Arduino Uno," *Jambura J. Electr. Electron. Eng.*, vol. 1, no. 2, pp. 32–39, Oct. 2019.
- [20] W. Helma, H. Alam, J. W. Syafrawali, and R. . Bangun, "Rancang Bangun Running Text Led Display Jadwal Waktu Sholat Berbasis Arduino Uno Sebagai Media Informasi," *J. Electr. Technol.*, vol. 5, no. 2, pp. 2502–3624, 2020.
- [21] S. Amri, W. M. Faizal, A. Azizul, P. Almubarak, and N. Azima, "Implementasi Jadwal Shalat Digital Dengan Menggunakan Running Text Di Mushalla Kampus Politeknik Negeri Bengkalis," *Tanjak J. Pengabd. Kpd. Masy.*, vol. 5, no. 1, Jun. 2024.
- [22] S. Rosad, A. Yudhana, and A. Fadlil, "Jadwal Sholat Digital Menggunakan Metode Ephemeris Berdasarkan Titik Koordinat Smartphone," *IT J. Res. Dev.*, vol. 3, no. 2, pp. 30–43, Jan. 2019.
- [23] S. Sarifudin, M. Manshur, and A. Tirtana, "Penggunaan Komunikasi Bluetooth Pada Smartphone Android Untuk Pengiriman Data Pada Jam Digital Berbasis Arduino," *J. ELTIKOM*, vol. 1, no. 2, pp. 102–112, Jan. 2018.
- [24] A. W. Putra, R. Nuryanto, and A. Tafrikhatin, "Fitur Pengingat Kegiatan Masjid Dengan Kontrol Wi-Fi Berbasis ESP-32 Pada Jam Digital," *J. Pendidik. Tambusai*, vol. 5, no. 3, pp. 6177–6187, 2021.
- [25] M. Anton and M. H. Basri, "Perancangan+Jam+Istiwa+Otomatis+Menggunakan+Running+Text+dan+Speaker+Sebagai+Alat+Bantu+Waktu+Sholat+Di+Masjid+Nurul+Hidayah+Al-Taqwa," vol. 5, no. 2, pp. 43–48, 2020.

LAMPIRAN

Lampiran Penelitian ini terdiri dari:

1. Lampiran Bukti pelaksanaan pembuatan jadwal sholat digital tahun 2016.
2. Manual Book penggunaan jadwal sholat digital tahun 2016.
3. Bukti pelaksanaan pembuatan jadwal sholat digital tahun 2019.
4. Manual Book penggunaan jadwal sholat digital tahun 2019.
5. Bukti pelaksanaan pembuatan jadwal sholat digital tahun 2025.
6. Manual Book penggunaan jadwal sholat digital tahun 2025.
7. Listing Program.

