



SEKOLAH TINGGI TEKNOLOGI TERPADU NURUL FIKRI

**PERANCANGAN SISTEM *BOOKING ONLINE* BERBASIS
WEB PADA MAHKOTA *BARBERSHOP* DENGAN INTEGRASI
TEKNOLOGI *INTERNET OF THINGS (IOT)***

TUGAS AKHIR

ASSAUFU ANGGIE ANGGELA ANGGRAENI

0110220056

PROGRAM STUDI TEKNIK INFORMATIKA

DEPOK

JULI 2024



**STT TERPADU
NURUL FIKRI**

SEKOLAH TINGGI TEKNOLOGI TERPADU NURUL FIKRI

**PERANCANGAN SISTEM *BOOKING ONLINE* BERBASIS
WEB PADA MAHKOTA *BARBERSHOP* DENGAN INTEGRASI
TEKNOLOGI *INTERNET OF THINGS (IOT)***

TUGAS AKHIR

Diajukan sebagai salah satu syarat untuk memperoleh gelar sarjana komputer

STT - NF

ASSAUFU ANGGIE ANGGELA ANGGRAENI

0110220056

PROGRAM STUDI TEKNIK INFORMATIKA

DEPOK

JULI 2024

HALAMAN PERNYATAAN ORISINALITAS

Skripsi/Tugas Akhir ini adalah hasil karya penulis, dan semua sumber baik yang dikutip maupun dirujuk telah saya nyatakan dengan benar.

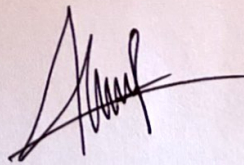
Nama : Assaafi Anggie Anggela Anggraeni

NIM : 0110220056

Depok, 24 Juli 2024

STT - NF

Tanda Tangan



Assaafi Anggie Anggela Anggraeni

HALAMAN PENGESAHAN

Skripsi/Tugas Akhir ini diajukan oleh :

Nama : Assaafi Anggie Anggela Anggredi

NIM : 0110220056

Program Studi : Teknik Informatika

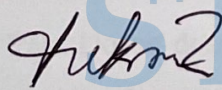
Judul Skripsi : Perancangan Sistem *Booking Online* Berbasis *Web* pada
Mahkota *Barbershop* dengan Integrasi Teknologi *Internet of Things* (IoT)

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Komputer pada Program Studi Teknik Informatika, Sekolah Tinggi Teknologi Terpadu Nurul Fikri

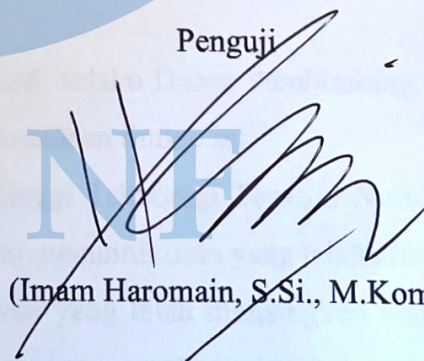
DEWAN PENGUJI

Pembimbing

Penguji



(Dr. Lukman Rosyidi, M.T., M.M.)



(Imam Haromain, S.Si., M.Kom.)

Ditetapkan di : Depok

Tanggal : 24 Juli 2024

KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Allah SWT, karena atas berkat dan rahmat-Nya, penulis dapat menyelesaikan skripsi/Tugas Akhir ini. Penulisan skripsi/Tugas Akhir ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Sarjana komputer Program Studi Teknik Informatika pada Sekolah Tinggi Teknologi Terpadu Nurul Fikri. Penulis menyadari bahwa, tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai pada penyusunan skripsi ini, sangatlah sulit bagi penulis untuk menyelesaikan skripsi/tugas akhir ini. Oleh karena itu, penulis mengucapkan terima kasih kepada:

1. Allah SWT.
2. Orang tua dan semua anggota keluarga yang telah memberikan dorongan baik secara moril maupun materil dalam penyelesaian tugas ini.
3. Bapak Dr. Lukman Rosyidi, M.T., M.M. selaku Ketua Sekolah Tinggi Teknologi Terpadu Nurul Fikri.
4. Ibu Tiffany Nabarian, S.Kom. M.T.I selaku Ketua Program Studi Teknik Informatika Sekolah Tinggi Teknologi Terpadu Nurul Fikri.
5. Bapak Zaki Imaduddin, S.T, M.Kom. selaku Dosen Pembimbing Akademik yang telah membimbing penulis selama perkuliahan di Sekolah Tinggi Teknologi Terpadu Nurul Fikri.
6. Bapak Dr. Lukman Rosyidi, M.T., M.M. selaku Dosen Pembimbing Tugas Akhir penulis dalam menyelesaikan penulisan ilmiah ini.
7. Para Dosen di lingkungan Sekolah Tinggi Teknologi Terpadu Nurul Fikri yang telah membimbing penulis dalam menuntut ilmu yang telah diberikan.
8. Mahkota *Hair Studio* beserta karyawan yang telah meluangkan waktunya untuk memberikan data yang diperlukan bagi penulisan ilmiah ini.

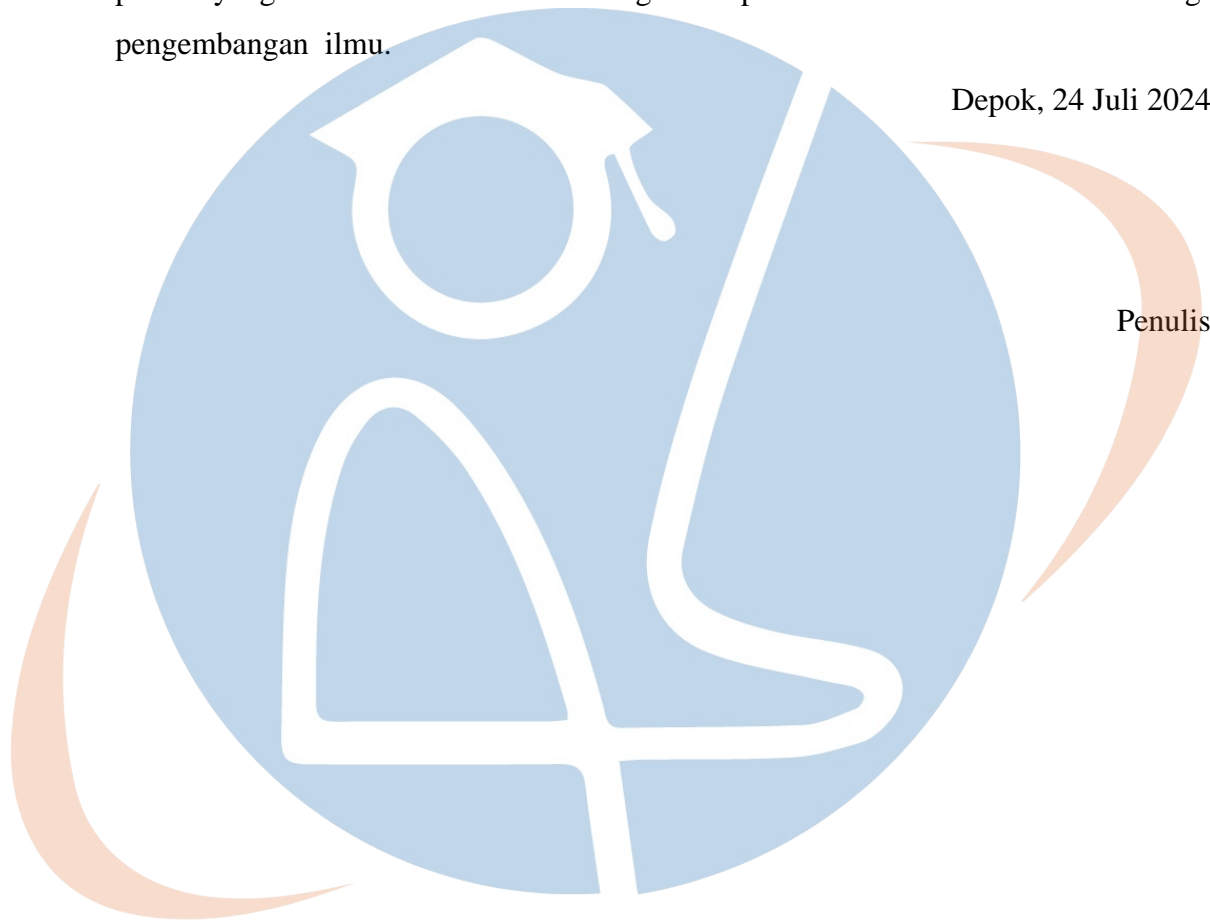
Dalam penulisan ilmiah ini tentu saja masih banyak terdapat kekurangan-kekurangan yang mungkin disebabkan oleh keterbatasan kemampuan dan pengetahuan yang penulis miliki. Walaupun demikian, penulis telah berusaha menyelesaikan penulisan ilmiah ini sebaik mungkin. Oleh karena itu apabila terdapat kekurangan di dalam

penulisan ilmiah ini, dengan rendah hati penulis menerima kritik dan saran dari pembaca.

Akhir kata, penulis berharap Allah SWT berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga skripsi ini membawa manfaat bagi pengembangan ilmu.

Depok, 24 Juli 2024

Penulis



STT - NF

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

Sebagai sivitas akademik Sekolah Tinggi Teknologi Terpadu Nurul Fikri, saya yang bertanda tangan di bawah ini:

Nama : Assaafi Anggie Anggela Anggraeni

NIM : 0110220056

Program Studi : Teknik Informatika

Jenis karya : Tugas Akhir

demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada STT-NF **Hak Bebas Royalti Noneksklusif (*Non-exclusive Royalty - Free Right*)** atas karya ilmiah saya yang berjudul :

Perancangan Sistem *Booking Online* Berbasis *Web* pada Mahkota *Barbershop* dengan Integrasi Teknologi *Internet of Things (IoT)*

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini STT-NF berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok

Pada tanggal : 24 Juli 2024

STT - NF

Yang Menyatakan



(Assaafi Anggie Anggela Anggraeni)

ABSTRAK

Nama : Assaufi Anggie Anggela Anggraeni
NIM : 0110220056
Program Studi : Teknik Informatika
Judul : Perancangan Sistem *Booking Online* Berbasis *Web* pada Mahkota *Barbershop* dengan Integrasi Teknologi *Internet of Things* (IoT)

Para pengelola bisnis perlu memiliki kemampuan untuk berkreasi dalam meningkatkan layanan yang ditawarkan kepada pelanggan serta dalam mengidentifikasi peluang bisnis yang terus berkembang. Mahkota *Barbershop* menghadapi tantangan dalam manajemen antrian dan peningkatan jumlah pelanggan. Penelitian ini bertujuan untuk mengembangkan aplikasi sistem *booking online* berbasis *web* untuk mengelola pesanan pelanggan dengan efisien dan mendeteksi ketersediaan tempat duduk secara *real-time* guna membantu manajemen antrian. Metode yang digunakan meliputi studi literatur, observasi, wawancara, eksperimen, dan analisis data kuantitatif. Implementasi menggunakan metode *agile* untuk penyesuaian cepat terhadap kebutuhan sistem. Aplikasi *web* dikembangkan menggunakan HTML, CSS, JavaScript, PHP, dan MySQL. Alat IoT menggunakan NodeMCU Lua ESP8266 dan sensor ultrasonik HC-SR04, dengan perangkat lunak Arduino IDE. Hasil pengujian menunjukkan sistem *booking online* mencapai tingkat keberhasilan 100% pada fungsi kritis seperti registrasi, *login*, pemilihan layanan, jadwal, konfirmasi *booking*, dan pemberian informasi kepada admin. Sistem IoT juga menunjukkan tingkat keberhasilan 100% dalam pengukuran jarak, pengiriman data, dan respon kendali jarak jauh. Kesimpulannya, kedua sistem tersebut siap diimplementasikan untuk meningkatkan efisiensi operasional dan kualitas layanan di Mahkota *Barbershop*.

Kata kunci : Sistem *Booking Online*, *Web*, *Internet of Things*, Mahkota *Barbershop*, *Black Box Testing*

ABSTRACT

Name : Assaufi Anggie Anggela Anggraeni

NIM : 0110220056

Study Program : *Informatics Engineering*

Title : *The Design of a Web-Based Online Booking System at Mahkota Barbershop with Internet of Things (IoT) Integration*

Business managers need the ability to innovate in enhancing the services offered to customers and identifying evolving business opportunities. Mahkota Barbershop faces challenges in queue management and increasing customer numbers. This study aims to develop a web-based online booking system application to efficiently manage customer orders and detect seat availability in real-time to assist with queue management. The methods used include literature review, observation, interviews, experiments, and quantitative data analysis. Implementation uses the agile method for quick adjustments to system requirements. The web application was developed using HTML, CSS, JavaScript, PHP, and MySQL. The IoT device uses NodeMCU Lua ESP8266 and HC-SR04 ultrasonic sensors, with software developed using the Arduino IDE. Testing results show that the online booking system achieved a 100% success rate on critical functions such as registration, login, service selection, schedule selection, booking confirmation, and providing information to the admin. The IoT system also achieved a 100% success rate in distance measurement, data transmission, and remote control response. In conclusion, both systems are ready to be implemented to improve operational efficiency and service quality at Mahkota Barbershop.

Key words : *Online Booking System, Web, Internet of Things, Mahkota Barbershop, Black Box Testing*

DAFTAR ISI

HALAMAN PERNYATAAN ORISINALITAS.....	3
HALAMAN PENGESAHAN.....	4
KATA PENGANTAR	v
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI.....	vii
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS	vii
ABSTRAK.....	viii
<i>ABSTRACT</i>	ix
DAFTAR ISI.....	x
DAFTAR GAMBAR	xiii
DAFTAR TABEL.....	xv
BAB I PENDAHULUAN.....	1
1.1 Latar belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan dan Manfaat Penelitian.....	2
1.4 Batasan Masalah.....	3
1.5 Sistematika Penulisan.....	3
BAB II KAJIAN LITERATUR.....	5
2.1 Sistem <i>Booking Online</i>	5
2.2 Pengembangan <i>Web</i>	5
2.3 PHP	5
2.4 MySQL.....	6
2.5 Metode <i>Agile</i>	6
2.6 <i>Black Box Testing</i>	6

2.7	<i>Internet of Things</i>	7
2.8	NodeMCU Lua ESP8266.....	7
2.9	Sensor Ultrasonik HC-SR04	8
2.10	<i>Inter Integrated Circuit</i>	9
2.11	<i>Liquid Crystal Display</i>	9
2.12	Arduino IDE.....	10
2.13	Telegram	11
2.14	Penelitian Terkait	12
2.15	Posisi Penelitian	15
BAB III METODOLOGI PENELITIAN		20
3.1	Tahapan Penelitian	20
3.2	Rancangan Penelitian	22
3.2.1	Jenis Penelitian.....	23
3.2.2	Metode Analisis Data.....	23
3.2.3	Metode Pengumpulan Data.....	24
3.2.4	Metode Pengujian.....	25
3.2.5	Metode Implementasi dan Evaluasi	28
3.2.6	Lingkungan Pengembangan	28
BAB IV IMPLEMENTASI DAN EVALUASI		30
4.1	Perancangan Sistem	30
4.1.1	Identifikasi Pengguna.....	30
4.1.2	Kebutuhan Pengguna	31
4.1.3	<i>Use Case Diagram</i>	33
4.1.4	<i>Activity Diagram</i>	34
4.1.5	<i>Sequence Diagram</i>	37

4.1.6	Struktur Menu	39
4.1.7	<i>Class Diagram</i>	40
4.1.8	Arsitektur Sistem IoT	42
4.1.9	Rangkaian Sistem IoT	46
4.1.10	<i>Flowchart</i> Sistem IoT	49
4.2	Implementasi Rancangan Sistem	52
4.2.1	Hasil Sistem <i>Booking Online</i> Pelanggan	52
4.2.2	Hasil Sistem <i>Booking Online</i> Admin	56
4.2.3	Hasil Prototipe Perangkat IoT	60
4.2.4	Kode Program Prototipe Perangkat IoT	61
4.3	Pengujian Kinerja dan Respon	65
4.3.1	Pengujian Hasil Pengukuran	65
4.3.2	Pengujian Pengiriman Data	66
4.4	Pengujian Sistem	67
4.4.1	Pengujian <i>Black Box Testing</i> Sistem <i>Booking Online</i>	68
4.4.2	Pengujian <i>Black Box Testing</i> Perangkat IoT	69
4.5	Evaluasi Hasil Pengujian	70
4.5.1	Evaluasi Hasil Pengujian Sistem <i>Booking Online</i>	71
4.5.2	Evaluasi Hasil Pengujian Perangkat IoT	71
4.5.3	Analisis Evaluasi Hasil Pengujian Sistem <i>Booking Online</i>	71
4.5.4	Analisis Evaluasi Hasil Pengujian Perangkat IoT	72
BAB V KESIMPULAN DAN SARAN		73
5.1	Kesimpulan	73
5.2	Saran	73
DAFTAR PUSTAKA		75

DAFTAR GAMBAR

Gambar 2. 1 NodeMCU ESP8266	7
Gambar 2. 2 Sensor Ultrasonik HC-SR04	8
Gambar 2. 3 <i>Inter Integrated Circuit</i>	9
Gambar 2. 4 LCD 16 x 2.....	10
Gambar 2. 5 <i>Interface</i> Arduino IDE	11
Gambar 2. 6 Aplikasi Telegram.....	11
Gambar 3. 1 Tahapan Penelitian	20
Gambar 4. 1 <i>Use Case Diagram</i> Pelanggan	33
Gambar 4. 2 <i>Use Case Diagram</i> Admin	34
Gambar 4. 3 <i>Activity Diagram</i> Pelanggan	35
Gambar 4. 4 <i>Activity Diagram</i> Admin	36
Gambar 4. 5 <i>Sequence Diagram</i> Pelanggan.....	37
Gambar 4. 6 <i>Sequence Diagram</i> Admin	38
Gambar 4. 7 Struktur Menu Pelanggan.....	39
Gambar 4. 8 Struktur Menu Admin	40
Gambar 4. 9 <i>Class Diagram</i>	41
Gambar 4. 10 Topologi Jaringan Antar Perangkat.....	42
Gambar 4. 11 Blok <i>Diagram</i> Pendeteksi Jumlah Kursi.....	43
Gambar 4. 12 Blok <i>Diagram</i> Pendeteksi Jumlah Pelanggan.....	45
Gambar 4. 13 Rangkaian Sistem IoT	47
Gambar 4. 14 <i>Flowchart</i> Pendeteksi Jumlah Kursi	49
Gambar 4. 15 <i>Flowchart</i> Pendeteksi Jumlah Pelanggan.....	51
Gambar 4. 16 Halaman Beranda Pelanggan	53
Gambar 4. 17 Halaman Layanan.....	53
Gambar 4. 18 Halaman Portofolio	54
Gambar 4. 19 Halaman Daftar Harga	54
Gambar 4. 20 Halaman Tampilan <i>Booking</i>	55

Gambar 4. 21 Halaman Tampilan Kapster.....	55
Gambar 4. 22 Halaman Tampilan Informasi.....	56
Gambar 4. 23 Halaman <i>Login</i> Admin.....	57
Gambar 4. 24 Halaman <i>Dashboard</i>	57
Gambar 4. 25 Halaman Kategori Layanan.....	58
Gambar 4. 26 Halaman Jenis Layanan.....	58
Gambar 4. 27 Halaman Data <i>Booking</i> Pelanggan.....	59
Gambar 4. 28 Halaman Data Kapster	59
Gambar 4. 29 Halaman Jadwal Kapster.....	60
Gambar 4. 30 Prototipe Perangkat IoT	61
Gambar 4. 31 Inisialisasi <i>WiFi</i> dan Koneksi ke <i>Bot</i> Telegram.....	61
Gambar 4. 32 Fungsi <i>Setup</i> Jumlah Kursi.....	62
Gambar 4. 33 Fungsi <i>Loop</i> Jumlah Kursi	62
Gambar 4. 34 Fungsi <i>handleTelegramMessages</i> Jumlah Kursi.....	63
Gambar 4. 35 Fungsi <i>Setup</i> Jumlah Pelanggan.....	63
Gambar 4. 36 Fungsi <i>Loop</i> Jumlah Pelanggan.....	64
Gambar 4. 37 Fungsi <i>handleTelegramMessages</i> Jumlah Pelanggan.....	65

STT - NF

DAFTAR TABEL

Tabel 2. 1 Penelitian Terkait	12
Tabel 2. 2 Posisi Penelitian	15
Tabel 3. 1 Instrumen Pengujian	27
Tabel 4. 1 Identifikasi Pengguna.....	30
Tabel 4. 2 Kebutuhan Pengguna	31
Tabel 4. 3 Pengujian Hasil Pengukuran	66
Tabel 4. 4 Pengujian Pengiriman Data.....	67
Tabel 4. 5 Pengujian Sistem <i>Booking Online</i>	68
Tabel 4. 6 Pengujian Sistem Perangkat IoT.....	69



STT - NF

BAB I

PENDAHULUAN

1.1 Latar belakang

Para pengelola bisnis perlu memiliki kemampuan untuk berkreasi dalam meningkatkan layanan yang ditawarkan kepada pelanggan serta dalam mengidentifikasi peluang bisnis yang terus berkembang. *Barbershop* adalah salah satu contoh bisnis yang harus berinovasi dalam menyediakan layanan potong rambut bagi para pria.

Barbershop merupakan salah satu Usaha Kecil dan Menengah (UKM). *Barbershop* berperan signifikan dalam ekonomi Indonesia dengan memberikan kontribusi melalui penciptaan lapangan kerja serta menyediakan layanan yang bermanfaat bagi masyarakat [1]. Kepuasan pelanggan menjadi faktor utama dalam kelangsungan bisnis *barbershop*, mengingat ketatnya persaingan di dunia bisnis dan pesatnya perkembangan industri yang mendorong penggunaan kemajuan teknologi informasi [2].

Seiring dengan kemajuan teknologi informasi, peluang baru terbuka bagi bisnis untuk meningkatkan produktivitas dan efisiensi. Penggunaan teknologi komputer telah menjadi norma dalam berbagai industri, termasuk sektor ritel dan jasa. Perusahaan-perusahaan berinovasi dalam penerapan teknologi komputer untuk menciptakan data yang menjadi dasar pengambilan keputusan strategis, dikenal sebagai teknologi informasi (TI).

Mahkota *Barbershop* adalah penyedia layanan potong rambut yang telah beroperasi sejak tahun 1996 di Kabupaten Lampung Tengah dengan mengusung konsep pelayanan yang ramah, suasana yang nyaman, dan harga yang terjangkau. Dalam upaya untuk menghadirkan layanan yang lebih efisien dan memperluas kenyamanan bagi pelanggan. Mahkota *Barbershop* menghadapi tantangan dalam manajemen antrian dan peningkatan jumlah pelanggan. Kendati telah menawarkan beragam layanan potong rambut dan perawatan dengan konsep pelayanan yang ramah dan harga yang terjangkau, proses pelayanan masih sering

menyebabkan penumpukan pelanggan di lokasi dan kesulitan dalam mengelola jadwal pemesanan.

Dalam menanggapi tantangan ini, penulis bermaksud untuk mengembangkan aplikasi sistem *booking online* berbasis *web* dan memanfaatkan teknologi *Internet of Things* (IoT) untuk Mahkota *Barbershop*. Aplikasi tersebut memungkinkan pengguna untuk mendapatkan informasi tentang ketersediaan layanan potong rambut, memilih waktu, jenis layanan, dan tukang cukur yang diinginkan melalui infrastruktur *web*. Kemudahan akses aplikasi ini tersedia bagi semua pengguna.

1.2 Rumusan Masalah

Berdasarkan latar belakang permasalahan diatas, maka rumusan masalah pada penelitian ini adalah sebagai berikut :

- a. Bagaimana merancang aplikasi sistem *booking online* berbasis *web* yang dapat membantu Mahkota *Barbershop* mengelola pesanan?
- b. Bagaimana merancang sistem yang dapat mendeteksi ketersediaan tempat duduk dan membantu dalam manajemen antrian di Mahkota *Barbershop*?

1.3 Tujuan dan Manfaat Penelitian

Penyusunan penelitian ini memiliki tujuan sebagai berikut :

- a. Mengembangkan aplikasi sistem *booking online* berbasis *web* yang dapat mengelola pesanan pelanggan dengan efisien di Mahkota *Barbershop*.
- b. Merancang dan mengimplementasikan alat bantu teknologi yang dapat mendeteksi ketersediaan tempat duduk secara *real-time* untuk membantu manajemen antrian di Mahkota *Barbershop*.

Adapun manfaat dari penelitian ini yaitu :

- a. Berpartisipasi dalam upaya pengembangan ilmu pengetahuan dan teknologi, terutama dalam penerapan teknologi *web* dan *Internet of Things* (IoT) dalam bisnis UKM.
- b. Memberikan solusi yang inovatif bagi Mahkota *Barbershop* dalam mengatasi tantangan dalam manajemen antrian dan peningkatan jumlah pelanggan.
- c. Meningkatkan pengalaman pelanggan dengan memberikan layanan yang lebih mudah diakses dan dikustomisasi melalui aplikasi sistem *booking online* berbasis *web*.

1.4 Batasan Masalah

Berdasarkan rumusan masalah yang telah dijelaskan sebelumnya, pentingnya menetapkan batasan penelitian menjadi hal yang krusial, terutama untuk memastikan kesesuaian fokus penelitian dengan rencana yang telah dirancang, seperti yang dijabarkan berikut ini :

- a. Penelitian ini hanya memfokuskan pada rancangan aplikasi sistem *booking online* berbasis *web* untuk Mahkota *Barbershop* di Kabupaten Lampung Tengah.
- b. Penelitian ini menggunakan integrasi IoT dalam bentuk prototipe dengan 2 sensor ditempatkan pada kursi potong rambut dan 2 sensor ditempatkan pada kursi tunggu di Mahkota *Barbershop*.
- c. Komponen elektronika yang digunakan berupa NodeMCU Lua ESP8266, sensor ultrasonik HC-SR04, LCD 16 x 2, dan indikator (*Light Emitting Diode*) LED Hijau.

1.5 Sistematika Penulisan

Penulis akan menyusun tugas akhir ini dengan sistematika penulisan sebagai berikut :

- a. BAB I PENDAHULUAN, membahas pentingnya inovasi dalam layanan pelanggan dan pengelolaan bisnis dengan Mahkota *Barbershop* sebagai studi kasus. Meliputi perumusan masalah, tujuan penelitian, manfaat, dan batasan masalah serta mencakup integrasi IoT dalam bentuk prototipe dan komponen elektronika yang digunakan.
- b. BAB II KAJIAN LITERATUR, membahas teori dan literatur dasar penelitian, termasuk pengembangan sistem *booking online*, penggunaan IoT dalam manajemen antrian, menghitung jumlah pelanggan, dan penggunaan komponen elektronika seperti NodeMCU Lua ESP8266 dan sensor ultrasonik HC-SR04. Selain itu, akan dijelaskan metode *Agile*, *black box testing*, serta konsep dasar I2C dan LCD.
- c. BAB III METODOLOGI PENELITIAN, menjelaskan tahapan penelitian, metode analisis data kuantitatif menggunakan pendekatan *Agile*, serta metode pengumpulan data melalui studi literatur, observasi, wawancara, dan eksperimen. Pengujian sistem dilakukan untuk memastikan kinerja sistem dan alat IoT.
- d. BAB IV ANALISIS DAN PERANCANGAN, bab ini akan membahas mengenai analisis serta perancangan *website* sistem *booking online* dan prototipe IoT menggunakan Metode *Agile*.
- e. BAB V PENGUJIAN DAN EVALUASI, bagian ini akan mencakup analisis hasil evaluasi serta pengujian *Functional Testing* dengan pendekatan *black box testing* yang telah dilaksanakan.
- f. BAB VI KESIMPULAN DAN SARAN, bab ini akan merangkum hasil penelitian tentang keberhasilan prototipe sistem *booking online* berbasis *web* dan integrasi IoT di Mahkota *Barbershop*. Terdapat saran-saran konkret untuk meningkatkan efisiensi dan pengalaman pengguna di Mahkota *Barbershop* serta rekomendasi pengembangan sistem serupa.

BAB II KAJIAN LITERATUR

2.1 Sistem *Booking Online*

Sistem *booking online* adalah sebuah perangkat lunak yang digunakan untuk mengatur reservasi sehingga penerimaan pesanan menjadi lebih efisien dan efektif. Melalui sistem pemesanan *online*, pelanggan dapat dengan mudah membuka situs *web*, memilih tanggal layanan yang diinginkan, dan melengkapi formulir. Upaya menghindari pelanggan menanyakan ketersediaan layanan secara manual pada saat melakukan pembelian, sistem memiliki kemampuan untuk memperbarui data atau informasi secara *real time*. Pemesanan *online* dianggap jauh lebih efisien dibandingkan teknik tradisional. Pelanggan dapat memesan layanan yang mereka inginkan hanya dengan beberapa klik [3].

2.2 Pengembangan *Web*

Pengembangan *web* adalah proses merancang, membangun, dan memelihara situs *web* termasuk aktivitas perencanaan desain, pembuatan konten, pengkodean, pengujian, dan peluncuran. Tujuan pengembangan *web* adalah untuk menciptakan pengalaman pengguna dan fungsionalitas ideal yang memenuhi kebutuhan. Proses pengembangan *web* juga melibatkan pemeliharaan dan peningkatan berkelanjutan untuk memastikan situs atau aplikasi *web* tetap relevan dan berkinerja baik seiring waktu [4].

2.3 PHP

Hypertext Preprocessor (PHP) adalah bahasa pemrograman *open source* yang diintegrasikan ke dalam kode HTML dan ditujukan untuk pengembangan *web*. PHP dapat dikatakan bahwa bahasa mudah dipelajari yang menggambarkan sejumlah bahasa pemrograman, termasuk C, Java, dan Perl. PHP berfungsi sebagai bahasa *scripting server-side* yang memungkinkan *server* menangani

pemrosesan data dengan klien yang membuat permintaan akan menerima hasilnya setelah *server* menerjemahkan skrip program [5] .

2.4 MySQL

My Structured Query Language (MySQL) adalah bahasa pemrograman terstruktur (*Structured Query Language*) yang dikenal dengan singkatan SQL sebagai salah satu jenis *database* yang umum digunakan untuk mengembangkan aplikasi *web* yang dinamis. MySQL termasuk dalam kelompok Sistem Manajemen Basis Data Relasional (*Relational Database Management System/RDBMS*) dan tersedia secara gratis di bawah lisensi *General Public License* (GPL). MySQL kompatibel dengan bahasa pemrograman PHP. Bahasa *query Structured Query Language* (SQL) dasar di MySQL menggunakan karakter *escape* seperti PHP [6].

2.5 Metode Agile

Metode *Agile* suatu metode dalam pengembangan perangkat lunak yang menekankan pembangunan sistem dalam periode waktu singkat dan menekankan fleksibilitas untuk beradaptasi dengan cepat terhadap perubahan. Penerapan metodologi *Agile* akan mempermudah penyelesaian proyek pengembangan situs *web* secara tepat waktu dan efisien, serta memungkinkan adaptasi yang lebih lancar terhadap perubahan kebutuhan pengguna [7].

2.6 Black Box Testing

Black box testing adalah jenis pengujian yang relatif mudah dijalankan. Setiap fungsionalitas sistem diuji dengan memasukkan *input* dan mengamati *output* sebagai bagian dari proses pengujian. Metode ini berbasis pada serangkaian skenario pengujian yang mencakup kasus pengujian, ekspektasi hasil, hasil pengujian, dan status pengujian. Dalam pendekatan ini, penguji hanya perlu memahami alur kerja yang diharapkan dari sistem tanpa perlu memiliki

pengetahuan tentang kode sumbernya. Jenis pengujian ini juga dikenal sebagai *Functional Testing* karena fokus utamanya adalah pada fungsionalitas sistem [8].

2.7 *Internet of Things*

Internet of Things (IoT) adalah konsep yang mengacu pada manfaat dari benda-benda yang terus terhubung dengan komunikasi *internet*. IoT berfungsi sebagai saluran komunikasi antara tiga perangkat yang terhubung secara bersamaan sehingga pengguna dapat mengoperasikannya dari jarak jauh. IoT memanfaatkan koneksi *internet* sebagai penghubung antara interaksi benda-benda dan pengguna yang memungkinkan pengawasan dan pengaturan langsung terhadap kinerja perangkat tersebut. Konsep IoT memiliki keunggulan dalam meningkatkan kecepatan eksekusi tugas yang dilakukan [9].

2.8 NodeMCU Lua ESP8266

NodeMCU adalah *platform* IoT dan kit pengembangan *open source* dengan memanfaatkan sketch Arduino IDE untuk membantu membuat prototipe produk IoT. NodeMCU dengan berat 7 gram dan memiliki dimensi 4,83 cm x 2,54 cm memiliki *firmware open source* dan fitur *WiFi* [10].



Gambar 2. 1 NodeMCU ESP8266

Seperti pada gambar 2.1 NodeMCU memiliki konsumsi daya yang sangat rendah dan harga terjangkau. Modul WiFi menjadi *System on Chip* (SoC) sehingga pemrograman pada ESP8266 dapat dilakukan tanpa memerlukan mikrokontroler. Rangkaian produk ESP8266 memiliki banyak variasi. Variasi yang paling sering ditemui adalah ESP8266 seri ESP-01. ESP8266 mempertahankan *deep sleep mode* yang membuat penggunaan sehari-hari relatif lebih efisien dengan catu daya 3,3V.

2.9 Sensor Ultrasonik HC-SR04

Sensor ultrasonik HC-SR04 adalah sensor yang mengukur jarak menggunakan gelombang ultrasonik. Sensor ini beroperasi seperti radar ultrasonik dengan memancarkan sinyal dan diterima kembali oleh sensor ultrasonik. Jarak objek diwakili oleh perbedaan waktu pengiriman dan penerimaan. Proses karakterisasi sensor melibatkan pengukuran sensitivitas, koefisien korelasi, dan fungsi transfer [11]. Tujuan dari metode karakterisasi ini adalah untuk memastikan karakteristik sensor ultrasonik HC-SR04 agar optimalisasinya maksimal. Sensor Ultrasonik HC-SR04 memiliki 4 pin seperti pada gambar 2.2



Gambar 2. 2 Sensor Ultrasonik HC-SR04

Berdasarkan pin yang terdapat pada sensor ultrasonik, terdapat pin *trig* untuk mengirim sinyal ultrasonik dan pin *echo* untuk menerima sinyal tersebut. Prosedur karakterisasi dapat dinyatakan dengan persamaan digunakan untuk menghitung fungsi *transfer* sensor HC-SR04.

$$Y_i = a + bX_i$$

Keterangan :

X = jarak sensor (cm)

Y = *pulsa echo* (μs)

b = sensitivitas ($\mu\text{s}/\text{cm}$)

a = *slope* (koefisien regresi) (μs)

2.10 *Inter Integrated Circuit*

Inter Integrated Circuit (I2C) adalah protokol komunikasi kontemporer untuk bertukar data antara sirkuit dan sistem terintegrasi. I2C adalah teknologi komunikasi serial dua arah yang mengirim dan menerima data melalui dua jalur khusus [12]. Perangkat I2C dan pengontrolnya berkomunikasi melalui saluran *Serial Data* (SDA) dan *Serial Clock* (SCL) yang membentuk sistem I2C.



Gambar 2. 3 *Inter Integrated Circuit*

Seperti pada gambar 2.3 adanya fleksibilitas dalam mengirim data, I2C menjadi pilihan yang sangat berguna dalam menghubungkan berbagai perangkat dalam sistem elektronik moderen. Salah satu contohnya adalah penggunaannya pada modul *Liquid Crystal Display* (LCD). I2C memungkinkan LCD untuk berinteraksi dengan pengontrol atau mikrokontroler tanpa membutuhkan banyak saluran kabel, sehingga mempermudah integrasi dalam desain sistem yang kompleks.

2.11 *Liquid Crystal Display*

Liquid Crystal Display (LCD) merupakan tampilan yang terbuat dari materi cair kristal dengan beroperasi pada sistem *dot* matriks. Penggunaan LCD sangat umum pada berbagai perangkat elektronik seperti multimeter digital, jam,

kalkulator, dan elektronik lainnya [13]. LCD 16 x 2 yang diterapkan dalam penelitian ini memiliki dimensi layar dua baris dengan 16 kolom. serta berwarna biru tipe 0x27 seperti pada gambar 2.4

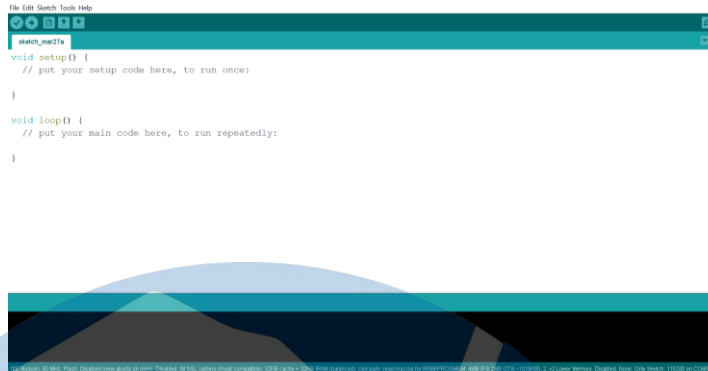


Gambar 2. 4 LCD 16 x 2

Penggunaan LCD memiliki manfaat yang signifikan dalam konteks Mahkota *Barbershop*, informasi tentang jumlah kursi kosong dapat ditampilkan kepada pelanggan dari luar ruangan. Dengan menggunakan LCD untuk menampilkan informasi seperti jumlah kursi kosong atau ucapan selamat datang, pelanggan dapat dengan mudah melihat ketersediaan tempat dan membuat keputusan sebelum memasuki ruangan. Hal ini tidak hanya meningkatkan pengalaman pelanggan dengan memberikan informasi yang berguna secara langsung, tetapi juga membantu dalam mengelola antrian dan kapasitas ruangan secara efisien.

2.12 Arduino IDE

Arduino *Integrated Development Environment* (IDE) adalah perangkat lunak yang memfasilitasi pembuatan *sketch* pemrograman dan juga berfungsi sebagai media pemrograman untuk *board* yang diinginkan. Arduino IDE memfasilitasi pengeditan, pengembangan, pengunggahan ke *board* yang ditentukan, serta penulisan kode spesifik. Perangkat lunak ini dikembangkan melalui bahasa pemrograman JAVA dan berbagai *library* C/C++ untuk menyederhanakan operasi *input* dan *output* [14].

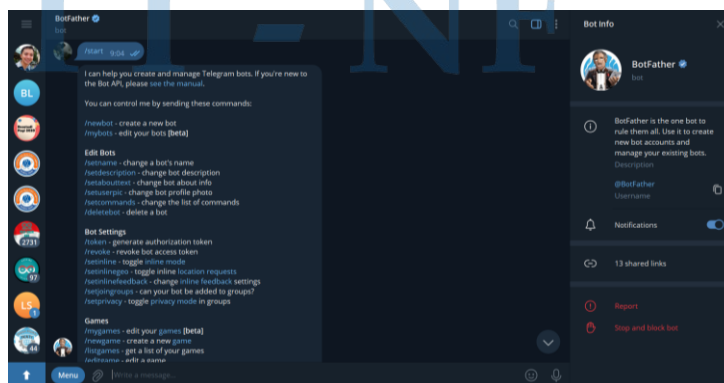


Gambar 2. 5 Interface Arduino IDE

Seperti pada gambar 2.5 program Arduino IDE disebut sebagai *sketch*. Setelah selesai ditulis, program atau *sketch* dapat langsung melalui proses kompilasi menjadi kode biner yang dapat dimengerti oleh mikrokontroler Arduino. Selama kompilasi, program diperiksa untuk kesalahan sintaks dan struktur. Jika tidak ada kesalahan, kode tersebut dikonversi menjadi bahasa mesin dan diunggah ke Arduino *board* untuk menjalankan perintah yang telah ditentukan dalam kode yang telah ditulis.

2.13 Telegram

Telegram adalah sebuah *platform* komunikasi instan yang menggunakan pengujian keamanan berdasarkan protokol *mtproto* karena enkripsi *end-to-end* yang diterapkan. Seperti halnya aplikasi lain yang sejenis, Telegram memungkinkan pengguna untuk membagikan lokasi antara satu sama lain serta bertukar pesan, gambar dan video [15].



Gambar 2. 6 Aplikasi Telegram

Seperti pada gambar 2.6 terdapat kelebihan-kelebihan yang ditemukan meliputi fitur *cloud* di *server Telegram Messenger* untuk penyimpanan data dan kehadiran robot (*bot*) dengan kecerdasan buatan yang dapat diintegrasikan dengan layanan *online* lainnya. Penulis bertujuan mengembangkan sistem untuk mendeteksi ketersediaan kursi dengan memanfaatkan kemampuan *bot* ini. *Bot Telegram* berfungsi seperti akun asli, dapat dikonfigurasi melalui pendaftaran di *@bofather* di Telegram.

2.14 Penelitian Terkait

Penelitian terkait merujuk pada penelitian sebelumnya dalam bidang atau topik yang serupa. Dalam penelitian ini, terdapat empat penelitian terkait yang memiliki kesamaan topik, yaitu sistem *booking online*, IoT, dan jenis sensor yang digunakan. Tabel berikut mencantumkan penelitian terkait dengan topik yang sedang diteliti.

Tabel 2. 1 Penelitian Terkait

No	Nama dan Tahun	Judul	Subjek	Hasil
1	Tri Ika Jaya Kusumawat dan Hafizh Taufiqul hakim, 2023	Pengembangan Sistem <i>Booking Online</i> Pada Cokro <i>Barbershop</i> Berbasis <i>Website</i>	Cokro <i>Barbershop</i>	<i>Website</i> yang menampilkan halaman <i>login</i> , jadwal ketersediaan dan <i>form booking</i> , untuk halaman admin menampilkan data <i>booking</i>

No	Nama dan Tahun	Judul	Subjek	Hasil
2	Ari Wibowo, Sri Arttini Dwi Prasetyowati dan Suryani Alifah, 2023	Sistem Pendeteksi Ketersediaan dan Lokasi Tempat Duduk pada Ruangan Menggunakan <i>Movable Passive Infra Red (Pir) Array</i>	Kursi Gereja	Alat yang dihasilkan dari deteksi sensor berupa lampu tidak menyala jika kursi kosong dan lampu menyala apabila kursi terisi.
3	Lutfi Triseptian Junaidi Putra, Widya khafa Nova, Chaerul Ilmi Al Ahyari dan Rio Wirawan, 2022	Pembuatan <i>Website Booking Online Barbershop Di Daerah Tebet</i>	Big Boyz <i>Barbershop,</i> Liberro <i>Barbershop dan</i> Chas <i>Barbershop</i>	<i>Website</i> yang menampilkan halaman <i>login,</i> jadwal ketersediaan dan <i>form booking,</i> untuk halaman admin menampilkan data <i>booking</i>

STT - NF

No	Nama dan Tahun	Judul	Subjek	Hasil
4	Marvelina Gracia Hernoko, Suryo Adi Wibowo, Nurlaily Vendyansyah, 2021	Penerapan IoT (<i>Internet Of Things</i>) Smart <i>Parking System</i> dan Pendeteksi Kebakaran Dengan Fitur Monitoring	Area parkir	Alat yang dihasilkan dari deteksi sensor akan menampilkan lampu hijau jika tempat parkir kosong lalu akan menyalakan lampu merah jika tempat parkir sudah ditempati. Sementara itu, ketika sensor mendeteksi kebakaran, <i>buzzer</i> dan <i>waterpump</i> akan diaktifkan.

Tabel ini mencakup beberapa penelitian terkait pengembangan sistem *booking online* dan penggunaan perangkat elektronika sebagai berikut :

- a. Penelitian pertama dalam pembuatan *website booking online*. Sebagai alat pengembangan dengan hasil berupa *website* yang menampilkan halaman *login*, jadwal ketersediaan, dan formulir pemesanan. Halaman admin menampilkan data pemesanan.
- b. Penelitian kedua dalam pembuatan alat pendeteksi ketersediaan kursi. Subjek penelitian adalah kursi gereja dan hasilnya adalah alat berupa sensor akan mendeteksi ketersediaan kursi pada ruangan gereja.

- c. Penelitian ketiga dalam pembuatan *website booking online*. Subjek penelitian meliputi *Big Boyz Barbershop*, *Liberro Barbershop*, dan *Chas Barbershop*. Hasilnya adalah *website* yang menyediakan halaman *login*, jadwal ketersediaan, dan formulir pemesanan, serta halaman admin yang menampilkan data pemesanan.
- d. Penelitian keempat dalam pembuatan alat monitoring ketersediaan slot parkir dan pendeteksi Kebakaran. Subjek penelitian adalah area parkir dengan hasil berupa lampu akan menyala hijau jika sensor ultrasonic mendeteksi tempat parkir masih kosong serta indikator *buzzer* dan *waterbump* akan menyala saat terindikasi kebakaran yang dapat dimonitoring melalui *website*.

2.15 Posisi Penelitian

Berikut adalah tabel yang menunjukkan posisi penelitian terdahulu hingga saat ini :

Tabel 2. 2 Posisi Penelitian

Peneliti	Judul	Subjek	Topik	Tools	Teknologi
Tri Ika Jaya Kusumawat dan Hafiizh Taufiqul hakim	Pengembangan Sistem <i>Booking Online</i> Pada Cokro <i>Barbershop</i> Berbasis <i>Website</i>	Cokro <i>Barbershop</i>	Sistem <i>Booking Online</i>	PHP dan MySQL	<i>Website</i>

Peneliti	Judul	Subjek	Topik	Tools	Teknologi
Ari Wibowo, Sri Arttini Dwi Prasetyowati dan Suryani Alifah	Sistem Pendeteksi Ketersediaan dan Lokasi Tempat Duduk pada Ruangan Menggunakan <i>Movable Passive Infra Red (Pir) Array</i>	Kursi Gereja	Alat Penunjuk Ketersediaan Tempat Duduk	Sensor PIR dan ATMEGA 16	<i>Embedded system</i>
Lutfi Triseptian Junaidi Putra, Widya khafa Nova, Chaerul Ilmi Al Ahyari dan Rio Wirawan	Pembuatan <i>Website Booking Online Barbershop</i> Di Daerah Tebet	Big Boyz <i>Barbershop</i> , <i>Liberro Barbershop</i> dan Chas <i>Barbershop</i>	Sistem <i>Booking Online</i>	PHP dan MySQL	<i>Website</i>

Peneliti	Judul	Subjek	Topik	Tools	Teknologi
Marvelina Gracia Hernoko, Suryo Adi Wibowo, Nurlaily Vendyansyah	Penerapan IoT (<i>Internet Of Things</i>) <i>Smart Parking System</i> dan Pendeteksi Kebakaran Dengan Fitur Monitoring	Area parkir	Alat Monitoring Ketersediaan Slot Parkir dan Pendeteksi Kebakaran	PHP, MySQL, Arduino Mega2560, Modul ESP8266, Sensor Ultrasonik HC-SR04, <i>Flame</i> Sensor dan Sensor MQ-135	<i>Website dan Internet of Things</i>
Assaafi Angie Angela Anggraeni	Perancangan Sistem <i>Booking Online</i> Berbasis Web Pada Mahkota <i>Barbershop</i> Dengan Integrasi Teknologi <i>Internet Of Things</i> (IoT)	Mahkota Hair Studio	Sistem <i>Booking Online</i>	PHP, MySQL, NodeMCU ESP8266, Sensor Ultrasonik HC-SR04 dan Telegram	<i>Website dan Internet of Things</i>

Tabel ini mencakup beberapa posisi penelitian pengembangan sistem *booking online* dan penggunaan perangkat elektronika sebagai berikut :

- a. Penelitian pertama fokus pada pengembangan sistem *booking online* untuk Cokro *Barbershop* dengan menggunakan metode *Waterfall* serta PHP dan MySQL. Hasilnya berupa *website* yang memberikan kemudahan bagi pengguna dengan menampilkan halaman *login*, jadwal ketersediaan, dan formulir *booking*. *Website* ini juga memudahkan admin dalam mengelola data *booking*.
- b. Penelitian kedua mengeksplorasi pengembangan alat pendeteksi ketersediaan tempat duduk di ruangan gereja dengan memanfaatkan sensor PIR dan ATMEGA16, serta menerapkan metode *Embedded system*, alat ini dapat memberikan informasi *real-time* tentang ketersediaan tempat duduk. Hasilnya adalah lampu yang tidak menyala saat kursi kosong dan menyala ketika kursi terisi.
- c. Penelitian ketiga memfokuskan pada pembuatan *website booking online* untuk beberapa *barbershop* di daerah Tebet dengan metode *Waterfall* serta PHP dan MySQL, *website* ini mempermudah pelanggan dalam memesan layanan secara *online*. *Website* ini juga menyediakan halaman khusus untuk admin dapat melihat dan mengelola data *booking*.
- d. Penelitian keempat mengimplementasikan teknologi IoT untuk *Smart Parking System* dan deteksi kebakaran melalui Arduino Mega2560, Sensor Ultrasonik HC-SR04, *Flame Sensor* dan Sensor MQ-135. Hasilnya sistem dapat memantau ketersediaan slot parkir dan mendeteksi kebakaran. Melalui *website* pengguna akan mengetahui ketersediaan tempat parkir serta pengguna juga akan mendapatkan informasi tentang ketersediaan tempat parkir melalui indikator lampu serta pengguna akan mendapatkan informasi terkait kejadian kebakaran melalui *buzzer* dan *waterpump*.
- e. Penelitian penulis merancang sebuah sistem *booking online* berbasis *web* untuk Mahkota *Barbershop* dengan integrasi teknologi IoT

melalui metode *agile* dan perangkat seperti NodeMCU ESP8266 dan sensor Ultrasonik HC-SR04. *Website* ini memberikan kenyamanan kepada pelanggan dalam melakukan pemesanan dengan fitur *booking online* dan *dashboard* admin serta integrasi melalui pesan Telegram untuk monitoring ketersediaan kursi tunggu, kursi cukur dan jumlah pelanggan yang sudah dicukur.

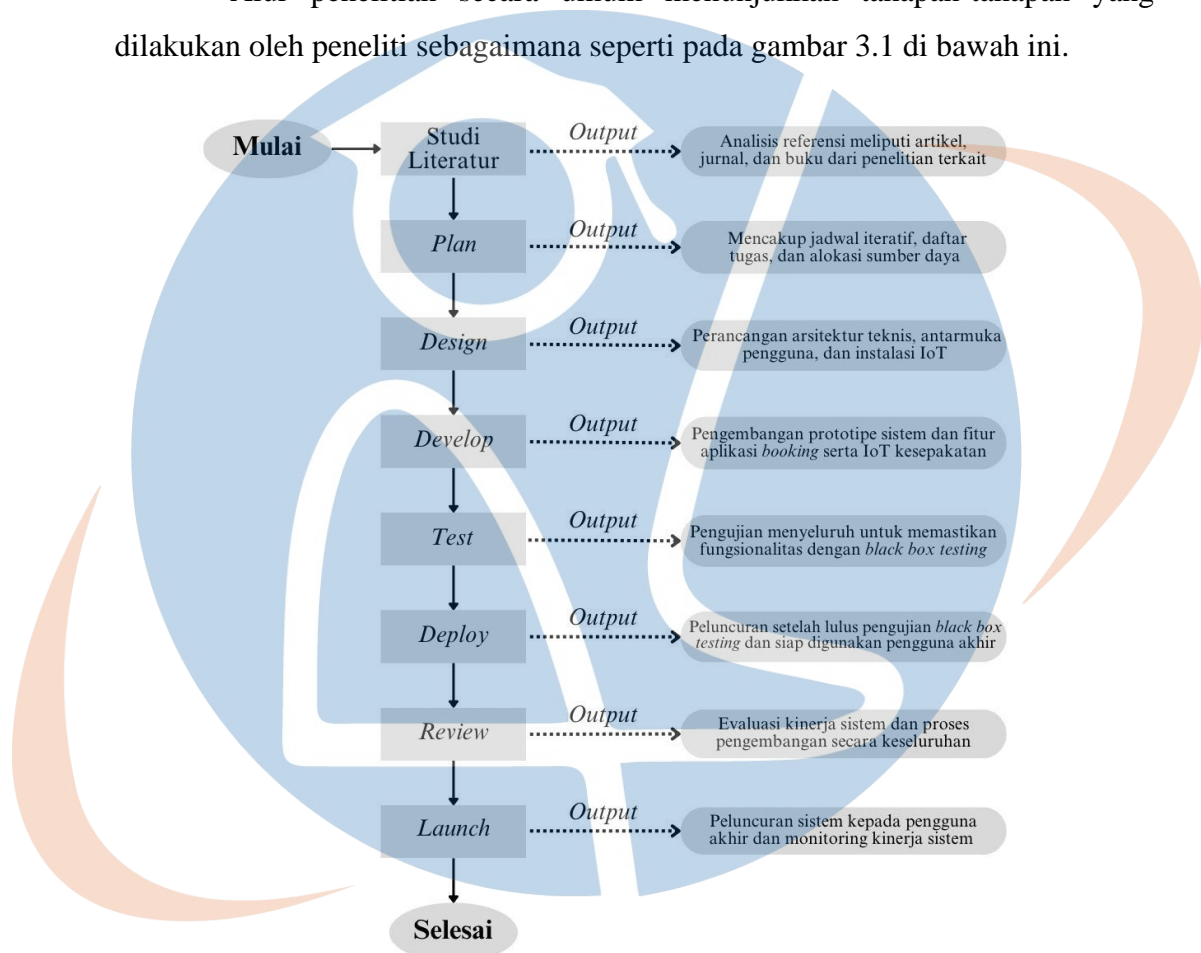


STT - NF

BAB III METODOLOGI PENELITIAN

3.1 Tahapan Penelitian

Alur penelitian secara umum menunjukkan tahapan-tahapan yang dilakukan oleh peneliti sebagaimana seperti pada gambar 3.1 di bawah ini.



Gambar 3.1 Tahapan Penelitian

Pada gambar 3.1 terdapat tahapan penelitian untuk perancangan sistem *booking online* dan penerapan teknologi IoT di *Mahkota Barbershop*. Berikut penjelasan dari tahapan penelitian yang dilakukan penulis :

a. Studi Literatur

Penulis melakukan studi literatur yang mendalam untuk memahami konsep dan teknologi terkini dalam pengembangan sistem *booking online* berbasis *web* dan integrasi IoT mencakup tinjauan literatur

tentang sistem *booking online*, teknologi *web*, dan *aplikasi IoT* dalam konteks barbershop dan literatur yang saling berkaitan.

b. *Plan*

Dalam tahap perencanaan, penulis merumuskan rencana kerja yang terperinci, termasuk tujuan penelitian, jadwal iteratif, dan alokasi sumber daya yang sesuai dengan prinsip-prinsip *agile*. Perencanaan ini memastikan bahwa pengembangan sistem berjalan sesuai dengan target yang ditetapkan dan dapat menyesuaikan diri dengan perubahan kebutuhan yang mungkin terjadi.

c. *Design*

Penulis merancang kerangka sistem *booking online* termasuk antarmuka pengguna, arsitektur teknis, dan fitur-fitur yang diperlukan. Perancangan ini mempertimbangkan kebutuhan pengguna dan prinsip desain UX (*User Experience*) untuk memastikan sistem yang ramah pengguna. Penulis juga membuat rangkaian prototipe alat elektronika yang akan berintegrasi melalui *internet*.

d. *Develop*

Tahap implementasi melibatkan pengembangan prototipe sistem berdasarkan desain yang telah dibuat sebelumnya. Melalui pendekatan *agile*, Pengembangan dilakukan secara berulang, dengan penekanan pada penyampaian fitur-fitur yang dapat dimanfaatkan oleh pengguna. Setiap iterasi diimplementasikan dan diuji sebelum lanjut ke tahap berikutnya.

e. *Test*

Setiap iterasi pengembangan sistem diuji secara menyeluruh oleh tim pengembang untuk memastikan fungsionalitas yang diharapkan. Pengujian *Functional Testing* dengan pendekatan *black box testing* dilakukan oleh pengguna akhir atau pemangku kepentingan utama untuk mengevaluasi sistem secara langsung. Hasil pengujian *black*

box testing memberikan umpan balik penting tentang kelayakan dan kepuasan pengguna sebelum peluncuran resmi.

f. *Deploy*

Setelah sistem lulus pengujian *black box testing* dan dianggap siap, tahap *deployment* dilakukan untuk meluncurkan sistem secara resmi ke lingkungan produksi. Proses *deployment* dilakukan dengan hati-hati untuk memastikan bahwa sistem berjalan lancar dan siap digunakan oleh pengguna akhir.

g. *Review*

Setelah peluncuran sistem, tahap *review* dilakukan untuk mengevaluasi kinerja sistem dan proses pengembangan secara keseluruhan. Tim pengembang dan pemangku kepentingan melakukan evaluasi terhadap keberhasilan mencapai tujuan penelitian, kualitas sistem, serta efisiensi penggunaan sumber daya. *Feedback* dari *review* ini akan menjadi dasar untuk penyempurnaan sistem dan perbaikan proses pengembangan di masa depan.

h. *Launch*

Setelah melewati tahap *review* dan perbaikan jika diperlukan, sistem siap untuk diluncurkan secara resmi kepada pengguna akhir. Peluncuran ini dilakukan dengan mengumumkan kepada pengguna tentang ketersediaan sistem baru dan memberikan dukungan teknis yang diperlukan untuk penggunaan sistem. Proses peluncuran ini harus disertai dengan monitoring yang cermat untuk memastikan kinerja sistem yang optimal setelah diimplementasikan.

3.2 Rancangan Penelitian

Dalam rancangan penelitian ini, penulis akan mengulas tentang jenis penelitian yang digunakan, metode analisis data, metode pengumpulan data, metode pengujian, dan lingkungan pengembangan yang relevan dalam konteks pengembangan aplikasi sistem *booking online* berbasis *web* dengan integrasi teknologi *Internet of Things (IoT)* untuk Mahkota *Barbershop*.

3.2.1 Jenis Penelitian

Penelitian ini mengadopsi pendekatan penelitian pengembangan untuk menciptakan sebuah sistem yang mengintegrasikan aplikasi sistem *booking online* berbasis *web* dengan teknologi IoT melalui aplikasi Telegram di Mahkota *Barbershop*. Pendekatan ini dipilih dengan tujuan utama untuk memperbaharui atau meningkatkan sistem yang sudah ada dalam operasional bisnis Mahkota *Barbershop*. Melalui pengembangan ini, diharapkan sistem yang dihasilkan dapat memberikan gambaran komprehensif tentang perancangan dan dampaknya dalam mengelola aktivitas pemesanan layanan dan manajemen tukang cukur di Mahkota *Barbershop*.

Dalam implementasinya, sistem yang dikembangkan akan menghasilkan dua luaran utama. Pertama, berupa *website booking online* yang memungkinkan pengguna untuk melakukan pemesanan layanan secara *online* melalui *platform* yang telah disediakan. Kedua, melalui alat IoT yang terintegrasi dengan Telegram, sistem akan memberikan informasi *real-time* mengenai jumlah kursi kosong dan jumlah pelanggan yang telah dicukur pada hari tersebut. Dengan demikian, pemantauan dan manajemen operasional Mahkota *Barbershop* dapat dilakukan secara lebih efisien, meningkatkan pengalaman pelanggan dan efektivitas layanan secara keseluruhan.

3.2.2 Metode Analisis Data

Metode analisis data yang diterapkan dalam penelitian ini adalah menggunakan metode kuantitatif. Pendekatan ini dipilih untuk mengukur secara numerik fungsionalitas alat dan pengalaman pengguna dalam pengembangan aplikasi sistem *booking online* berbasis *web* di Mahkota *Barbershop*, serta untuk menilai dampak integrasi teknologi IoT terhadap ketersediaan tempat duduk dan jumlah pelanggan secara *real-time*.

3.2.3 Metode Pengumpulan Data

Metode yang diterapkan untuk mengumpulkan data dalam penelitian ini adalah seperti yang diuraikan berikut ini:

a. Studi Literatur

Penulis melakukan studi literatur dengan mengumpulkan beberapa artikel, jurnal, dan buku relevan dengan topik penelitian. Hasil dari studi literatur ini akan memberikan penulis informasi mengenai perancangan *web*, perancangan IoT, kebutuhan teknologi, fitur yang diperlukan, dan metode yang dapat mendukung proses penelitian dan penulisan.

b. Observasi

Penulis melakukan pengamatan langsung sebagai bagian dari observasi yang dilakukannya terhadap Mahkota *Barbershop*. Hasil observasi menunjukkan bahwa saat ini Mahkota *Barbershop* masih mengandalkan sistem pemesanan manual tanpa adanya *platform booking online* serta ruangnya yang padat karena jumlah pelanggan yang banyak. Dengan demikian, dirancang dan diimplementasikan sistem *booking online* berbasis *web* dengan teknologi IoT untuk menghitung ketersediaan tempat duduk dan total pelanggan, hal ini menjadi relevan untuk meningkatkan efisiensi dan kualitas layanan di Mahkota *Barbershop*.

c. Wawancara

Penulis melakukan wawancara dengan pemilik Mahkota *Barbershop* dan pelanggan untuk mengumpulkan rincian yang lebih akurat mengenai prosedur tersebut pemesanan saat ini serta harapan dan kebutuhan terkait dengan implementasi sistem *booking online*. Hasil wawancara menunjukkan bahwa pemilik berharap sistem *booking online* yang diimplementasikan akan membantu meningkatkan efisiensi operasional dan memperluas jangkauan promosi Mahkota *Barbershop* melalui platform *web*.

Pelanggan juga menyampaikan harapan bahwa dengan bantuan teknologi IoT dapat memberikan pengalaman dalam mengunjungi Mahkota *Barbershop* menjadi lebih nyaman dan efisien.

d. Eksperimen

Penulis melakukan eksperimen untuk menguji kinerja sensor ultrasonik yang diintegrasikan dalam sistem IoT dengan tujuan untuk memastikan keakuratan deteksi objek, responsivitas yang cepat terhadap perubahan status, stabilitas operasional yang konsisten, serta integrasi yang handal dengan platform Telegram untuk pengiriman notifikasi dan melakukan kontrol.

3.2.4 Metode Pengujian

Dalam penelitian ini digunakan metode *Functional Testing* dengan pendekatan *black box testing* untuk menguji fungsionalitas sistem *booking online* dan integrasi IoT pada Telegram untuk Mahkota *Barbershop*. Pendekatan pada pengujian ini dilakukan tanpa memperhatikan struktur internal sistem, namun fokus pada masukan dan keluaran yang dihasilkan oleh sistem. Hal ini memungkinkan evaluasi yang komprehensif terhadap setiap komponen sistem tanpa memerlukan pengetahuan terperinci tentang implementasi internalnya.

Upaya untuk memastikan kualitas dan kinerja sistem *booking online* serta integrasi IoT pada Mahkota *Barbershop* penulis menerapkan pendekatan yang cermat terhadap pengujian data dengan mencakup cara, prosedur, dan instrumen untuk menguji data dengan seksama. Pengujian dilakukan dengan mengikutsertakan partisipasi dari pengguna akhir, termasuk pelanggan dan admin, untuk menggunakan sistem *booking online* dan alat IoT yang telah dirancang.

Selama proses penggunaan sistem, setiap interaksi antara pengguna dan sistem diamati secara cermat, dan semua respons yang dihasilkan dicatat secara sistematis. Data yang terkumpul kemudian

dianalisis secara menyeluruh untuk mengevaluasi apakah fungsionalitas sistem dan alat IoT berjalan sesuai yang diharapkan.

Analisis ini dimaksudkan untuk mengidentifikasi kekuatan dan kelemahan dari sistem serta memberikan wawasan yang diperlukan untuk melakukan perbaikan dan meningkatkan kinerja keseluruhan. Dalam pengujian, langkah-langkah prosedur sebagai berikut :

1. Identifikasi fitur utama sistem yang akan diuji, seperti proses pemesanan, manajemen *booking*, dan kontrol alat IoT.
2. Tentukan aspek penting dari setiap fitur yang akan dievaluasi, seperti kemudahan penggunaan, responsivitas, dan kehandalan.
3. Jadwalkan sesi pengujian dengan pengguna akhir.
4. Pastikan instruksi yang jelas diberikan kepada pengguna tentang tugas yang harus dilakukan selama sesi pengujian.
5. Fasilitasi sesi pengujian dengan pengguna akhir sesuai dengan jadwal yang telah ditetapkan.
6. Amati setiap interaksi pengguna dengan sistem secara cermat.
7. Catat setiap interaksi yang dilakukan pengguna dengan sistem dan respons yang diberikan oleh sistem.
8. Evaluasi hasil pengujian untuk menentukan apakah sistem dan alat IoT berperilaku sesuai dengan harapan.
9. Tinjau catatan interaksi pengguna dan respons sistem untuk mengidentifikasi kekuatan dan kelemahan dari setiap fitur yang diuji.
10. Sesi pengujian diatur dengan pengguna akhir dan instruksi yang jelas diberikan tentang tugas yang harus dilakukan.
11. Setiap interaksi pengguna dengan sistem dan respons yang diberikan oleh sistem diamati dan dicatat.
12. Hasil pengujian dievaluasi untuk menentukan apakah sistem dan alat IoT berfungsi sesuai dengan harapan.

Berikut adalah instrumen-instrumen pengujian yang dirancang untuk mencakup berbagai aspek penting dalam pengujian, mulai dari identifikasi fitur hingga evaluasi respons sistem. Instrumen ini akan diuraikan lebih lanjut dalam Tabel 3.1.

Tabel 3. 1 Instrumen Pengujian

No.	Aspek yang Diuji	Interaksi Pengguna	Respon Sistem	Hasil Pengujian (Berhasil / Tidak)
1	Proses Pemesanan	Pelanggan membuat <i>booking</i>	Sistem menerima dan mengonfirmasi <i>booking</i> dengan sukses	
2	Manajemen <i>Booking Online</i>	Admin melihat dan mengelola <i>booking</i>	Sistem menampilkan <i>booking</i> dengan benar ada <i>dashboard</i> admin	
3	Integrasi Alat IoT	Pengguna mengirim perintah melalui Telegram	Sistem merespon dengan benar terhadap perintah dan memberikan hasil yang sesuai	

No.	Aspek yang Diuji	Interaksi Pengguna	Respon Sistem	Hasil Pengujian (Berhasil / Tidak)
4	Sensor Ultrasonik	Sensor mendeteksi kehadiran pengguna di kursi tunggu atau kursi cukur	Sistem memberikan notifikasi melalui Telegram tentang status kursi	

Rumus dari pengujian *functional testing* dengan pendekatan *black box testing* berfokus pada tingkat keberhasilan atau kegagalan dalam menjalankan kasus uji sebagai berikut :

$$\text{Tingkat Keberhasilan} = \frac{\text{Jumlah Kasus Uji Berhasil}}{\text{Total Kasus Uji}} \times 100\%$$

3.2.5 Metode Implementasi dan Evaluasi

Dalam penelitian ini, penulis memilih untuk menerapkan pendekatan metode *agile*. Pendekatan ini dipilih karena sesuai untuk pengembangan sistem sederhana dan dapat menyesuaikan diri dengan perubahan kebutuhan sistem yang cepat.

Pendekatan metode *agile* mempunyai sejumlah keuntungan, seperti sangat cocok untuk digunakan dengan sistem yang sederhana, memungkinkan dilakukannya perubahan cepat terhadap kebutuhan sistem, dan menjamin bahwa konsekuensi dari kegagalan tidak akan terlalu berat, baik dalam hal materi maupun non-materi [16].

3.2.6 Lingkungan Pengembangan

Dalam pengembangan sistem *booking online* berbasis *web* pada Mahkota *Barbershop* dengan integrasi teknologi IoT, lingkungan

pengembangan yang digunakan mencakup perangkat keras dan lunak serta lokasi penelitian yang relevan. Berikut adalah detail dari lingkungan pengembangan yang digunakan untuk meneliti dan mengimplementasikan sistem ini :

1. Laptop Asus A456U

- Processor: Intel Core i5-6200U (2.3GHz up to 2.8GHz)
- RAM: 8GB DDR4
- Penyimpanan: 1TB HDD
- Grafis: NVIDIA GeForce 930MX 2GB
- Sistem Operasi: Windows 10

2. *Tools*

- PHP: Digunakan untuk pengembangan *backend* sistem *booking online*.
- MySQL: Digunakan sebagai basis data untuk menyimpan informasi pelanggan, *booking*, dan data lainnya.

3. Perangkat IoT

- NodeMCU ESP8266: Digunakan sebagai mikrokontroler untuk mengontrol sensor ultrasonik dan mengirimkan data ke *server*.
- Sensor Ultrasonik: Digunakan untuk mendeteksi ketersediaan kursi di Mahkota *Barbershop*.
- LCD 16 x 2: Dijadikan sebagai layar untuk menampilkan informasi dengan lebar tampilan 2 baris dan 16 kolom.
- Telegram: Digunakan sebagai platform komunikasi untuk mengontrol dan menerima notifikasi dari perangkat IoT.

4. Lokasi Penelitian

- Penelitian dilakukan di Mahkota *Hair Studio* atau Mahkota *Barbershop*, tempat dimana sistem *booking online* dan perangkat IoT diimplementasikan.

BAB IV IMPLEMENTASI DAN EVALUASI

4.1 Perancangan Sistem

Dalam perancangan sistem ini, penulis akan menjelaskan perancangan sistem *booking online* dan sistem perangkat IoT. Penjelasan mencakup identifikasi pengguna, kebutuhan pengguna, *use case diagram*, *activity diagram*, *sequence diagram*, struktur program, dan *class diagram* untuk sistem *booking online*, serta arsitektur sistem, rangkaian sistem, dan *flowchart* sistem untuk perangkat IoT. Tujuan dari perancangan sistem ini adalah untuk menyajikan gambaran menyeluruh tentang bagaimana sistem *booking online* berbasis *web* pada Mahkota *Barbershop* dan integrasi teknologi IoT akan diimplementasikan.

4.1.1 Identifikasi Pengguna

Pada subbab ini, akan dijelaskan mengenai berbagai jenis pengguna yang terlibat dalam sistem *booking online* dan sistem perangkat IoT pada Mahkota *Barbershop*. Pemahaman siapa yang akan berinteraksi dengan sistem, serta peran dan kebutuhan, bergantung pada identifikasi pengguna yang dapat dilihat pada tabel 4.1.

Tabel 4. 1 Identifikasi Pengguna

No.	Pengguna	Deskripsi
1	Pelanggan Mahkota <i>Barbershop</i>	Pelanggan Mahkota <i>Barbershop</i> adalah pengguna yang akan memanfaatkan sistem ini untuk memesan jadwal potong rambut secara <i>online</i> di Mahkota <i>Barbershop</i> , yang berlokasi di Kabupaten Lampung Tengah. Pelanggan dapat melihat informasi layanan, portofolio, harga, dan melakukan <i>booking</i> melalui

No.	Pengguna	Deskripsi
		<i>website</i> . Selain itu, pelanggan juga dapat melihat ketersediaan kursi tunggu melalui layar LCD yang terletak di pintu masuk.
2	Admin Mahkota <i>Barbershop</i>	Admin Mahkota <i>Barbershop</i> adalah pengguna yang bertanggung jawab untuk mengelola data jadwal potong rambut dalam sistem <i>website</i> admin dan mengontrol perangkat IoT melalui telegram. Admin dapat mengedit, menambah, dan menghapus kategori layanan, jenis layanan, data tukang cukur, serta jadwal tukang cukur.

4.1.2 Kebutuhan Pengguna

Bagian ini menguraikan kebutuhan dan ekspektasi dari setiap jenis pengguna yang telah diidentifikasi. Penjelasan mengenai kebutuhan pengguna ini bertujuan untuk menjamin bahwa sistem tersebut mampu memenuhi kebutuhan dan memberikan pengalaman yang baik bagi pengguna dapat dilihat pada tabel 4.2.

Tabel 4. 2 Kebutuhan Pengguna

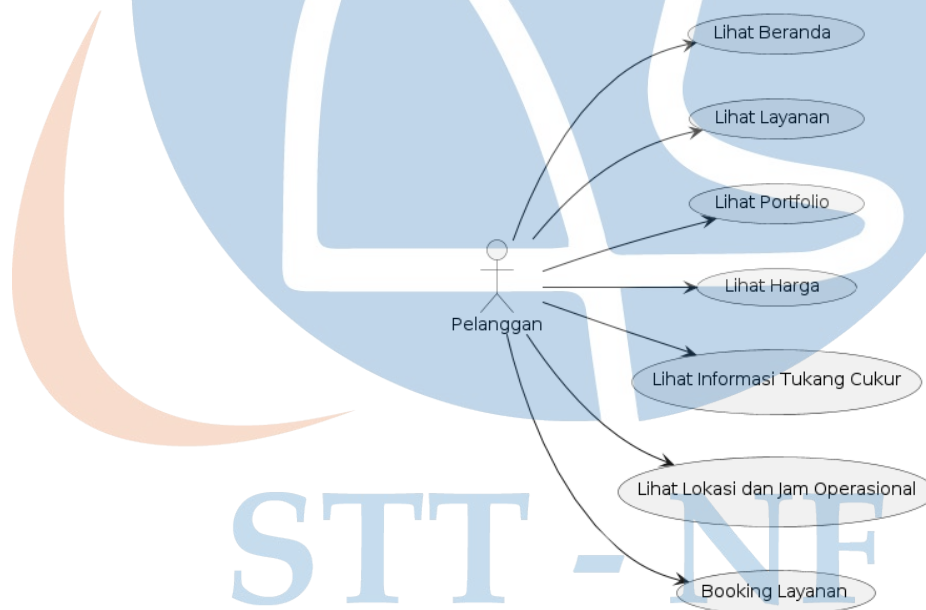
No.	Pengguna	Kebutuhan
1	Pelanggan Mahkota <i>Barbershop</i>	<ul style="list-style-type: none"> - Pelanggan ingin mendapatkan informasi terkait jadwal operasional Mahkota <i>Barbershop</i>. - Pelanggan ingin melakukan pemesanan jadwal potong rambut di Mahkota <i>Barbershop</i>.

No.	Pengguna	Kebutuhan
		<ul style="list-style-type: none"> - Pelanggan ingin mendapatkan informasi terkait jadwal dari masing-masing kapster. - Pelanggan ingin melihat kondisi ketersediaan kursi tunggu. - Pelanggan ingin melihat portofolio model rambut yang dapat menjadi referensi. - Pelanggan ingin mendapatkan informasi terkait harga layanan di Mahkota <i>Barbershop</i>.
2	Admin Mahkota <i>Barbershop</i>	<ul style="list-style-type: none"> - Admin ingin menyampaikan informasi terkait jam operasional Mahkota <i>Barbershop</i>. - Admin ingin menyampaikan informasi terkait jadwal kapster. - Admin ingin menyampaikan informasi terkait layanan apa saja yang tersedia di Mahkota <i>Barbershop</i>. - Admin ingin menyampaikan rekomendasi terkait model rambut yang sedang populer. - Admin ingin mengetahui jumlah pelanggan per hari. - Admin ingin mengelola data jadwal potong rambut dalam sistem. - Admin ingin mengontrol perangkat IoT untuk mendeteksi ketersediaan

No.	Pengguna	Kebutuhan
		kursi tunggu dan jumlah pelanggan yang berhasil dicukur. - Admin ingin mengelola kategori layanan, jenis layanan, data tukang cukur, dan jadwal tukang cukur.

4.1.3 Use Case Diagram

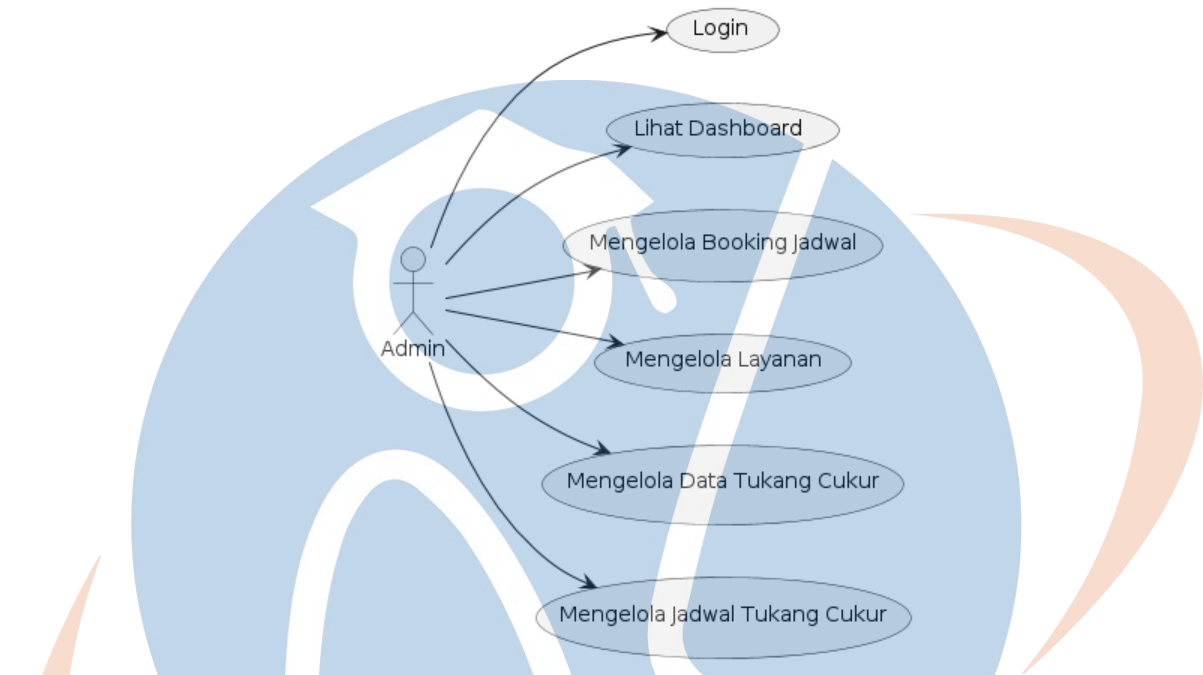
Pada subbab ini, akan ditampilkan *use case diagram* yang menggambarkan bagaimana pengguna berinteraksi dengan sistem *booking online*. *Use case diagram* membantu dalam memvisualisasikan fungsionalitas sistem dari sudut pandang pengguna.



Gambar 4. 1 Use Case Diagram Pelanggan

Pada gambar 4.1 terlihat alur interaksi antara pelanggan dengan berbagai fitur pada sistem *booking online* Mahkota *Barbershop*. Pelanggan dapat melihat berbagai halaman informasi yang tersedia, termasuk Beranda, Layanan, Portfolio, Harga, Informasi Tukang Cukur, dan Lokasi serta Jam Operasional. Selain itu, pelanggan juga dapat melakukan *booking* layanan yang diinginkan. Diagram ini

menggambarkan bagaimana pelanggan berinteraksi dengan sistem untuk mendapatkan informasi dan memesan layanan yang ditawarkan oleh Mahkota *Barbershop*.

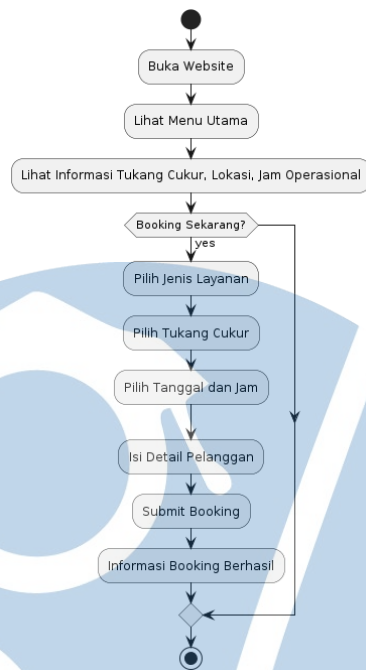


Gambar 4. 2 Use Case Diagram Admin

Pada gambar 4.2 menggambarkan interaksi Admin dengan sistem manajemen Mahkota *Barbershop*. Admin dapat melakukan *login* ke dalam sistem untuk mengakses *Dashboard*. Di dalam *Dashboard*, Admin memiliki akses untuk mengelola berbagai fitur seperti *booking* jadwal, manajemen layanan yang ditawarkan, pengelolaan data tukang cukur, serta pengaturan jadwal kerja dari tukang cukur yang tersedia. *Diagram* ini mencerminkan bagaimana Admin menggunakan sistem untuk mengatur operasional sehari-hari dari Mahkota *Barbershop*, mulai dari manajemen pelanggan hingga pengelolaan layanan dan personel yang terlibat.

4.1.4 Activity Diagram

Bagian ini menyajikan *activity diagram* yang memperlihatkan alur aktivitas pengguna saat berinteraksi dengan sistem. *Diagram* ini membantu dalam memahami proses bisnis yang terjadi di dalam sistem.

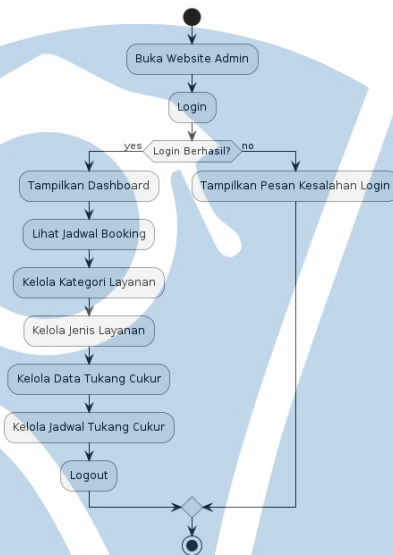


Gambar 4. 3 Activity Diagram Pelanggan

Pada gambar 4.3 menggambarkan langkah-langkah yang dilakukan oleh pelanggan saat berinteraksi dengan *website* Mahkota *Barbershop*. Proses dimulai dengan membuka *website*, kemudian pelanggan dapat melihat beranda untuk mendapatkan informasi umum tentang *barbershop* tersebut. Selanjutnya, pelanggan dapat menjelajahi layanan yang ditawarkan dan melihat portofolio hasil kerja sebelumnya. Pelanggan juga dapat melihat daftar harga layanan yang tersedia serta informasi terkait tukang cukur, lokasi *barbershop*, dan jam operasional.

Pada tahap berikutnya, jika pelanggan memutuskan untuk melakukan *booking* layanan, mereka akan memilih jenis layanan yang diinginkan, memilih tukang cukur yang tersedia, serta menentukan tanggal dan jam untuk melakukan *booking*. Setelah memasukkan detail pelanggan yang diperlukan, pelanggan kemudian mengirimkan *booking* melalui *website*. Jika *booking* berhasil, pelanggan akan menerima konfirmasi bahwa informasi *booking* telah berhasil.

Diagram ini mencerminkan proses interaksi yang sistematis antara pelanggan dengan *website* Mahkota *Barbershop*, mulai dari eksplorasi informasi hingga proses *booking* layanan secara *online*.



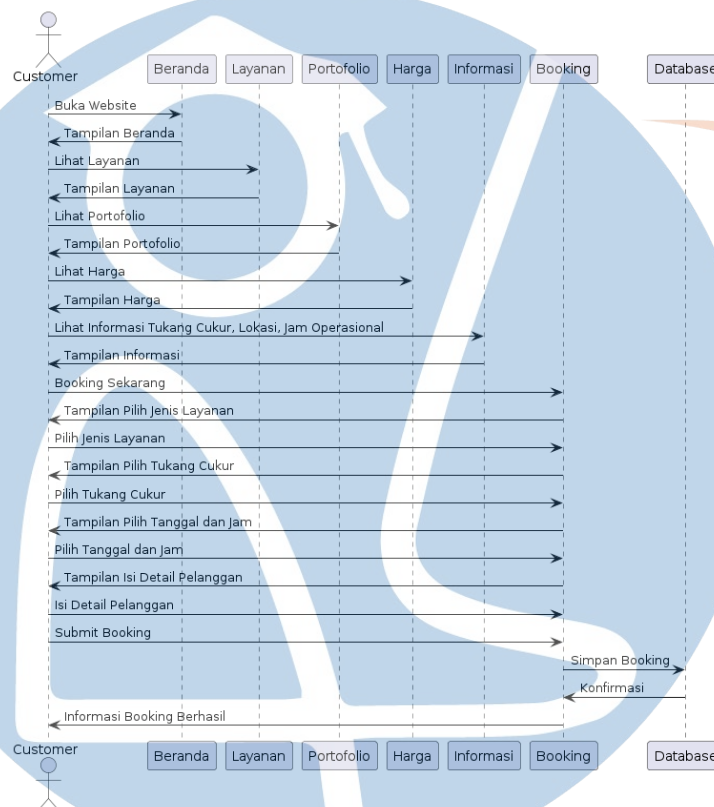
Gambar 4. 4 Activity Diagram Admin

Pada gambar 4.4 mengilustrasikan interaksi seorang admin saat menggunakan sistem manajemen Mahkota *Barbershop* melalui *website* admin. Proses dimulai dengan admin membuka halaman *login* dan memasukkan kredensialnya. Sistem kemudian memvalidasi informasi tersebut. Jika *login* berhasil, admin akan diarahkan ke *dashboard* utama.

Admin memiliki beberapa opsi, termasuk melihat jadwal *booking* yang terjadwal, mengelola kategori dan jenis layanan yang ditawarkan, serta mengatur informasi tukang cukur dan jadwal mereka. Setelah menyelesaikan tugasnya, admin dapat keluar dari sistem dengan aman menggunakan opsi *logout*. *Diagram* ini memberikan gambaran tentang bagaimana admin secara efektif mengelola aspek-aspek kunci dari sistem *booking online* untuk Mahkota *Barbershop*.

4.1.5 Sequence Diagram

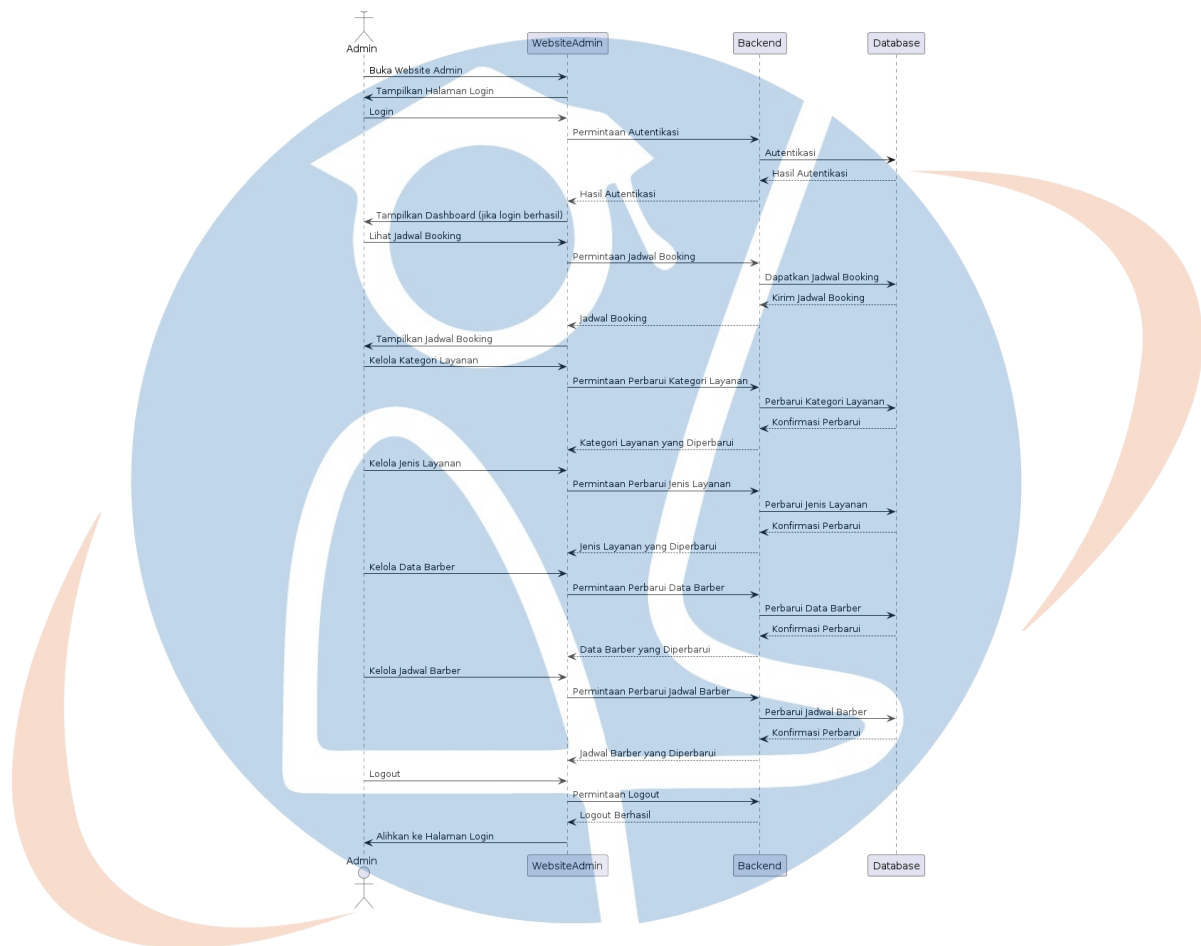
Subbab ini memaparkan *sequence diagram* yang menunjukkan urutan interaksi antara objek dalam sistem selama eksekusi suatu *use case*. *Diagram* ini penting untuk memahami dinamika sistem dan urutan pesan yang dikirim antar objek.



Gambar 4. 5 Sequence Diagram Pelanggan

Pada gambar 4.5 menggambarkan interaksi antara pelanggan dan sistem *booking online Mahkota Barbershop*. Pertama, pelanggan membuka *website* dan menjelajahi fitur-fitur seperti melihat beranda, layanan yang tersedia, portofolio, harga layanan, dan informasi tentang tukang cukur serta lokasi *barbershop*. Jika pelanggan memilih untuk melakukan *booking*, pelanggan akan memilih jenis layanan yang diinginkan, tukang cukur favorit, serta tanggal dan jam untuk *booking*. Setelah memasukkan detail pelanggan, mereka mengirimkan permintaan *booking*. Sistem kemudian menyimpan informasi *booking* ke dalam *database* dan memberikan konfirmasi bahwa *booking* telah berhasil dilakukan kepada

pelanggan. *Diagram* ini memberikan gambaran visual tentang alur interaksi pelanggan dari mulai masuk ke *website* hingga proses *booking* selesai dengan sukses di Mahkota *Barbershop*.



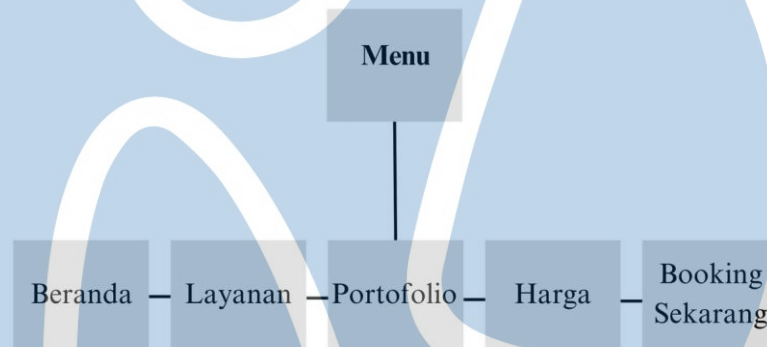
Gambar 4. 6 Sequence Diagram Admin

Pada gambar 4.6 menggambarkan interaksi antara admin dan sistem administrasi Mahkota *Barbershop*. Admin memulai dengan membuka *website* admin dan mengakses halaman *login* untuk masuk ke sistem. Setelah *login*, sistem melakukan autentikasi dengan mengirimkan permintaan ke *database* untuk memverifikasi keaslian admin. Jika autentikasi berhasil, sistem menampilkan dasbor admin yang memuat berbagai fitur seperti melihat jadwal *booking*, mengelola kategori layanan, jenis layanan, data *barber*, dan jadwal *barber*. Setiap kali admin melakukan perubahan pada data atau jadwal, permintaan diteruskan ke backend yang kemudian mengupdate informasi tersebut di *database*. Admin dapat

pula melakukan *logout*, yang mana sistem akan mengonfirmasi bahwa proses *logout* telah berhasil dan mengarahkan kembali admin ke halaman *login*. Diagram ini memberikan gambaran visual tentang alur interaksi admin dalam mengelola berbagai aspek operasional Mahkota *Barbershop* melalui antarmuka *web* administrasi.

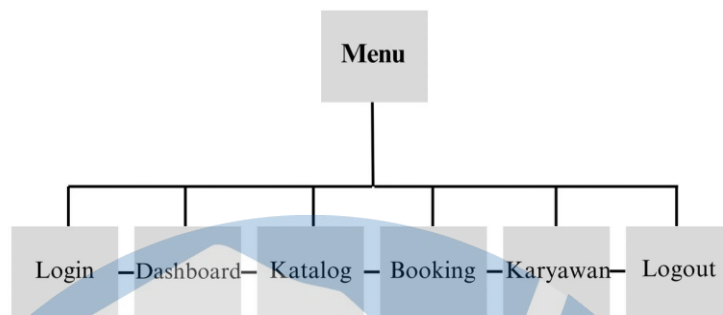
4.1.6 Struktur Menu

Di bagian ini, akan dijelaskan struktur menu untuk sistem *booking online*. Penjelasan ini mencakup organisasi kode program, modul-modul yang digunakan, dan bagaimana modul tersebut saling berinteraksi.



Gambar 4. 7 Struktur Menu Pelanggan

Seperti pada gambar 4.7 menu untuk pelanggan di *website* Mahkota *Barbershop* dirancang untuk memastikan pengalaman pengguna yang efisien dan mudah. Saat mengunjungi halaman utama ("Beranda"), pelanggan akan menemukan informasi terkini tentang *barbershop* serta berita terbaru. Menu "Layanan" memberikan akses cepat ke daftar lengkap layanan yang ditawarkan, lengkap dengan deskripsi dan harga masing-masing layanan. Pelanggan dapat melihat hasil pekerjaan sebelumnya di menu "Portofolio", sementara menu "Harga" memberikan transparansi mengenai biaya layanan. Untuk melakukan reservasi, pelanggan dapat langsung menggunakan menu "Booking Sekarang". Dengan struktur yang jelas ini, pelanggan dapat dengan cepat menemukan informasi yang diperlukan dan melakukan reservasi layanan dengan cepat dan nyaman.

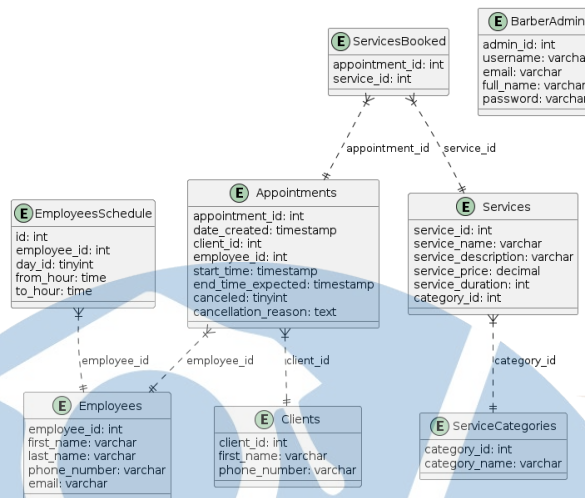


Gambar 4. 8 Struktur Menu Admin

Seperti pada gambar 4.8 menu untuk admin di *website* Mahkota *Barbershop* didesain untuk memfasilitasi manajemen operasional yang efisien. Admin dapat mengakses halaman *login* untuk masuk ke sistem. Setelah berhasil *login*, admin akan diarahkan ke *dashboard* yang menyajikan data penting seperti jumlah *booking* dan informasi terkini lainnya. Menu "Katalog" memungkinkan admin untuk mengelola kategori dan jenis layanan yang ditawarkan oleh *barbershop*. Selain itu, admin dapat mengelola data karyawan seperti informasi kontak dan jadwal kerja melalui menu "Karyawan". Proses manajemen ini didukung dengan fitur *logout* untuk keluar dari sesi admin saat selesai menggunakan sistem. Dengan struktur menu yang terorganisir dengan baik ini, admin dapat mengelola operasional *barbershop* dengan efektif dan efisien.

4.1.7 Class Diagram

Subbab ini menampilkan *class diagram* yang menggambarkan struktur kelas dan hubungan antar kelas dalam sistem *booking online*. *Diagram* ini memberikan gambaran tentang atribut dan metode yang dimiliki oleh setiap kelas.



Gambar 4. 9 Class Diagram

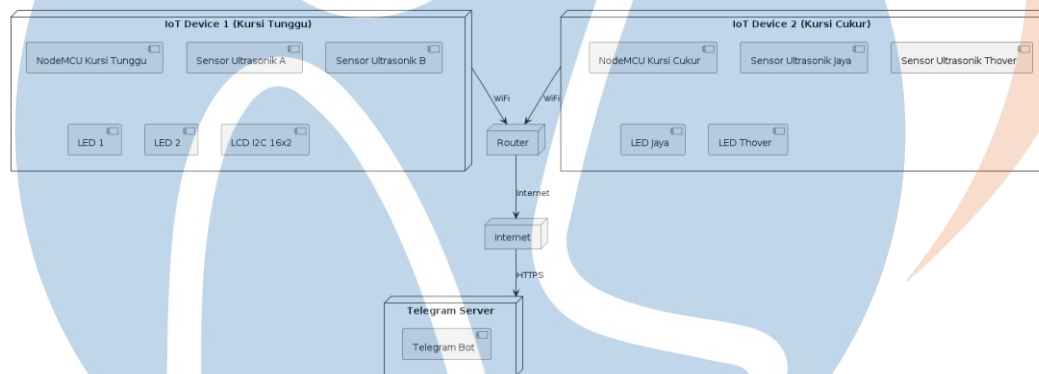
Pada gambar 4.9 merupakan representasi struktur *database* untuk sistem manajemen *booking* pada Mahkota *Barbershop*. Entitas utama yang terlibat termasuk *booking* (*Appointments*), admin (*BarberAdmin*), pelanggan (*Clients*), karyawan (*Employees*), jadwal karyawan (*EmployeesSchedule*), layanan (*Services*), kategori layanan (*ServiceCategories*), dan layanan yang dipesan (*ServicesBooked*).

Appointments menyimpan detail *booking* seperti tanggal pembuatan, pelanggan yang terlibat, karyawan yang bertugas, serta status pembatalan jika ada. *BarberAdmin* mencatat informasi admin yang memiliki akses ke sistem, termasuk detail otentikasi seperti *username* dan *password*. *Clients* berisi informasi pribadi pelanggan seperti nama, dan nomor telepon. *Employees* mencatat informasi karyawan termasuk nama, kontak, dan *email*. *EmployeesSchedule* menangani jadwal kerja karyawan, mencatat jam kerja harian dari setiap karyawan. *Services* berisi daftar layanan yang ditawarkan, dengan detail harga, durasi, dan kategori masing-masing layanan. *ServiceCategories* membagi layanan ke dalam kategori-kategori yang relevan. *ServicesBooked* menyimpan hubungan antara janji temu dan layanan-layanan yang dipesan oleh pelanggan.

Setiap entitas terhubung melalui *foreign key* yang menghubungkan entitas-entitas tersebut secara logis dalam basis data. Diagram ini membantu memvisualisasikan struktur data yang diperlukan untuk mendukung fungsi operasional sistem manajemen *booking* pada Mahkota *Barbershop*, memudahkan pengelolaan informasi dan interaksi antara pelanggan, karyawan, dan admin dalam berbagai aspek layanan dan operasional *barbershop* tersebut.

4.1.8 Arsitektur Sistem IoT

Bagian ini menguraikan arsitektur sistem perangkat IoT yang digunakan di Mahkota *Barbershop*. Penjelasan meliputi komponen-komponen yang digunakan dan bagaimana komponen tersebut saling berhubungan.

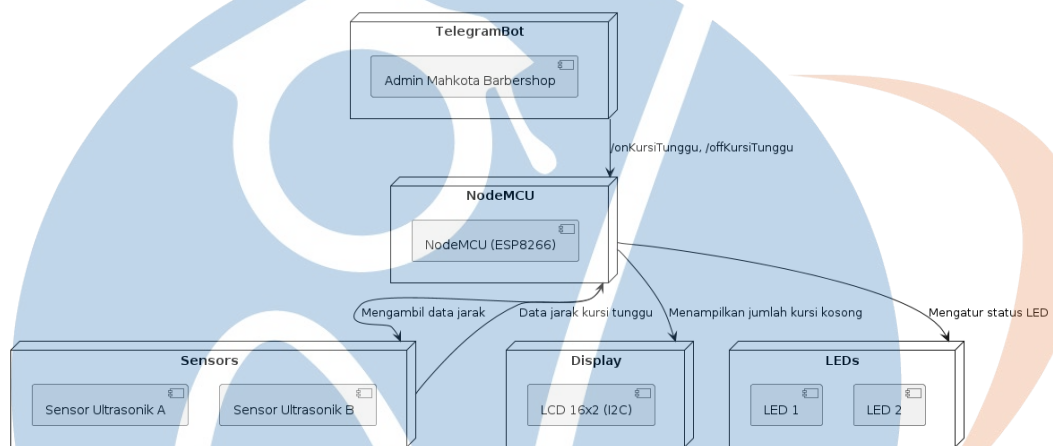


Gambar 4. 10 Topologi Jaringan Antar Perangkat

Seperti pada gambar 4.10 terdiri dari dua perangkat utama yang terhubung ke *internet* dan berkomunikasi dengan *server* Telegram melalui *bot* Telegram. Perangkat pertama adalah *IoT Device 1* yang bertanggung jawab untuk mengelola kursi tunggu. Sistem ini menggunakan NodeMCU sebagai pengontrol utama, yang terhubung dengan dua sensor ultrasonik yang dipasang pada kursi tunggu A dan B. Sensor ini mendeteksi keberadaan orang dengan mengukur jarak.

Perangkat kedua, *IoT Device 2*, bertanggung jawab untuk mengelola kursi cukur. Perangkat ini juga menggunakan NodeMCU sebagai pengontrol utama dan terhubung dengan dua sensor ultrasonik yang dipasang pada kursi cukur Jaya dan Thover. Sensor ini berfungsi mirip dengan yang ada di kursi tunggu, mendeteksi keberadaan orang yang sedang dicukur dan mengirimkan sinyal ke NodeMCU.

Kedua perangkat IoT ini terhubung ke *router* melalui jaringan *WiFi*, yang kemudian menghubungkan mereka ke *internet*. Melalui *internet*, data dari perangkat IoT dikirim ke *server* Telegram. *Server* Telegram berperan penting dalam sistem ini, karena admin Mahkota *Barbershop* menggunakan *bot* Telegram untuk mengirim perintah dan menerima notifikasi.



Gambar 4. 11 Blok Diagram Pendeteksi Jumlah Kursi

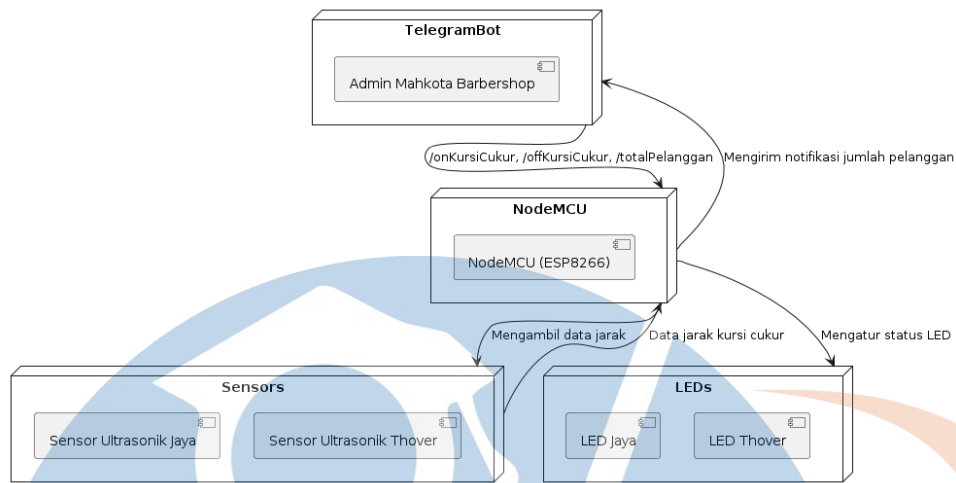
Seperti pada gambar 4.11 pendeteksi jumlah kursi di Mahkota *Barbershop* dirancang untuk memberikan informasi *real-time* tentang ketersediaan kursi tunggu. Sistem ini terdiri dari beberapa bagian utama yang bekerja secara terintegrasi untuk memastikan operasional yang efektif dan efisien. Berikut penjelasan rinci mengenai setiap komponen dalam blok diagram ini:

1. Admin: Admin adalah individu yang bertanggung jawab untuk mengoperasikan sistem perangkat IoT melalui Telegram *Bot*. Admin mengirimkan perintah untuk menyalakan atau mematikan perangkat dan juga memantau status ketersediaan kursi tunggu.
2. Telegram *Bot*: *Bot* ini berfungsi sebagai antarmuka komunikasi antara admin dan sistem perangkat IoT. Telegram *Bot* menerima perintah dari admin, seperti perintah */onKursiTunggu* untuk menyalakan perangkat dan memulai monitoring kursi tunggu, serta perintah */offKursiTunggu* untuk mematikan perangkat dan menghentikan monitoring. *Bot* ini memastikan

bahwa perintah yang diterima diteruskan ke NodeMCU untuk diproses lebih lanjut.

3. NodeMCU (ESP8266): NodeMCU adalah mikrokontroler yang mengendalikan seluruh sistem. Setelah menerima perintah dari Telegram *Bot*, NodeMCU membaca data dari sensor ultrasonik yang terpasang di kursi tunggu A dan B. Selain itu, NodeMCU juga memperbarui status ketersediaan kursi pada layar LCD dan mengatur nyala LED berdasarkan hasil deteksi sensor. NodeMCU terhubung ke jaringan *WiFi* untuk memastikan komunikasi yang lancar dengan Telegram *Bot*.
4. Sensor Ultrasonik A dan B: Dua sensor ultrasonik ini ditempatkan di depan kursi tunggu A dan B untuk mendeteksi jarak antara sensor dan kursi. Jika sensor mendeteksi jarak kurang dari 5 cm, ini menandakan bahwa kursi sedang diduduki oleh pelanggan. Data dari sensor ini kemudian dikirimkan ke NodeMCU untuk diproses dan ditampilkan.
5. LCD 16x2 (I2C): LCD ini berfungsi sebagai layar tampilan yang menunjukkan jumlah kursi kosong di ruang tunggu. Informasi yang ditampilkan membantu pelanggan untuk mengetahui ketersediaan kursi tanpa harus masuk ke ruang tunggu. LCD dioperasikan oleh NodeMCU yang memperbarui tampilan berdasarkan data yang diterima dari sensor ultrasonik.
6. LED: LED digunakan sebagai indikator visual untuk menunjukkan status ketersediaan kursi tunggu. LED akan menyala atau mati berdasarkan jumlah kursi kosong yang terdeteksi oleh sensor. Jika jumlah kursi kosong sedikit (0-2 kursi), LED akan menyala untuk memberikan indikasi visual yang jelas kepada pelanggan dan admin.

Secara keseluruhan, sistem pendeteksi jumlah kursi ini memungkinkan admin Mahkota *Barbershop* untuk mengontrol dan memonitor ketersediaan kursi tunggu secara efisien melalui antarmuka Telegram *Bot*. Dengan penggunaan NodeMCU sebagai pusat kendali, sensor ultrasonik untuk deteksi, LCD untuk tampilan informasi, dan LED untuk indikasi visual, sistem ini memastikan bahwa informasi yang disajikan selalu akurat dan mudah diakses.



Gambar 4. 12 Blok Diagram Pendeteksi Jumlah Pelanggan

Seperti pada gambar 4.12 pendeteksi jumlah pelanggan di Mahkota Barbershop terdiri dari beberapa komponen yang saling terintegrasi untuk menyediakan informasi tentang status kursi cukur dan jumlah pelanggan yang telah dilayani. Berikut adalah penjelasan rinci dari masing-masing komponen dalam blok diagram ini:

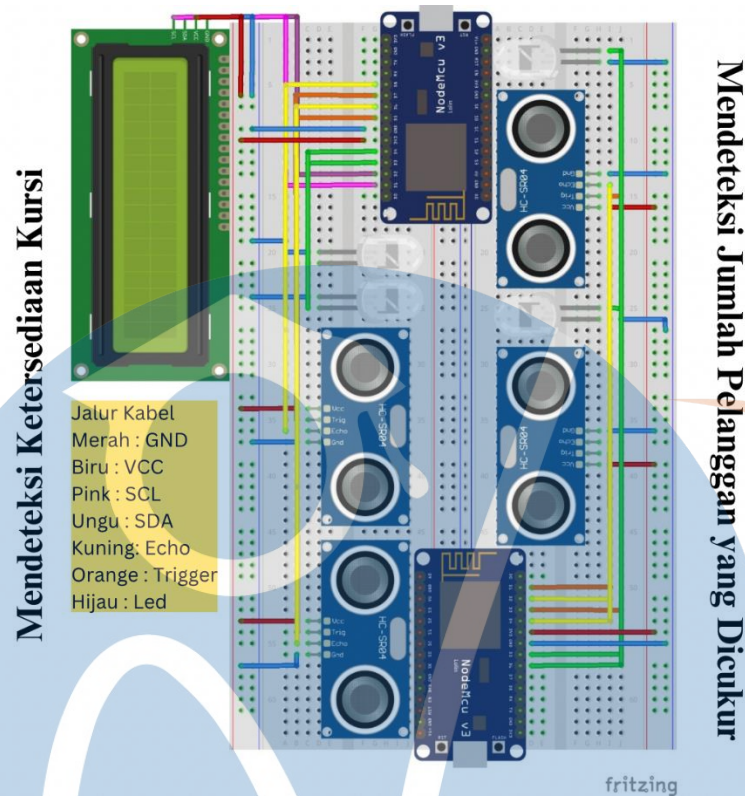
1. *Telegram Bot*: Komponen ini berfungsi sebagai antarmuka antara admin Mahkota Barbershop dan sistem perangkat IoT. *Telegram Bot* memungkinkan admin untuk mengontrol perangkat menggunakan perintah yang dikirimkan melalui aplikasi Telegram. Perintah yang dapat dikirim meliputi `/onKursiCukur` untuk menyalakan perangkat dan memulai monitoring kursi cukur, `/offKursiCukur` untuk mematikan perangkat dan menghentikan monitoring, serta `/totalPelanggan` untuk mendapatkan informasi jumlah total pelanggan yang telah dicukur.
2. *NodeMCU (ESP8266)*: NodeMCU adalah mikrokontroler yang mengontrol seluruh operasi sistem. Perangkat ini menerima perintah dari *Telegram Bot* dan bertindak sesuai dengan instruksi yang diterima. NodeMCU mengelola sensor untuk mengukur jarak kursi cukur, mengatur status LED berdasarkan kondisi kursi, dan mengirimkan notifikasi ke *Telegram Bot*. NodeMCU juga terhubung ke jaringan *WiFi* untuk memastikan komunikasi yang lancar dengan *Telegram Bot*.

3. Sensor: Sistem ini dilengkapi dengan dua sensor ultrasonik. Kedua sensor ini berfungsi untuk mendeteksi jarak antara kursi cukur dan sensor itu sendiri. Jika sensor mendeteksi jarak kurang dari 5 cm, itu berarti kursi sedang diduduki oleh pelanggan yang akan dicukur. Informasi jarak ini dikirim ke NodeMCU untuk diproses lebih lanjut.
4. LED: Sistem ini menggunakan dua LED untuk memberikan indikasi visual tentang status kursi cukur. LED akan menyala jika kursi kosong dan akan mati jika kursi diduduki. Indikasi ini memudahkan pelanggan dan admin untuk mengetahui ketersediaan kursi cukur secara langsung.
5. Notifikasi Telegram: Setelah memproses data dari sensor, NodeMCU mengirimkan notifikasi ke Telegram *Bot* tentang jumlah pelanggan yang telah dicukur. Notifikasi ini juga membantu pemilik untuk memantau aktivitas Mahkota *Barbershop* secara *real-time* tanpa harus berada di tempat.

Secara keseluruhan, sistem ini memungkinkan admin Mahkota *Barbershop* untuk mengontrol dan memantau status kursi cukur serta jumlah pelanggan yang telah dilayani dengan mudah melalui antarmuka Telegram *Bot*. Dengan menggunakan NodeMCU sebagai pusat kendali dan sensor ultrasonik untuk deteksi jarak, sistem ini memastikan data yang diberikan benar dan terkini. LED memberikan indikasi visual yang langsung terlihat, dan notifikasi telegram memastikan admin selalu mendapatkan *update* terbaru mengenai operasional Mahkota *Barbershop*.

4.1.9 Rangkaian Sistem IoT

Pada subbab ini, akan dijelaskan rangkaian elektronik dari sistem perangkat IoT yang digunakan. Rangkaian sistem ini mencakup skema *wiring* dan koneksi antara berbagai komponen seperti pada gambar 4.13.



Gambar 4. 13 Rangkaian Sistem IoT

Rangkaian sistem IoT untuk Mahkota *Barbershop* dirancang untuk mendeteksi jumlah kursi yang kosong di area tunggu dan jumlah pelanggan yang sedang menggunakan kursi cukur. Meskipun kedua sistem ini bekerja secara terpisah, keduanya terintegrasi dalam satu *Printed Circuit Board (PCB)* untuk efisiensi dan kemudahan pengelolaan. Berikut adalah penjelasan rinci mengenai setiap komponen dalam kedua rangkaian ini:

1. Rangkaian Pendeteksi Jumlah Kursi

a. Sensor Ultrasonik pada Kursi Tunggu A:

- TRIGGER: Sinyal pemicu untuk mengukur jarak dikirim dari PIN D5 pada NodeMCU.
- ECHO: Sinyal pantulan diterima di PIN D6, yang digunakan untuk menghitung jarak.
- VCC: Sensor diberi daya dari pin 3V pada NodeMCU.

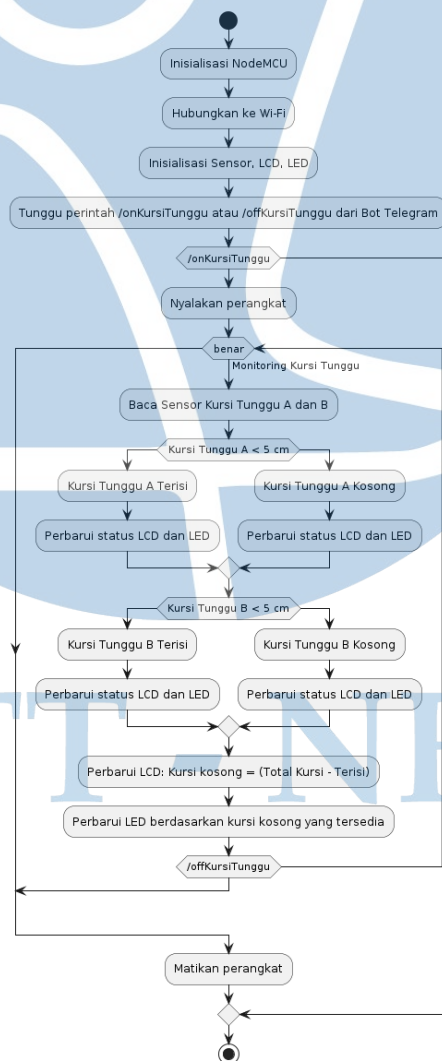
- GND: Sensor dihubungkan ke *ground* untuk melengkapi sirkuit listrik.
- b. Sensor Ultrasonik pada Kursi Tunggu B:
- TRIGGER: Sinyal pemicu dikirim dari PIN D7.
 - ECHO: Sinyal pantulan diterima di PIN D8.
 - VCC: Sensor diberi daya dari pin 3V pada NodeMCU.
 - GND: Sensor dihubungkan ke *ground*.
- c. LED:
- LED: LED pertama dihubungkan ke PIN D3 untuk menunjukkan status ketersediaan kursi.
 - LED: LED kedua dihubungkan ke PIN D4 untuk fungsi yang sama.
- d. LCD:
- SDA: Jalur data serial SDA dihubungkan ke PIN D2 untuk komunikasi I2C.
 - SCL: Jalur clock serial SCL dihubungkan ke PIN D1.
 - VCC: LCD diberi daya dari pin 3V pada NodeMCU.
 - GND: LCD dihubungkan ke *ground*.
2. Rangkaian Pendeteksi Jumlah Pelanggan
- a. Sensor Ultrasonik pada Kursi Cukur Jaya:
- TRIGGER: Sinyal pemicu dikirim dari PIN D1 pada NodeMCU.
 - ECHO: Sinyal pantulan diterima di PIN D2.
 - VCC: Sensor diberi daya dari pin 3V pada NodeMCU.
 - GND: Sensor dihubungkan ke *ground*.
- b. Sensor Ultrasonik pada Kursi Cukur Thover:
- TRIGGER: Sinyal pemicu dikirim dari PIN D3.
 - ECHO: Sinyal pantulan diterima di PIN D4.
 - VCC: Sensor diberi daya dari pin 3V pada NodeMCU.
 - GND: Sensor dihubungkan ke *ground*.

c. LED:

- LED 1: LED dihubungkan ke PIN D5 untuk menunjukkan status kursi cukur Jaya.
- LED 2: LED dihubungkan ke PIN D6 untuk menunjukkan status kursi cukur Thover.

4.1.10 Flowchart Sistem IoT

Bagian ini menyajikan *flowchart* yang menggambarkan alur proses dari sistem perangkat IoT. *Flowchart* ini penting untuk memahami logika operasi dan alur kerja sistem.



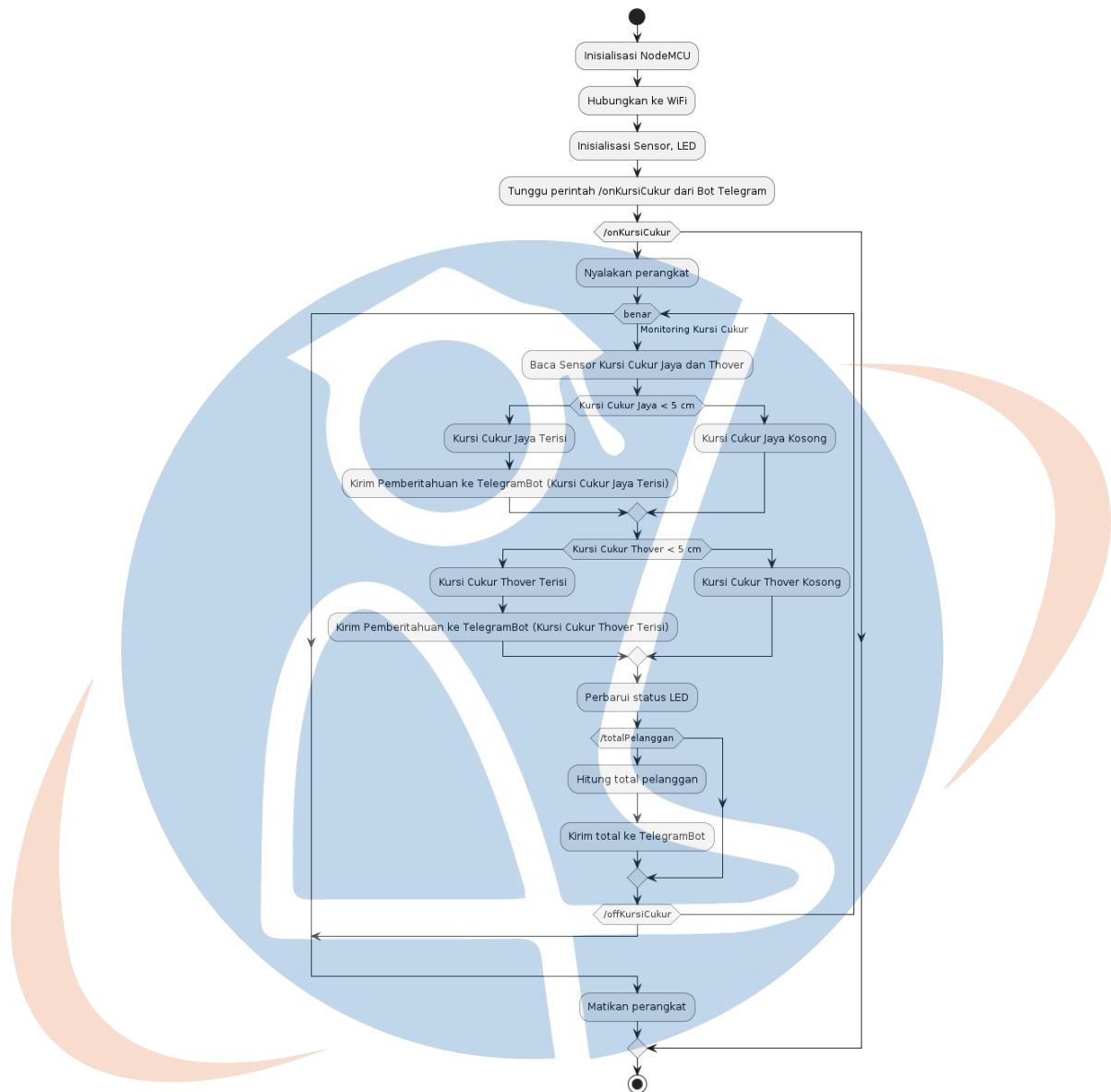
Gambar 4. 14 Flowchart Pendeteksi Jumlah Kursi

Seperti pada gambar 4.14 proses dimulai dengan inisialisasi NodeMCU, di mana perangkat ini dinyalakan dengan menghubungkan ke jaringan *WiFi* yang telah dikonfigurasi. Selanjutnya, sensor ultrasonik dan LED yang akan digunakan untuk monitoring kursi tunggu dipersiapkan. Setelah inisialisasi selesai, sistem memasuki tahap menunggu perintah dari *bot* Telegram. Pada tahap ini, sistem siap menerima perintah */onKursiTunggu* untuk menyalakan perangkat dan memulai monitoring atau perintah */offKursiTunggu* untuk mematikan perangkat.

Ketika sistem menerima perintah */onKursiTunggu*, perangkat akan menyala dan memulai proses monitoring kursi tunggu A dan B. Sistem secara berkala membaca data jarak yang dikirim oleh sensor ultrasonik yang dipasang di kursi-kursi tersebut. Jika sensor mendeteksi jarak kurang dari 5 cm antara sensor dan kursi, hal ini menunjukkan bahwa kursi sedang diduduki. Sistem kemudian mengupdate jumlah kursi kosong yang ditampilkan pada LCD yang terletak di pintu masuk, sehingga pelanggan dapat dengan mudah mengetahui kondisi ketersediaan kursi tunggu.

Selain itu, sistem juga mengatur status LED berdasarkan jumlah kursi kosong. Jika semua kursi terisi, LED akan menunjukkan kondisi tersebut, dan sebaliknya, jika ada kursi kosong, LED akan menunjukkan jumlah kursi yang tersedia. Hal ini memberikan indikasi visual yang mudah dipahami oleh pelanggan.

Jika sistem menerima perintah */offKursiTunggu* dari *bot* Telegram, perangkat akan dimatikan dan proses monitoring kursi tunggu dihentikan. Setelah perintah ini dijalankan, sistem akan berhenti dan kembali ke keadaan awal, siap untuk menerima perintah berikutnya dari *bot* Telegram. Proses berhenti ini memastikan bahwa sistem hanya aktif dan melakukan monitoring ketika diperlukan, sesuai dengan perintah yang diberikan oleh admin melalui *bot* Telegram.



Gambar 4. 15 Flowchart Pendeteksi Jumlah Pelanggan

Seperti pada gambar 4.15 proses dimulai dengan inisialisasi NodeMCU, di mana perangkat dihidupkan dengan menghubungkan ke jaringan *WiFi* yang telah dikonfigurasi, serta mempersiapkan sensor ultrasonik dan LED yang akan digunakan. Setelah inisialisasi selesai, sistem masuk ke tahap menunggu perintah dari *bot* Telegram, khususnya perintah */onKursiCukur* untuk mulai menyalakan perangkat dan memulai proses monitoring kursi cukur.

Ketika sistem menerima perintah */onKursiCukur*, perangkat akan menyala dan mulai memonitor kursi cukur Jaya dan Thover. Sistem secara berkala membaca data jarak yang dikirim oleh sensor ultrasonik yang ditempatkan di kursi-kursi tersebut. Jika sensor mendeteksi jarak kurang dari 5 cm antara sensor dan kursi, hal ini menunjukkan bahwa kursi sedang diduduki. Sistem kemudian mengirimkan notifikasi ke *bot* Telegram yang memberi tahu status kursi tersebut, apakah sedang diduduki atau tidak. Selain itu, status LED juga akan diperbarui untuk menunjukkan kondisi kursi, memberikan indikasi visual apakah kursi sedang digunakan atau kosong.

Selama proses monitoring, jika sistem menerima perintah */totalPelanggan* dari *bot* Telegram, sistem akan menghitung total pelanggan yang telah dicukur dan mengirimkan hasil perhitungan ini kembali ke *bot* Telegram. Ini memungkinkan admin untuk mengetahui jumlah pelanggan yang telah dilayani.

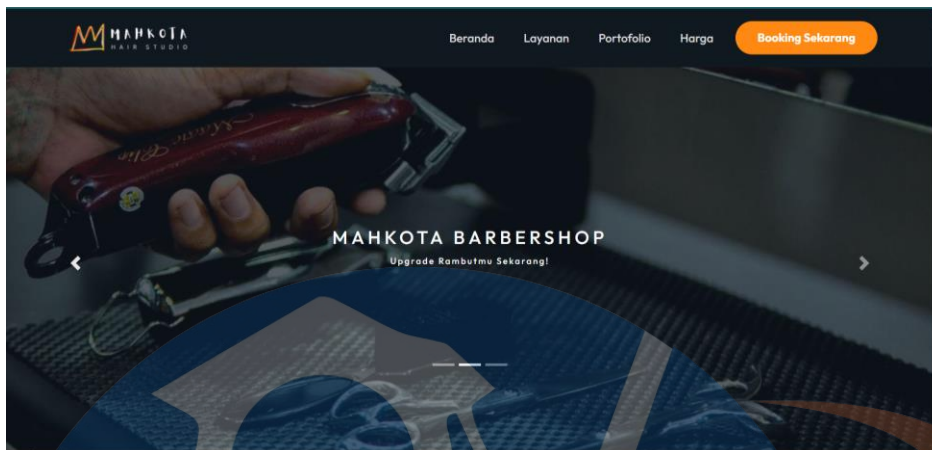
Jika sistem menerima perintah */offKursiCukur*, perangkat akan dimatikan dan proses monitoring kursi cukur dihentikan. Setelah perintah ini dijalankan, sistem berhenti dan kembali ke keadaan awal, siap untuk menerima perintah berikutnya dari *bot* Telegram. Proses berhenti ini memastikan bahwa sistem hanya aktif dan memonitor saat diperlukan, sesuai dengan perintah yang diberikan oleh admin melalui *bot* Telegram.

4.2 Implementasi Rancangan Sistem

Pada bagian ini, penulis akan membahas implementasi dari rancangan sistem yang telah dirancang sebelumnya. Penulis akan menguraikan hasil dari dua aspek utama dari sistem yang dirancang, yaitu sistem *booking online* untuk Mahkota *Barbershop* dan prototipe perangkat IoT yang mendeteksi jumlah ketersediaan kursi tunggu beserta jumlah pelanggan yang berhasil dicukur.

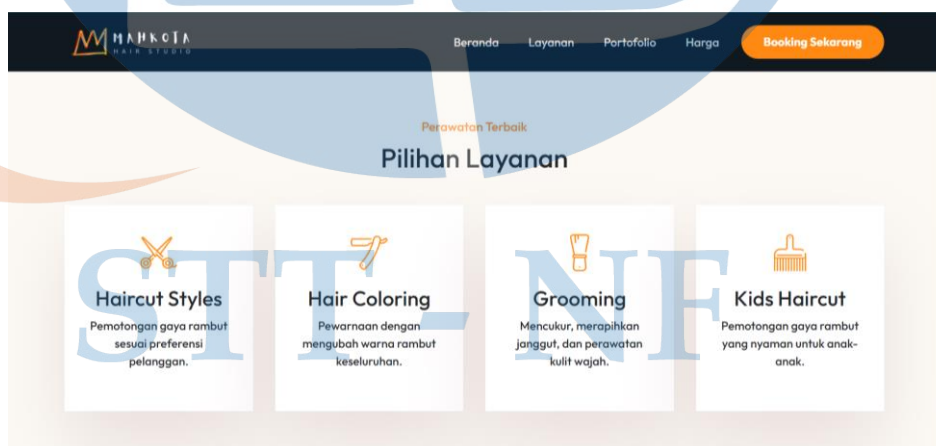
4.2.1 Hasil Sistem *Booking Online* Pelanggan

Penulis memulai dengan mengeksplorasi hasil dari implementasi sistem *booking online* untuk Mahkota *Barbershop*. Sistem beroperasi secara praktis dan mengetahui detail-fitur yang telah diimplementasikan untuk memenuhi kebutuhan pengguna.



Gambar 4. 16 Halaman Beranda Pelanggan

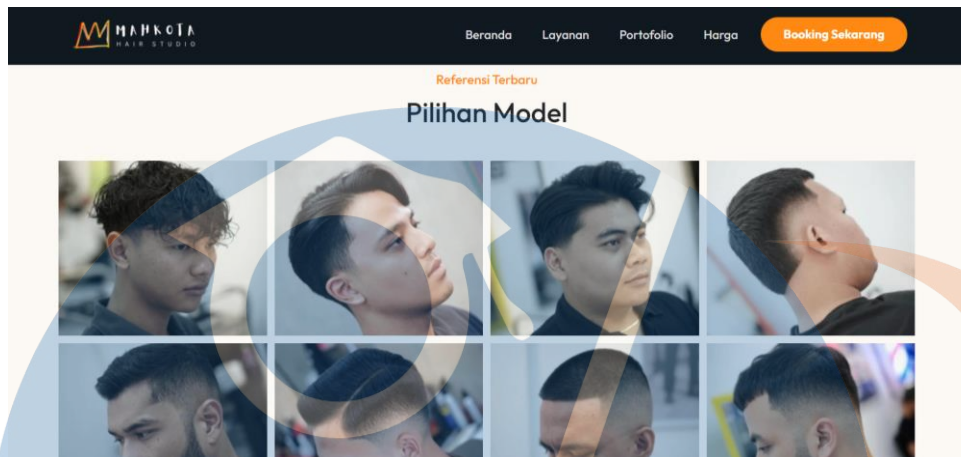
Pada gambar 4.16 menunjukkan halaman beranda pelanggan. Pada halaman ini terdapat *carousel* yang menampilkan berbagai gambar tentang *barbershop*, memberikan kesan visual yang menarik dan dinamis. Di tengah halaman, terdapat tulisan besar "Mahkota *Barbershop*" yang mencolok dan di bawahnya terdapat slogan "Upgrade Rambutmu Sekarang," yang mengajak pelanggan untuk memperbaiki penampilan rambut mereka dengan layanan yang ditawarkan oleh Mahkota *Barbershop*.



Gambar 4. 17 Halaman Layanan

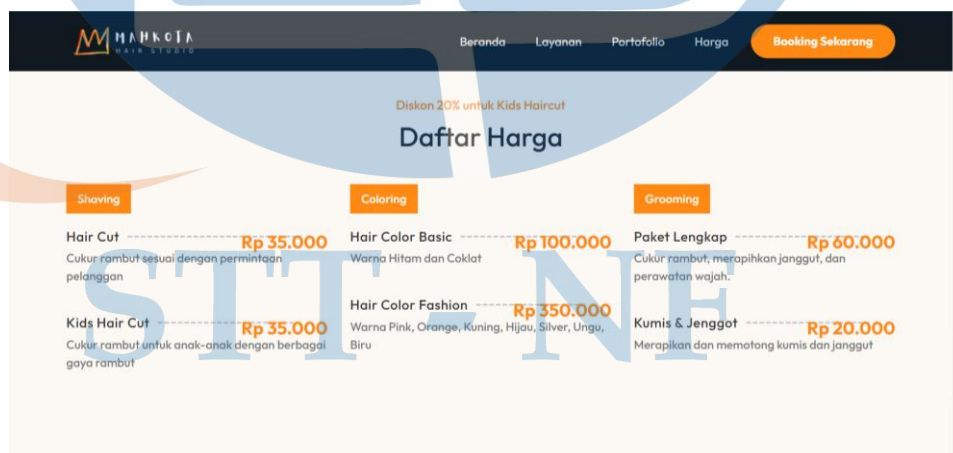
Pada gambar 4.17 menampilkan halaman layanan. Halaman ini menyediakan informasi detail mengenai berbagai layanan yang ditawarkan oleh *barbershop*, seperti gaya potongan rambut (*haircut styles*), pewarnaan rambut

(*hair coloring*), perawatan (*grooming*), dan potongan rambut untuk anak-anak (*kids haircuts*).



Gambar 4. 18 Halaman Portofolio

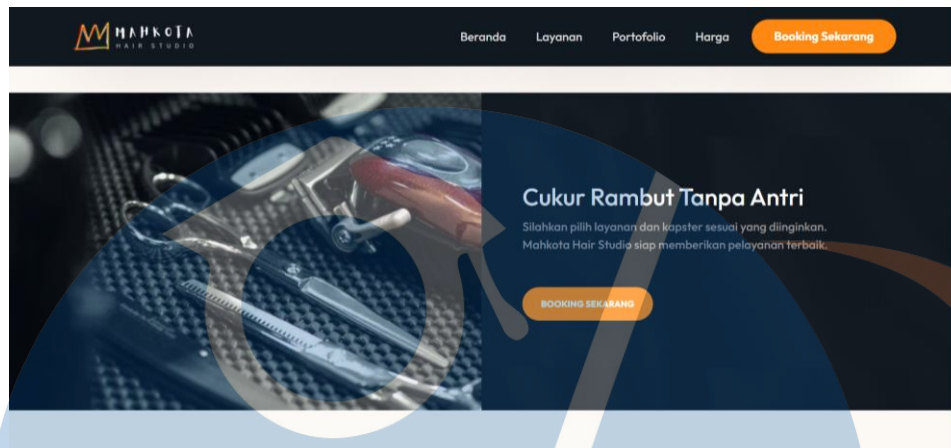
Pada gambar 4.18 memperlihatkan halaman portofolio. Di halaman ini, terdapat berbagai foto model rambut yang telah berhasil dicukur, memberikan pelanggan referensi mengenai kualitas dan gaya yang dapat diharapkan dari layanan yang ditawarkan.



Gambar 4. 19 Halaman Daftar Harga

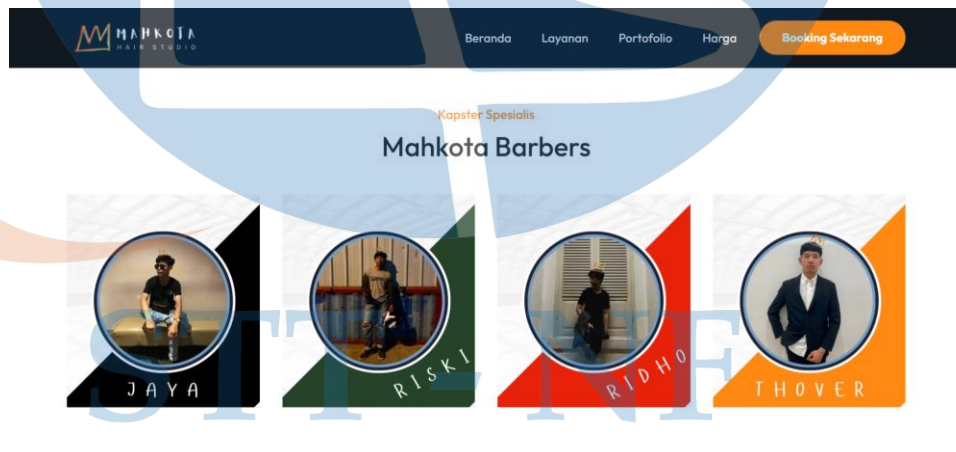
Pada gambar 4.19 menunjukkan halaman daftar harga. Halaman ini memuat informasi mengenai berbagai layanan dan harga yang ditawarkan,

termasuk harga untuk potongan rambut (*hair cut*), potongan rambut anak-anak (*kids hair cut*), pewarnaan rambut (*hair color*), dan paket lengkap.



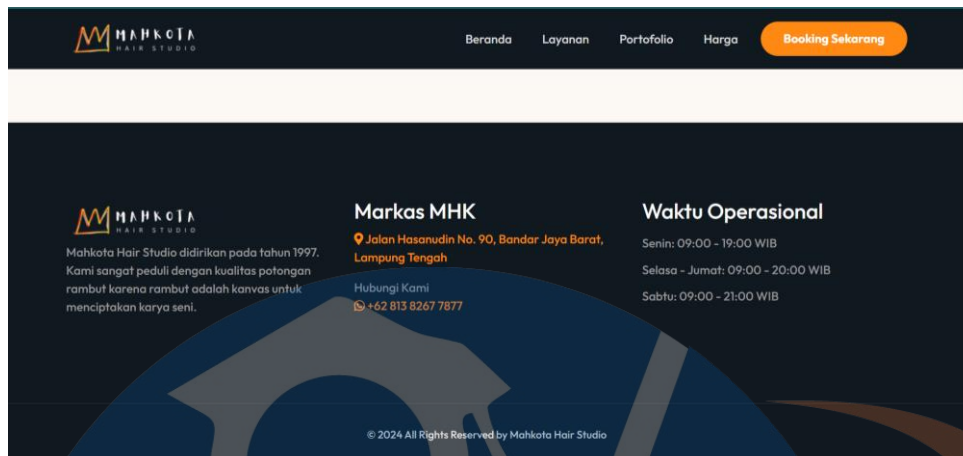
Gambar 4. 20 Halaman Tampilan Booking

Pada gambar 4.20 menggambarkan halaman tampilan *booking*. Halaman ini mengajak pelanggan untuk melakukan *booking* layanan dengan mengisi data diri mereka, sehingga proses reservasi dapat dilakukan dengan mudah dan efisien.



Gambar 4. 21 Halaman Tampilan Kapster

Pada gambar 4.21 menampilkan halaman kapster. Halaman ini memberikan informasi mengenai karyawan cukur, termasuk foto dan nama mereka, sehingga pelanggan dapat memilih kapster favorit atau yang mereka inginkan untuk layanan cukur.



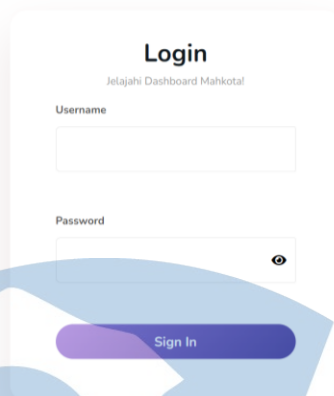
Gambar 4. 22 Halaman Tampilan Informasi

Pada gambar 4.22 menunjukkan halaman tampilan informasi. Pada halaman ini, pelanggan dapat menemukan informasi penting seperti alamat *barbershop*, nomor telepon yang dapat dihubungi, dan jam operasional, sehingga memudahkan pelanggan untuk mengetahui lokasi dan waktu layanan yang tersedia.

4.2.2 Hasil Sistem *Booking Online Admin*

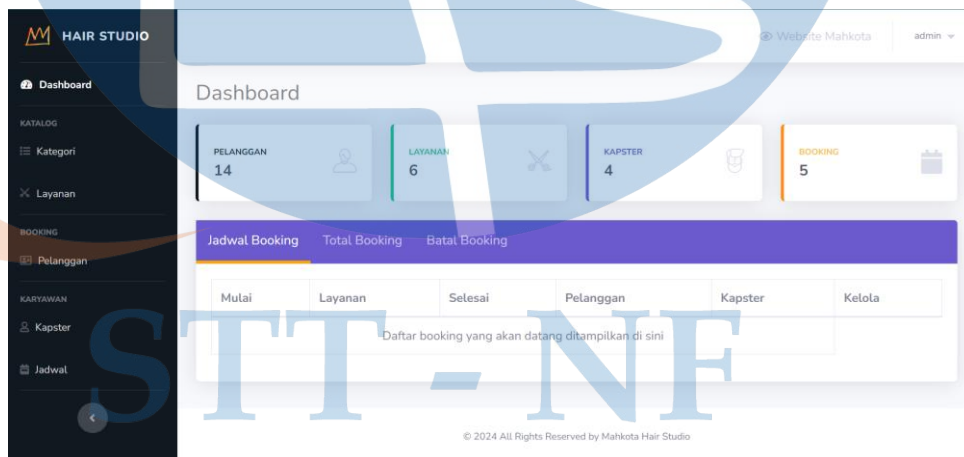
Dalam bagian ini, penulis akan membahas hasil dari implementasi sistem *booking online* untuk Mahkota *Barbershop* dari perspektif admin. Sistem beroperasi secara praktis dan mengetahui detail-fitur yang telah diimplementasikan untuk memenuhi kebutuhan administratif dan manajemen Mahkota *Barbershop*.

STT - NF



Gambar 4. 23 Halaman Login Admin

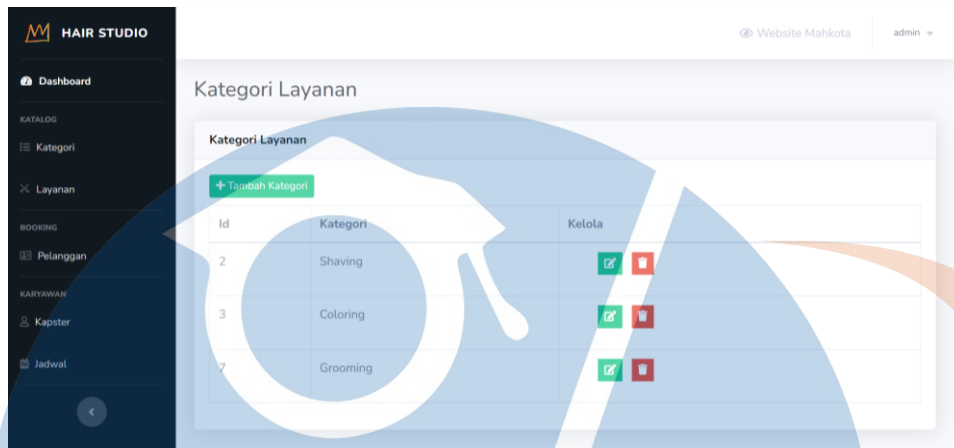
Pada gambar 4.23 menunjukkan halaman *login* admin. Pada halaman ini, admin diharuskan mengisi *username* dan *password* untuk dapat mengakses sistem. Halaman ini memastikan bahwa hanya admin yang memiliki kredensial benar yang dapat ditambahkan ke sistem untuk manajemen data dan informasi Mahkota Barbershop.



Gambar 4. 24 Halaman Dashboard

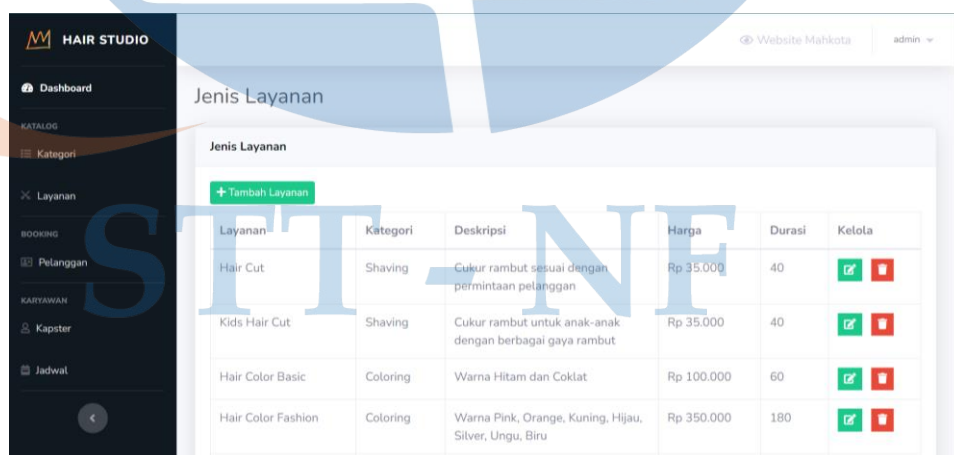
Pada gambar 4.24 menampilkan halaman *dashboard*. Halaman ini memberikan informasi ringkas dan penting berupa jumlah pelanggan, jumlah layanan, jumlah karyawan, dan jumlah *booking*. Selain itu, terdapat juga informasi terkait jadwal *booking* atau *booking* yang masuk serta informasi terkait

pembatalan yang dilakukan admin. *Dashboard* ini memberikan gambaran umum tentang aktivitas dan status operasional Mahkota *Barbershop*.



Gambar 4. 25 Halaman Kategori Layanan

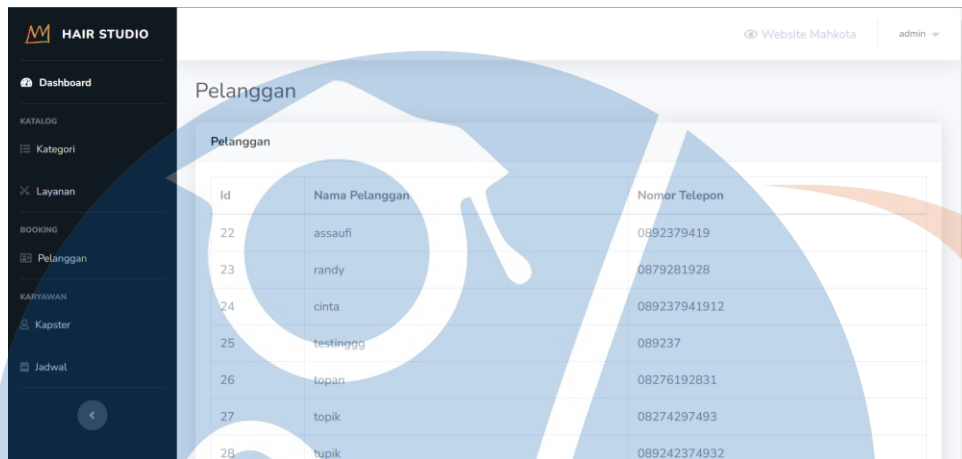
Pada gambar 4.25 memperlihatkan halaman kategori layanan. Pada halaman ini, admin dapat melihat berbagai kategori layanan yang saat ini ada, seperti *shaving*, *coloring*, dan *grooming*. Admin juga memiliki kemampuan untuk menambahkan kategori layanan baru, mengedit kategori yang ada, atau menghapus kategori yang tidak diperlukan.



Gambar 4. 26 Halaman Jenis Layanan

Pada gambar 4.26 menunjukkan halaman jenis layanan. Halaman ini memuat informasi detail mengenai layanan yang ditawarkan, termasuk nama

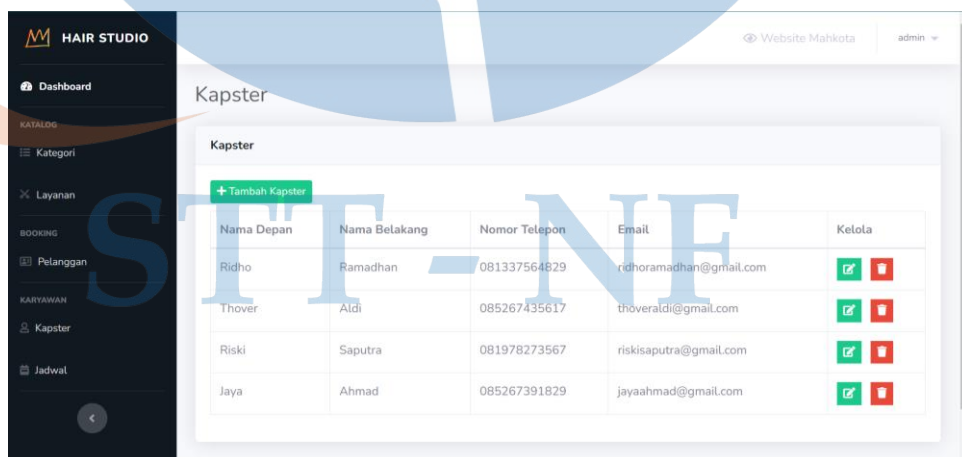
layanan, kategori layanan, deskripsi, harga, dan durasi pengerjaan. Admin juga memiliki opsi untuk menambahkan layanan baru, menghapus layanan yang ada, atau mengedit informasi layanan.











Id	Nama Pelanggan	Nomor Telepon
22	assaufi	0892379419
23	randy	0879281928
24	cinta	089237941912
25	testinggg	089237
26	lopan	08276192831
27	topik	08274297493
28	tupik	089242374932

Gambar 4. 27 Halaman Data Booking Pelanggan

Pada gambar 4.27 menggambarkan halaman data *booking* pelanggan. Di halaman ini, terdapat informasi terkait data pelanggan seperti nama dan nomor telepon dari pelanggan yang pernah melakukan *booking*. Halaman ini membantu admin untuk melacak dan mengelola informasi kontak pelanggan dengan mudah.

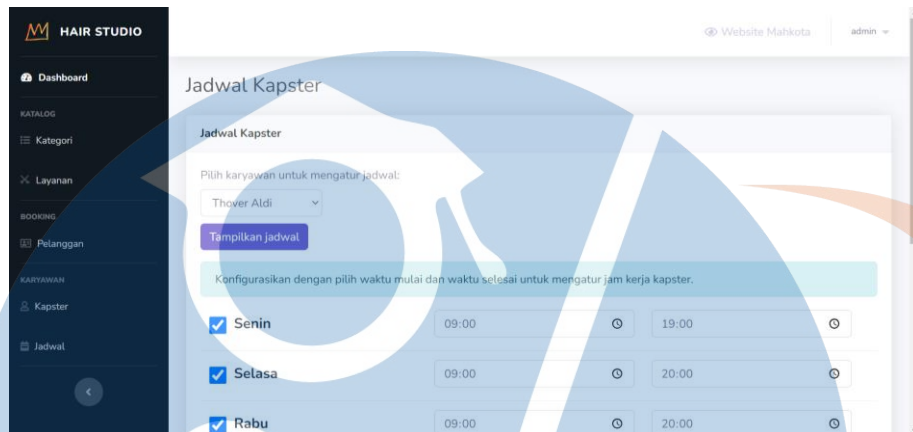


Nama Depan	Nama Belakang	Nomor Telepon	Email	Kelola
Ridho	Ramadhan	081337564829	ridhoramadhan@gmail.com	 
Thover	Aldi	085267435617	thoveraldi@gmail.com	 
Riski	Saputra	081978273567	riskisaputra@gmail.com	 
Jaya	Ahmad	085267391829	jayaahmad@gmail.com	 

Gambar 4. 28 Halaman Data Kapster

Pada gambar 4.28 menampilkan halaman data kapster. Halaman ini berisi informasi mengenai karyawan cukur, termasuk nama depan, nama

belakang, nomor telepon, dan *email*. Admin juga dapat menambahkan karyawan baru, menghapus karyawan yang sudah tidak bekerja, atau mengedit informasi karyawan yang ada.



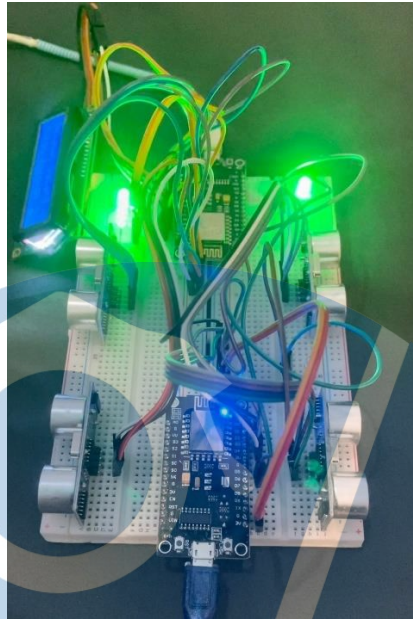
Gambar 4. 29 Halaman Jadwal Kapster

Pada gambar 4.29 menunjukkan halaman jadwal kapster. Pada halaman ini, terdapat informasi mengenai waktu ketersediaan karyawan. Admin dapat mengatur hari dan jam kerja dari karyawan, sehingga memudahkan dalam pengelolaan jadwal kerja dan memastikan ketersediaan layanan bagi pelanggan.

4.2.3 Hasil Prototipe Perangkat IoT

Bagian ini akan menyajikan rangkaian hasil dari prototipe perangkat IoT yang dirancang untuk mendeteksi jumlah ketersediaan kursi dan jumlah pelanggan yang berhasil dicukur di Mahkota *Barbershop*.

STT - NF



Gambar 4. 30 Prototipe Perangkat IoT

Pada gambar 4.30 menunjukkan prototipe hasil dari rangkaian perangkat IoT untuk mendeteksi ketersediaan jumlah kursi tunggu dan jumlah pelanggan yang berhasil dicukur. Terdapat komponen yang digunakan 2 NODEMCU ESP8266, 4 Sensor Ultrasonik, 4 LED, dan 1 LCD ukuran 16x2.

4.2.4 Kode Program Prototipe Perangkat IoT

Penulis akan menyajikan kode program tentang logika dan alur kerja yang digunakan untuk mengoperasikan prototipe perangkat IoT ini.

```
const char* ssid = "Cantik";  
const char* password = "Permisi";  
const char* botToken = "7088790416:AAHRFEuZ0Nr9ApCwCLN1Klowx23iN-aw-1s";  
const int64 t chatId = 649756578;  
  
CTBot myBot;
```

Gambar 4. 31 Inisialisasi WiFi dan Koneksi ke Bot Telegram

Pada gambar 4.31 kode ini untuk menginisialisasi koneksi *WiFi* ke jaringan (*ssid* dan *password*), serta mengatur *token bot* Telegram (*botToken*) dan ID obrolan (*chatId*). Hal ini memungkinkan perangkat untuk terhubung dan berkomunikasi dengan *bot* Telegram.

```

void setup() {
  Serial.begin(9600);
  pinMode(LED_PIN1, OUTPUT);
  pinMode(LED_PIN2, OUTPUT);
  connectToWiFi();
  lcd.begin();
}

```

Gambar 4. 32 Fungsi Setup Jumlah Kursi

Pada gambar 4.32 fungsi *setup()* adalah langkah awal yang penting saat perangkat dinyalakan. Di dalamnya, serial komunikasi diinisialisasi, pin LED diatur sebagai *output*, dilakukan koneksi ke *WiFi*, dan LCD diinisialisasi untuk menampilkan informasi perangkat.

```

void loop() {
  handleTelegramMessages();
  if (perangkatAktif) {
    unsigned int distance3 = ultrasonic3.read();
    unsigned int distance4 = ultrasonic4.read();
    jumlahKursiKosong = 0;
    if (distance3 > MAX_DISTANCE) jumlahKursiKosong++;
    if (distance4 > MAX_DISTANCE) jumlahKursiKosong++;
    lcd.setCursor(0, 1);
    lcd.print("Kursi Kosong: ");
    lcd.print(jumlahKursiKosong);
    if (jumlahKursiKosong == 0) {
      digitalWrite(LED_PIN1, LOW); // Matikan LED jika tidak ada kursi kosong
      digitalWrite(LED_PIN2, LOW); // Matikan LED jika tidak ada kursi kosong
    } else if (jumlahKursiKosong == 1) {
      digitalWrite(LED_PIN1, LOW); // Nyalakan LED1 jika hanya ada 1 kursi kosong
      digitalWrite(LED_PIN2, HIGH); // Matikan LED2 jika hanya ada 1 kursi kosong
    } else if (jumlahKursiKosong == 2) {
      digitalWrite(LED_PIN1, HIGH); // Nyalakan LED1 jika ada 2 kursi kosong
      digitalWrite(LED_PIN2, HIGH); // Nyalakan LED2 jika ada 2 kursi kosong
    }
  }

  if (kursiTungguAStatus != kursiTungguAStatusPrev) {
    if (kursiTungguAStatus) {
      sendTelegramMessage("Kursi Tunggu A = Seseorang sedang duduk");
      digitalWrite(LED_PIN1, LOW);
    } else {
      sendTelegramMessage("Kursi Tunggu A = Tidak ada yang duduk");
      digitalWrite(LED_PIN1, HIGH);
    }
    kursiTungguAStatusPrev = kursiTungguAStatus;
  }

  if (kursiTungguBStatus != kursiTungguBStatusPrev) {
    if (kursiTungguBStatus) {
      sendTelegramMessage("Kursi Tunggu B = Seseorang sedang duduk");
      digitalWrite(LED_PIN2, LOW);
    } else {
      sendTelegramMessage("Kursi Tunggu B = Tidak ada yang duduk");
      digitalWrite(LED_PIN2, HIGH);
    }
    kursiTungguBStatusPrev = kursiTungguBStatus;
  }
  delay(100);
}

```

Gambar 4. 33 Fungsi Loop Jumlah Kursi

Pada gambar 4.33 fungsi *loop()* adalah inti dari program yang berjalan secara terus-menerus. Ini mengurus pengolahan pesan Telegram, membaca sensor ultrasonik untuk menghitung jumlah kursi kosong, mengontrol LED untuk

menunjukkan status, dan mengirimkan pesan Telegram jika status kursi tunggu berubah.

```
void handleTelegramMessages() {
  TMessage msg;
  if (myBot.getNewMessage(msg)) {
    String command = msg.text;
    if (command == "/onKursiTunggu") {
      perangkatAktif = true;
      sendTelegramMessage("Perangkat telah diaktifkan.");
      // Hidupkan kembali LCD
      lcd.backlight();
      lcd.clear();
      lcd.setCursor(0, 0);
      lcd.print("Selamat Datang");
    } else if (command == "/offKursiTunggu") {
      perangkatAktif = false;
      sendTelegramMessage("Perangkat telah dinonaktifkan.");
      digitalWrite(LED_PIN1, LOW);
      digitalWrite(LED_PIN2, LOW);
      // Matikan LCD
      lcd.clear();
      lcd.noBacklight();
    } else if (command == "/cekStatus") {
      jumlahKursiKosong = 0;
      if (!kursiTungguAStatus) {
        jumlahKursiKosong++;
      }
      if (!kursiTungguBStatus) {
        jumlahKursiKosong++;
      }
      String response = "Jumlah Kursi Kosong: " + String(jumlahKursiKosong);
      sendTelegramMessage(response);

      // Kirim status kursi kosong atau terisi
      if (kursiTungguAStatus) {
        sendTelegramMessage("Kursi Tunggu A = Seseorang sedang duduk");
      } else {
        sendTelegramMessage("Kursi Tunggu A = Tidak ada yang duduk");
      }

      if (kursiTungguBStatus) {
        sendTelegramMessage("Kursi Tunggu B = Seseorang sedang duduk");
      } else {
        sendTelegramMessage("Kursi Tunggu B = Tidak ada yang duduk");
      }
    }
  }
}
```

Gambar 4. 34 Fungsi *handleTelegramMessages* Jumlah Kursi

Pada gambar 4.34 fungsi *handleTelegramMessages()* mengelola pesan Telegram yang diterima dari bot. Ini memungkinkan pengguna untuk mengontrol aktivasi perangkat, memeriksa status kursi kosong, dan mendapatkan informasi tentang status kursi tunggu A dan B.

```
void setup() {
  Serial.begin(9600);
  pinMode(LED_PIN1, OUTPUT);
  pinMode(LED_PIN2, OUTPUT);

  connectToWiFi();
}
```

Gambar 4. 35 Fungsi *Setup* Jumlah Pelanggan

Pada gambar 4.35 fungsi *setup()* melakukan inialisasi awal saat perangkat dinyalakan. Serial komunikasi diatur pada 9600 bps, dan pin LED (LED_PIN1 dan LED_PIN2) diatur sebagai *output*. Fungsi *connectToWiFi()* dipanggil untuk menghubungkan perangkat ke jaringan *WiFi*.

```
void loop() {
  handleTelegramMessages(); // Memproses pesan Telegram
  // Cek apakah perangkat aktif sebelum membaca sensor
  if (perangkatAktif) {
    unsigned int distance1 = ultrasonic1.read();
    unsigned int distance2 = ultrasonic2.read();
    if (distance1 <= MAX_DISTANCE) {
      kursiCukurJaya = true;
    } else {
      kursiCukurJaya = false;
    }
    if (distance2 <= MAX_DISTANCE) {
      kursiCukurThover = true;
    } else {
      kursiCukurThover = false;
    }
    if (kursiCukurJaya != kursiCukurJayaPrev) {
      if (kursiCukurJaya) {
        sendTelegramMessage("Kursi Cukur Jaya = Seseorang sedang duduk");
        digitalWrite(LED_PIN1, LOW);
        jumlahTerisiKursiCukurJaya++; // Tambah jumlah terisi kursi Cukur Jaya
      } else {
        sendTelegramMessage("Kursi Cukur Jaya = Tidak ada yang duduk");
        digitalWrite(LED_PIN1, HIGH);
      }
      kursiCukurJayaPrev = kursiCukurJaya;
    }
    if (kursiCukurThover != kursiCukurThoverPrev) {
      if (kursiCukurThover) {
        sendTelegramMessage("Kursi Cukur Thover = Seseorang sedang duduk");
        digitalWrite(LED_PIN2, LOW);
        jumlahTerisiKursiCukurThover++; // Tambah jumlah terisi kursi Cukur Thover
      } else {
        sendTelegramMessage("Kursi Cukur Thover = Tidak ada yang duduk");
        digitalWrite(LED_PIN2, HIGH);
      }
      kursiCukurThoverPrev = kursiCukurThover;
    }
  }
  delay(50); // Tunggu sebentar sebelum membaca sensor lagi
}
```

Gambar 4. 36 Fungsi Loop Jumlah Pelanggan

Pada gambar 4.36 fungsi *loop()* berjalan secara terus-menerus untuk memproses pesan Telegram dan membaca sensor ultrasonik. Jika perangkat aktif (perangkatAktif), itu akan membaca jarak dari sensor dan memperbarui status kursi (kursiCukurJaya dan kursiCukurThover). Jika terjadi perubahan status, notifikasi dikirim ke Telegram dan status LED diperbarui.


```

void handleTelegramMessages() {
  TMessage msg;
  if (myBot.getNewMessage(msg)) {
    String command = msg.text;
    if (command == "/on") {
      perangkatAktif = true;
      sendTelegramMessage("Perangkat telah diaktifkan.");
    } else if (command == "/off") {
      perangkatAktif = false;
      jumlahTerisiKursiCukurJaya = 0; // Reset jumlah terisi kursi
      jumlahTerisiKursiCukurThover = 0; // Reset jumlah terisi kursi
      sendTelegramMessage("Perangkat telah dinonaktifkan.");
      // Matikan LED saat perangkat tidak aktif
      digitalWrite(LED_PIN1, LOW);
      digitalWrite(LED_PIN2, LOW);
    } else if (command == "/totalPelanggan") {
      String replyMessage = "Total Pelanggan Kursi Cukur Jaya = " + String(jumlahTerisiKursiCukurJaya) + "\n";
      replyMessage += "Total Pelanggan Kursi Cukur Thover = " + String(jumlahTerisiKursiCukurThover) + "\n";
      int totalSemua = jumlahTerisiKursiCukurJaya + jumlahTerisiKursiCukurThover;
      replyMessage += "Total semua = " + String(totalSemua);
      sendTelegramMessage(replyMessage);
    }
  }
}

```

Gambar 4. 37 Fungsi *handleTelegramMessages* Jumlah Pelanggan

Pada gambar 4.37 fungsi *handleTelegramMessages()* mengelola pesan yang diterima dari *bot* Telegram. Jika menerima perintah */on*, perangkat diaktifkan dan notifikasi dikirim. Jika menerima perintah */off*, perangkat dinonaktifkan, jumlah pelanggan *reset*, dan status LED dimatikan. Perintah */totalPelanggan* digunakan untuk mengirim total pelanggan yang telah menggunakan kursi cukur.

4.3 Pengujian Kinerja dan Respon

Pada bagian ini, dilakukan pengujian untuk mengevaluasi kinerja dan respon dari sistem yang telah dikembangkan. Pengujian ini mencakup dua aspek utama: hasil pengukuran dan pengiriman data. Pengujian dilakukan secara menyeluruh untuk memastikan bahwa sistem berfungsi sesuai dengan yang diharapkan.

4.3.1 Pengujian Hasil Pengukuran

Pengujian hasil pengukuran dilakukan untuk memastikan bahwa sensor ultrasonik yang digunakan dapat mendeteksi keberadaan objek (dalam hal ini, kursi yang diduduki atau kosong) dengan akurat. Pengukuran dilakukan sebanyak 30 kali pada berbagai kondisi untuk mengevaluasi konsistensi dan keakuratan hasil yang diperoleh. Data hasil pengukuran dicatat dan dibandingkan dengan kondisi nyata untuk menentukan apakah hasil pengukuran sesuai atau tidak sesuai dengan kenyataan yang dapat dilihat pada tabel 4.3.

Tabel 4. 3 Pengujian Hasil Pengukuran

No.	Tanggal/Jam Pengukuran	Hasil Pengukuran (cm)	Status
1	2024-06-14 10:00	4.5	Sesuai
2	2024-06-14 10:01	3.8	Sesuai
3	2024-06-14 10:02	5.1	Tidak Sesuai
4	2024-06-14 10:03	4.0	Sesuai
5	2024-06-14 10:04	4.2	Sesuai
6	2024-06-14 10:05	2.9	Sesuai
7	2024-06-14 10:06	3.5	Sesuai
8	2024-06-14 10:07	4.8	Sesuai
9	2024-06-14 10:08	5.0	Sesuai
10	2024-06-14 10:09	2.9	Sesuai
11	2024-06-14 10:10	3.5	Sesuai
12	2024-06-14 10:11	4.8	Sesuai
13	2024-06-14 10:12	5.0	Sesuai
14	2024-06-14 10:13	5.2	Tidak Sesuai
15	2024-06-14 10:14	3.7	Sesuai
16	2024-06-14 10:15	4.9	Sesuai
17	2024-06-14 10:16	3.4	Sesuai
18	2024-06-14 10:17	4.7	Sesuai
19	2024-06-14 10:18	5.6	Tidak Sesuai
20	2024-06-14 10:19	3.9	Sesuai
21	2024-06-14 10:20	4.1	Sesuai
22	2024-06-14 10:21	5.0	Sesuai
23	2024-06-14 10:22	5.2	Tidak Sesuai
24	2024-06-14 10:23	3.5	Sesuai
25	2024-06-14 10:24	4.6	Sesuai
26	2024-06-14 10:25	4.2	Sesuai
27	2024-06-14 10:26	3.3	Sesuai
28	2024-06-14 10:27	4.4	Sesuai
29	2024-06-14 10:28	5.8	Tidak Sesuai
30	2024-06-14 10:29	3.8	Sesuai

4.3.2 Pengujian Pengiriman Data

Pengujian pengiriman data dilakukan untuk mengevaluasi kinerja sistem dalam mengirim data hasil pengukuran ke *server* atau aplikasi berbasis *web*. Pengujian ini bertujuan untuk memastikan bahwa data yang dikirim oleh sistem diterima dengan baik dan dalam waktu yang wajar. Pengujian dilakukan sebanyak 30 kali, dengan mencatat waktu pengiriman dan waktu penerimaan data. Hasil pengujian ini akan menunjukkan tingkat keberhasilan pengiriman data dan

kehandalan sistem dalam mentransmisikan informasi yang dapat dilihat pada tabel 4.4.

Tabel 4. 4 Pengujian Pengiriman Data

No.	Tanggal/Jam Pengukuran	Tanggal/Jam Diterima	Status
1	2024-06-14 10:00	2024-06-14 10:00:03	Berhasil
2	2024-06-14 10:01	2024-06-14 10:01:02	Berhasil
3	2024-06-14 10:02	2024-06-14 10:02:01	Berhasil
4	2024-06-14 10:03	2024-06-14 10:03:04	Berhasil
5	2024-06-14 10:04	2024-06-14 10:04:05	Berhasil
6	2024-06-14 10:05	2024-06-14 10:05:03	Berhasil
7	2024-06-14 10:06	2024-06-14 10:06:02	Berhasil
8	2024-06-14 10:07	2024-06-14 10:07:01	Berhasil
9	2024-06-14 10:08	2024-06-14 10:08:04	Berhasil
10	2024-06-14 10:09	2024-06-14 10:09:06	Berhasil
11	2024-06-14 10:10	2024-06-14 10:10:05	Berhasil
12	2024-06-14 10:11	2024-06-14 10:11:07	Berhasil
13	2024-06-14 10:12	2024-06-14 10:12:03	Berhasil
14	2024-06-14 10:13	2024-06-14 10:13:04	Berhasil
15	2024-06-14 10:14	2024-06-14 10:14:02	Berhasil
16	2024-06-14 10:15	2024-06-14 10:15:03	Berhasil
17	2024-06-14 10:16	2024-06-14 10:16:05	Berhasil
18	2024-06-14 10:17	2024-06-14 10:17:04	Berhasil
19	2024-06-14 10:18	2024-06-14 10:18:01	Berhasil
20	2024-06-14 10:19	2024-06-14 10:19:03	Berhasil
21	2024-06-14 10:20	2024-06-14 10:20:02	Berhasil
22	2024-06-14 10:21	2024-06-14 10:21:05	Berhasil
23	2024-06-14 10:22	2024-06-14 10:22:03	Berhasil
24	2024-06-14 10:23	2024-06-14 10:23:04	Berhasil
25	2024-06-14 10:24	2024-06-14 10:24:01	Berhasil
26	2024-06-14 10:25	2024-06-14 10:25:06	Berhasil
27	2024-06-14 10:26	2024-06-14 10:26:02	Berhasil
28	2024-06-14 10:27	2024-06-14 10:27:04	Berhasil
29	2024-06-14 10:28	2024-06-14 10:28:03	Berhasil
30	2024-06-14 10:29	2024-06-14 10:29:05	Berhasil

4.4 Pengujian Sistem

Pengujian sistem adalah tahapan penting dalam proses pengembangan yang memastikan bahwa sistem yang dibangun telah berfungsi sesuai dengan yang diharapkan. Pada sub bab ini, akan dilakukan pengujian menyeluruh terhadap

sistem yang telah dirancang, dengan fokus pada fungsionalitas dan keandalan sistem.

4.4.1 Pengujian Black Box Testing Sistem Booking Online

Sub bab ini mengevaluasi sistem *booking online* dari Mahkota *Barbershop* menggunakan metode pengujian *black box testing*. Pengujian ini bertujuan untuk menguji interaksi antara pengguna dan sistem secara keseluruhan, tanpa memperhatikan struktur internal dari sistem. Hasil pengujian ini akan memberikan gambaran tentang seberapa baik sistem dapat merespons input pengguna dan menjalankan fungsi-fungsi yang telah ditentukan.

Tabel 4. 5 Pengujian Sistem Booking Online

No.	Aspek yang Diuji	Interaksi Pengguna	Respon Sistem	Hasil Pengujian
1	Navigasi	Pengguna mengeksplorasi tautan dan tombol	<i>Website</i> menampilkan halaman yang sesuai	Berhasil
2	Formulir Pemesanan	Pengguna mengisi formulir pemesanan	Data pengguna <i>disubmit</i> dengan sukses ke sistem <i>backend</i>	Berhasil
3	Ketersediaan Informasi	Pengguna mencari informasi jadwal operasional, layanan, tukang cukur, dsb.	<i>Website</i> menampilkan informasi yang diperlukan	Berhasil
4	Responsivitas	Pengguna mengakses <i>website</i> dari berbagai perangkat (<i>desktop</i> ,	<i>Website</i> menyesuaikan tampilan dengan baik	Berhasil

No.	Aspek yang Diuji	Interaksi Pengguna	Respon Sistem	Hasil Pengujian
		<i>tablet, smartphone)</i>		
5	Integrasi dengan <i>Backend</i>	Pengguna melakukan pemesanan melalui <i>website</i>	Data pemesanan terintegrasi dengan sistem <i>backend</i>	Berhasil
6	Kompatibilitas <i>Browser</i>	Pengguna mengakses <i>website</i> dari berbagai <i>browser</i>	<i>Website</i> kompatibel dengan berbagai <i>browser</i>	Berhasil
7	Kompatibilitas Sistem Operasi	Pengguna mengakses <i>website</i> dari berbagai sistem operasi	<i>Website</i> berfungsi dengan baik di berbagai sistem operasi	Berhasil

4.4.2 Pengujian *Black Box Testing* Perangkat IoT

Pada sub bab ini, akan dilakukan pengujian terhadap perangkat IoT yang terintegrasi dalam sistem. Pengujian ini menggunakan metode *black box testing* untuk mengevaluasi fungsionalitas perangkat secara keseluruhan, tanpa memperhatikan detail implementasi internal. Pengujian ini akan mengevaluasi respons perangkat terhadap input eksternal dan kemampuannya dalam memenuhi tugas-tugas yang telah ditetapkan.

Tabel 4. 6 Pengujian Sistem Perangkat IoT

No.	Aspek yang Diuji	Interaksi Pengguna	Respon Sistem	Hasil Pengujian
1	Inisialisasi Perangkat	Admin mengirim perintah untuk mengaktifkan perangkat IoT	NodeMCU menginisialisasi dan terhubung ke <i>WiFi</i>	Berhasil
2	Sensor Ultrasonik	Pengguna mendekati kursi	Sensor mengirim data	Berhasil

No.	Aspek yang Diuji	Interaksi Pengguna	Respon Sistem	Hasil Pengujian
		tunggu atau kursi cukur	deteksi jarak ke NodeMCU	
3	Integrasi dengan Telegram	Admin mengirim perintah melalui Telegram	NodeMCU menerima dan merespon perintah dengan benar	Berhasil
4	Notifikasi	Sensor mendeteksi perubahan status (kursi diduduki/kosong)	NodeMCU mengirim notifikasi ke Telegram Bot	Berhasil
5	Penanganan Perintah	Admin mengirim perintah untuk mengubah status perangkat	NodeMCU menanggapi perintah dan mengubah status	Berhasil
6	Kinerja	Sensor mengirimkan data secara berkala	NodeMCU menerima dan memproses data dengan cepat	Berhasil
7	Responsivitas	Perubahan status sensor langsung diterjemahkan ke notifikasi Telegram	NodeMCU merespons dengan cepat	Berhasil

4.5 Evaluasi Hasil Pengujian

Pada tahap ini, hasil pengujian sistem *booking online* dan perangkat IoT akan dievaluasi untuk menentukan sejauh mana sistem telah memenuhi kebutuhan dan spesifikasi yang telah ditetapkan. Evaluasi ini mencakup pengukuran tingkat keberhasilan pengujian dan analisis terhadap hasil yang diperoleh, guna memastikan bahwa sistem dan perangkat dapat beroperasi secara optimal dalam mendukung operasional Mahkota *Barbershop*.

4.5.1 Evaluasi Hasil Pengujian Sistem Booking Online

Setelah dilakukan pengujian menggunakan metode *functional testing* dengan pendekatan *black box testing*, tingkat keberhasilan pengujian sistem *booking online* mencapai 100%. Dari total 7 kasus uji yang dilakukan, semuanya berhasil dilaksanakan sesuai dengan yang diharapkan. Setiap skenario uji mencakup berbagai fungsi kritis seperti registrasi pengguna, *login*, pemilihan layanan, pemilihan jadwal, konfirmasi *booking*, dan pemberian informasi *booking* kepada admin. Keberhasilan ini menunjukkan bahwa sistem mampu menangani setiap fungsi tersebut tanpa kesalahan, memastikan bahwa pengguna dapat melakukan *booking* secara efisien dan tanpa hambatan.

4.5.2 Evaluasi Hasil Pengujian Perangkat IoT

Pengujian perangkat IoT juga menghasilkan tingkat keberhasilan sebesar 100%. Dari total 7 kasus uji yang dilakukan pada perangkat IoT berhasil diselesaikan tanpa ada kegagalan. Kasus uji mencakup pengukuran jarak dengan sensor ultrasonik, pengiriman data ke *server*, penerimaan dan pemrosesan data oleh *server*, pengaktifan dan penonaktifan perangkat melalui perintah Telegram, serta penanganan kegagalan jaringan. Keberhasilan dalam setiap kasus uji ini menunjukkan bahwa perangkat IoT dapat berfungsi sesuai dengan desain dan spesifikasi teknis yang telah ditetapkan, memastikan keandalan dan konsistensi dalam operasional sehari-hari.

4.5.3 Analisis Evaluasi Hasil Pengujian Sistem Booking Online

Dengan tingkat keberhasilan pengujian sebesar 100%, dapat disimpulkan bahwa sistem *booking online* telah teruji dengan baik dan mampu menjalankan fungsinya sesuai dengan yang diharapkan. Sistem ini telah memenuhi semua persyaratan fungsional yang kritis, seperti kemudahan penggunaan, kecepatan respon, dan informasi *booking* berhasil kepada admin. Evaluasi juga menunjukkan bahwa sistem ini mampu menangani berbagai situasi dan skenario pengguna, dari registrasi hingga konfirmasi *booking*, dengan performa yang memuaskan. Keberhasilan ini membuktikan bahwa sistem siap untuk diimplementasikan secara

penuh dan dapat diandalkan oleh Mahkota *Barbershop* untuk meningkatkan efisiensi operasional dan pengalaman pelanggan.

4.5.4 Analisis Evaluasi Hasil Pengujian Perangkat IoT

Hasil pengujian perangkat IoT menunjukkan bahwa perangkat tersebut berhasil dalam menjalankan fungsi-fungsinya dengan sempurna. Dengan tingkat keberhasilan pengujian 100%, dapat disimpulkan bahwa perangkat IoT telah teruji dengan baik dan dapat digunakan secara efektif dalam operasional Mahkota *Barbershop*. Evaluasi ini mencakup aspek-aspek penting seperti akurasi pengukuran sensor, keandalan komunikasi data, dan respon terhadap perintah kendali jarak jauh. Keberhasilan dalam pengujian ini memastikan bahwa perangkat IoT dapat mendukung pengelolaan kursi cukur dan kursi tunggu secara efisien, memberikan notifikasi *real-time* kepada admin, serta memastikan ketersediaan layanan yang optimal. Implementasi perangkat IoT yang sukses ini akan memberikan nilai tambah yang signifikan dalam pengelolaan operasional dan layanan pelanggan di Mahkota *Barbershop*.

STT - NF

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan mengenai perancangan sistem *booking online* berbasis *web* pada Mahkota *Barbershop* dengan integrasi IoT, dapat ditarik kesimpulan sebagai berikut:

1. Aplikasi sistem *booking online* berbasis *web* untuk Mahkota *Barbershop* berhasil dirancang dan diimplementasikan untuk mengelola pesanan secara efisien. Aplikasi ini dirancang dengan *frontend* menggunakan HTML, CSS, dan JavaScript, serta *backend* menggunakan PHP dan MySQL. Pengujian *functional testing* dengan pendekatan *black box testing* menunjukkan tingkat keberhasilan 100% pada fungsi-fungsi kritis seperti registrasi, *login*, pemilihan layanan, pemilihan jadwal, konfirmasi *booking*, dan pemberian informasi *booking* kepada admin.
2. Sistem deteksi ketersediaan tempat duduk dan manajemen antrian di Mahkota *Barbershop* berhasil dirancang dan diimplementasikan menggunakan teknologi IoT. Sistem ini dirancang menggunakan teknologi IoT dengan komponen NodeMCU Lua ESP8266 dan sensor ultrasonik HC-SR04, serta perangkat lunak yang dikembangkan menggunakan Arduino IDE. Pengujian menunjukkan tingkat keberhasilan 100% dalam pengukuran jarak dengan sensor, pengiriman data ke *server*, dan respon terhadap perintah kendali jarak jauh.

5.2 Saran

Perancangan sistem *booking online* berbasis *web* pada Mahkota *Barbershop* dengan IoT menunjukkan beberapa kekurangan dalam implementasinya. Berikut adalah saran untuk pengembangan penelitian selanjutnya:

1. Peningkatan fitur untuk lebih meningkatkan pengalaman pengguna seperti fitur pembayaran *online*, notifikasi pengingat jadwal, dan integrasi dengan media sosial dapat dipertimbangkan. Selain itu, pengembangan aplikasi mobile untuk sistem *booking online* dapat meningkatkan aksesibilitas dan kenyamanan pelanggan. Fitur-fitur ini dapat menambah nilai tambah bagi pelanggan dan membuat proses *booking* lebih mudah dan efisien.
2. Pengembangan Sistem IoT lebih lanjut dapat dilakukan untuk meningkatkan akurasi dan efisiensi sistem deteksi ketersediaan tempat duduk. Sensor tambahan atau teknologi berbeda dapat diujicobakan untuk meningkatkan keandalan sistem. Selain itu, pengembangan fitur analitik untuk memantau dan menganalisis data penggunaan kursi dapat membantu dalam manajemen operasional *barbershop*. Ini akan memungkinkan *barbershop* untuk mengoptimalkan penggunaan ruang dan memberikan pelayanan yang lebih baik kepada pelanggan.

STT - NF

DAFTAR PUSTAKA

- [1] A. O. Akbar Nagara and A. Emaputra, "Analisis Kepuasan Konsumen terhadap Pelayanan Barbershop dengan Metode Service Quality (Servqual) dan Importance Performance Analysis (IPA) Studi Kasus XYZ Barbershop," *J. Rekayasa Ind. JRI*, vol. 2, no. 2, pp. 97–104, Oct. 2020, doi: 10.37631/jri.v2i2.183.
- [2] N. Salim, A. Fatkhudin, and E. Subowo, "Sistem Informasi Pemesanan Dan Transaksi Jasa Pangkas Rambut Pada Aka Barbershop Berbasis Web Dan Android," vol. 10, 2021.
- [3] D. Wulandari, T. Hidayati, and I. H. Ikasari, "Pelatihan Sistem Booking Online Berbasis Web untuk Para Karyawan dan Customer di Salon Muslimah Mumtaza Pamulang," *J. Pengabd. Kpd. Masy.*, vol. 2, no. 2, 2022.
- [4] F. F. D. Imaniawan and H. M. Nur, "Perancangan Dan Pembuatan Website Penjualan Biji Kopi Pada Society Coffee House Purwokerto," *Evolusi - J. Sains Dan Manaj.*, vol. 7, no. 1, Mar. 2019, doi: 10.31294/evolusi.v7i1.5030.
- [5] A. Firman, H. F. Wowor, and X. Naj Joan, "Sistem Informasi Perpustakaan Online Berbasis Web," 2016.
- [6] A. Hidayat and A. Yani, "Membangun Website Sma PGRI Gunung Raya Ranau Menggunakan Php Dan Mysql," vol. 2, no. 2, 2019.
- [7] M. F. Rizky and A. Alfarizy, "Rancang Bangun Website Promosi Dan Reservasi Pada Resto Danau Abah Menggunakan Metode Agile," 2023.
- [8] A. Buhori, D. A. Ramadhan, M. F. Alwan, and R. Andiyani, "Penguujian Aplikasi Reservasi Restaurant di Dream Restaurant Dengan Metode Black Box Menggunakan Teknik State Transitions Testing," *J. Teknol.*, vol. 1, no. 1, 2023.
- [9] A. Siswanto, A. Efendi, and A. Yulianti, "Alat Kontrol Akses Pintu Rumah Dengan Teknologi Sidik Jari Di Lingkungan Rumah Pintar Dengan Data Yang Di Enkripsi," *J. Penelit. Pos Dan Inform.*, vol. 8, no. 2, p. 97, Dec. 2018, doi: 10.17933/jppi.2018.080201.
- [10] G. Hergika, Siswanto, and S. S, "Perancangan Internet Of Things (Iot) Sebagai Kontrol Infrastruktur Dan Peralatan Toll Pada Pt. Astra Infratoll Road,"

PROSISKO J. Pengemb. Ris. Dan Obs. Sist. Komput., vol. 8, no. 2, pp. 86–98, Sep. 2021, doi: 10.30656/prosisko.v8i2.3862.

- [11] A. H. Nadyawan, “Aplikasi Sensor Ultrasonic Hc-Sr04 Pada Robot Anti Penghalang”.
- [12] A. Cahyo Putranto and M. Yaser, “Sistem Pengukuran Dan Pemutusan Penggunaan Daya Listrik Secara Real Time Berbasis Internet Of Things,” *Tesla J. Tek. Elektro*, vol. 24, no. 1, p. 70, May 2022, doi: 10.24912/tesla.v24i1.12106.
- [13] A. K. Lubis, A. Yanie, and D. Sawitri, “Desain Dan Perancangan Alat Pantau Energi Listrik Di Rumah Jarak Jauh Berbasis IoT,” vol. 01, no. 01, 2022.
- [14] Tri Sulistyorini, Nelly Sofi, and Erma Sova, “Pemanfaatan Nodemcu Esp8266 Berbasis Android (Blynk) Sebagai Alat Alat Mematikan Dan Menghidupkan Lampu,” *J. Ilm. Tek.*, vol. 1, no. 3, pp. 40–53, Sep. 2022, doi: 10.56127/juit.v1i3.334.
- [15] R. Rifandi, S. S, and Anharudin, “Rancang Bangun Kamera Pengawas Menggunakan Raspberry Dengan Aplikasi Telegram Berbasis Internet Of Things,” *Prosisko J. Pengemb. Ris. Dan Obs. Sist. Komput.*, vol. 8, no. 1, pp. 18–32, Mar. 2021, doi: 10.30656/prosisko.v8i1.3101.
- [16] S. D. Parameswari, N. A. Prasetyo, and A. Junaidi, “Aplikasi Berbasis Web Deteksi Undertone Menggunakan Metode Agile Untuk Rekomendasi Makeup,” *J. Ilm. Media Sisfo*, vol. 16, no. 1, pp. 62–70, Apr. 2022, doi: 10.33998/mediasisfo.2022.16.1.1111.

STT - NF