



ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR - IF184802

PERANCANGAN DAN IMPLEMENTASI SISTEM HOTEL CERDAS

FARHAN ZUHDI
0511164000070

Dosen Pembimbing I
Ir. Muchammad Husni, M.Kom.

Dosen Pembimbing II
Dr.Eng. Radityo Anggoro , S.Kom, M.Sc.

DEPARTEMEN INFORMATIKA
Fakultas Teknologi Elektro dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember
Surabaya
2020

[Halaman ini sengaja dikosongkan]



TUGAS AKHIR - IF184802

PERANCANGAN DAN IMPLEMENTASI SISTEM HOTEL CERDAS

FARHAN ZUHDI
0511164000070

Dosen Pembimbing I
Ir. Muchammad Husni, M.Kom.

Dosen Pembimbing II
Dr.Eng. Radityo Anggoro , S.Kom, M.Sc.

DEPARTEMEN INFORMATIKA
Fakultas Teknologi Elektro dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember
Surabaya
2020

[Halaman ini sengaja dikosongkan]



FINAL PROJECT - IF184802

DESIGN AND IMPLEMENTATION OF SMART HOTEL

FARHAN ZUHDI
0511164000070

Supervisor I
Ir. Muchammad Husni, M.Kom.

Supervisor II
Dr.Eng. Radityo Anggoro , S.Kom, M.Sc.

DEPARTMENT OF INFORMATICS
Faculty of Intelligent Electrical and Informatics Technology
Institut Teknologi Sepuluh Nopember
Surabaya
2020

[Halaman ini sengaja dikosongkan]

LEMBAR PENGESAHAN
PERANCANGAN DAN IMPLEMENTASI SISTEM HOTEL
CERDAS

TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Bidang Studi Algoritma dan Pemrograman
Program Studi S-1 Teknik Informatika

Departemen Teknik Informatika
Fakultas Teknologi Elektro dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember

Oleh:

Farhan Zuhdi

NRP: 05111640000070

Disetujui oleh Dosen Pembimbing Tugas Akhir:

Ir. Muchammad Husni, M.Kom.
(NIP. 196002211984031001)

Dr.Eng. Radityo Anggoro , S.Kom, M.Sc.
(NIP. 198410162008121002)



(Pembimbing 1)

(Pembimbing 2)

SURABAYA

JANUARI 2020

[Halaman ini sengaja dikosongkan]

PERANCANGAN DAN IMPLEMENTASI SISTEM HOTEL CERDAS

Nama Mahasiswa : Farhan Zuhdi
NRP : 051116 40000 070
Departemen : Departemen Teknik Informatika
Fakultas Teknologi Elektro dan
Informatika Cerdas – ITS
Dosen Pembimbing 1 : Ir. Muchammad Husni, M.Kom.
Dosen Pembimbing 2 : Dr.Eng. Radityo Anggoro , S.Kom,
M.Sc.

Abstrak

Industri perhotelan adalah usaha bisnis untuk pemilik hotel dan hiburan bagi wisatawan. Seorang pelanggan dapat terdampar dengan sistem perhotelan umum yang banyak digunakan saat ini dalam upaya untuk mendapatkan kamar hotel untuk melewati malam jika ia belum membuat rencana yang memadai. Disadari bahwa bagi seorang pelanggan agar terjamin mendapatkan kamar, sering sekali ia harus secara fisik datang ke hotel karena para pelayan lebih memperhatikan hal itu. Ia juga dapat mengirim teman atau kerabat yang tinggal di sekitar lingkungan hotel pada daerah itu untuk melakukan pemesanan untuknya. Namun, tidak ada yang mengikat hotel dan pelanggan secara langsung bahwa ia memang pelanggan yang memesan. Dan meskipun berhasil mendapatkan kamar, pelanggan tersebut juga harus melakukan check-in dengan proses yang cukup panjang, apalagi jika antrean sedang panjang. Pelanggan hotel juga akan kesulitan apabila harus membawa kunci maupun kartu, dan akses kamar hanya dapat dibawa oleh satu orang. Metode penyelesaian yang digunakan adalah pemanfaatan teknologi IoT.

Teknologi IoT diimplementasikan dalam bentuk prototipe atau maket, dengan sebuah aplikasi berbasis android untuk pemesanan hotel dan room controlling.

Aplikasi pemesanan dan room controlling hotel diimplementasikan dengan menggunakan Android Studio dan database MySQL. Sedangkan perangkat-perangkat yang ada pada kamar akan diimplementasikan dengan menggunakan Bahasa pemrograman C yang di compile melalui Arduino IDE dan database MySQL. Dari uji coba yang telah dilakukan, didapatkan kesimpulan bahwa alat tersebut telah berjalan dan mampu berinteraksi dengan WiFi, dapat dipantau dan dikontrol melalui website oleh admin, serta peningkatan kemudahan dalam hal pemesanan hotel, antrian check-in yang berkurang karena hanya perlu melakukan scan QR Code, dan akses kamar dapat dibagikan kepada lebih dari satu pengunjung yang menginap dalam kamar tersebut tanpa menggunakan kunci maupun kartu menggunakan teknologi IoT telah sesuai untuk memecahkan persoalan yang ada.

Kata kunci: Teknologi IoT, Android, Android Studio, MySQL, Codeigniter, Bahasa Pemrograman C, Arduino IDE

DESIGN AND IMPLEMENTATION OF SMART HOTEL

Student Name : Farhan Zuhdi
Registration Number : 051116 40000 070
Department : Department of Informatics
Faculty of Intelligent Electrical and
Informatics Technology – ITS
First Supervisor : Ir. Muchammad Husni, M.Kom.
Second Supervisor : Dr.Eng. Radityo Anggoro , S.Kom,
M.Sc.

Abstract

The hotel industry is a business venture for owners and entertainment for tourists. A customer can be stranded with the general hotel system used today in search of a hotel room to spend the night if he has not made an adequate plan. It was realized that for customers to be able to get a room, he often had to come to the hotel because the waiters paid more attention to it. He can also send a friend or relative who lives near a hotel at that area to book a room for him. However, nothing binds the hotel and customers who book the room. And even if he managed to get a room, he also had to check-in with a fairly long process, especially if the line is long. Hotel customers will also have difficulty if they must be carrying cards, and if room can only be opened by one person. The method used to solve this problem is the IoT technology application.

IoT technology is implemented in the form of prototypes or models, with an Android-based application for hotel reservation and room controlling.

The hotel reservation and room controlling application is implemented using Android Studio and MySQL database. While the devices in the room will be implemented using C

programming language compiled via Arduino IDE and MySQL database. From the trials that have been carried out, it can be concluded that the device is already running and able to interact with WiFi, can be monitored and controlled through the website by the admins, as well as increased convenience in terms of hotel reservation, reduced check-in queues because they only need to scan the QR Code, and room access can be distributed to more than one visitor who stays in the room without using a key or card using IoT technology which is appropriate to solve the problems.

KATA PENGANTAR

Puji syukur penulis ucapkan kepada Tuhan Yang Maha Esa atas segala rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul:

PERANCANGAN DAN IMPLEMENTASI SISTEM HOTEL CERDAS

Pengerjaan Tugas Akhir ini dilakukan untuk memenuhi salah satu syarat meraih gelar Sarjana di Departemen Informatika Fakultas Teknologi Informasi dan Komunikasi Institut Teknologi Sepuluh Nopember.

Dengan selesainya Tugas Akhir ini diharapkan apa yang telah dikerjakan penulis dapat memberikan manfaat bagi perkembangan ilmu pengetahuan terutama di bidang teknologi informasi serta bagi orang lain dan diri penulis sendiri.

Penulis mengucapkan terima kasih kepada semua pihak yang telah memberikan dukungan baik secara langsung maupun tidak langsung selama penulis mengerjakan Tugas Akhir maupun selama menempuh masa studi antara lain:

1. Terima kasih kepada Allah SWT, di mana penulis masih diberi kesempatan, kesehatan dan umur untuk menempuh kuliah disini dan menjalani hidup dengan baik sehingga mampu menyelesaikan Tugas Akhir ini.
2. Bapak Yeviar Idris, Ibu Risnawati, Thufail Akhdan Zaki dan Dhiya Zahira Salsabila selaku keluarga penulis yang selalu memberikan perhatian, do'a, dorongan, dan juga kasih sayang agar lebih semangat menempuh kuliah dan segera menyelesaikan Tugas Akhir ini.
3. Bapak Ir. Muchammad Husni, M.Kom. selaku Dosen Pembimbing yang telah membimbing penulis selama masa kuliah maupun selama penyelesaian Tugas Akhir ini.
4. Bapak Dr. Radityo Anggoro , S.Kom, M.Sc. Selaku Dosen Pembimbing yang telah memberikan ilmu, dan masukan kepada penulis.

5. Nadya Eka Fauziyah, yang selalu memberikan doa maupun usaha dalam membantu penulis serta menyemangati penulis untuk menyelesaikan Tugas Akhir ini.
6. Teman-teman kontrakan, yang selalu menemani penulis dalam penyusunan buku Tugas Akhir ini.
7. Teman-teman SMA penulis yang pernah tinggal bersama penulis di kost Keputih gang 3 no.50, yang selalu menemani penulis selama kuliah dan selalu memberikan motivasi kepada penulis dalam menyelesaikan buku Tugas Akhir ini.
8. Teman-teman Jamaah IoT Ceria, karena selalu saling membantu dan memberi semangat kepada penulis selama mengerjakan Tugas Akhir.
9. Teman-teman angkatan 2016 jurusan Teknik Informatika ITS yang telah menemani perjuangan penulis selama masa perkuliahan.
10. Serta pihak-pihak lain yang tidak dapat disebutkan di sini yang telah banyak membantu penulis dalam penyusunan Tugas Akhir ini.

Penulis mohon maaf apabila masih ada kekurangan pada Tugas Akhir ini. Penulis juga mengharapkan kritik dan saran yang membangun untuk pembelajaran dan perbaikan di kemudian hari. Semoga melalui Tugas Akhir ini penulis dapat memberikan kontribusi dan manfaat yang sebaik-baiknya.

Surabaya, Juni 2020

Farhan Zuhdi

DAFTAR ISI

LEMBAR PENGESAHAN.....	Error! Bookmark not defined.
<i>Abstrak</i>	vii
<i>Abstract</i>	x
KATA PENGANTAR.....	xii
DAFTAR ISI.....	xiv
DAFTAR GAMBAR.....	xviii
DAFTAR TABEL.....	xxi
DAFTAR KODE SUMBER.....	xxiii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Permasalahan.....	2
1.3 Batasan Permasalahan.....	3
1.4 Tujuan Pembuatan Tugas Akhir.....	3
1.5 Manfaat Tugas Akhir.....	3
1.6 Metodologi.....	3
1.6.1 Penyusunan Proposal Tugas Akhir.....	3
1.6.2 Studi Literatur.....	4
1.6.3 Implementasi Perangkat Lunak.....	4
1.6.4 Pengujian dan Evaluasi.....	4
1.6.5 Penyusunan Buku Tugas Akhir.....	5
1.7 Sistematika Penulisan.....	5
2 BAB II TINJAUAN PUSTAKA.....	7
2.1 <i>Hardware Tools</i>	7
2.1.1 WeMos D1 R1.....	7
2.1.2 <i>Solenoid Door Lock</i>	7
2.1.3 <i>Relay 4 Channel 5V</i>	8
2.1.4 <i>Green LED</i>	8
2.1.5 <i>Jumper Cable</i>	9
2.2 <i>Software Tools</i>	9
2.2.1 Arduino IDE.....	9
2.2.2 XAMPP.....	10
2.2.3 <i>Web Server</i>	11

2.2.4 <i>HTTP</i>	11
2.2.5 CodeIgniter 3.....	12
2.2.6 Sublime Text 3.....	12
2.2.7 SQLyog.....	12
2.2.8 Android Studio.....	13
3 BAB III PERANCANGAN.....	15
3.1 Arsitektur Sistem.....	15
3.2 Perancangan <i>Database</i>	22
3.3 Perancangan Alat.....	28
3.4 Perancangan Sistem <i>Administrator</i>	30
4 BAB IV IMPLEMENTASI.....	34
4.1 Implementasi <i>Database</i>	34
4.1.1 Implementasi Tabel.....	34
4.1.2 Implementasi <i>Trigger</i> dan <i>Event</i>	39
4.1.3 Hasil <i>Query MySQL</i>	41
4.2 Implementasi Alat.....	42
4.2.1 Rangkaian Alat.....	42
4.2.2 Implementasi Arduino.....	46
4.3 Implementasi Sistem Berbasis Android.....	52
4.3.1 Implementasi Fitur Pemesanan Kamar Hotel.....	53
4.3.2 Implementasi Pembayaran Online Menggunakan Midtrans.....	54
4.3.3 Implementasi <i>Monitoring</i> dan <i>Controlling</i> Kamar Hotel.....	55
4.4 Codeigniter.....	59
4.4.1 Implementasi <i>Web Service</i>	60
4.4.2 Implementasi Website Sistem <i>Monitoring</i> dan <i>Controlling</i> Hotel.....	67
5 BAB V UJICOBA DAN EVALUASI.....	75
5.1 Lingkup Uji Coba.....	75
5.2 Skenario Pengujian.....	76
5.2.1 Aplikasi Hotel Cerdas.....	76
5.2.2 Sistem <i>Administrator Monitoring</i> dan <i>Controlling</i> Hotel.....	82
5.2.3 <i>Controlling</i> Alat Pada Maket atau Miniatur.....	87
6 BAB VI KESIMPULAN.....	91

6.1 Kesimpulan.....	91
6.2 Saran.....	92
DAFTAR PUSTAKA.....	94
LAMPIRAN.....	97
BIODATA PENULIS.....	99

[Halaman ini sengaja dikosongkan]

DAFTAR GAMBAR

Gambar 2.1	Wemos D1 R1.....	7
Gambar 2.2	<i>Solenoid door lock</i>	8
Gambar 2.3	Relay 4 Channel 5V.....	8
Gambar 2.4	<i>Green LED</i>	9
Gambar 2.5	<i>Jumper Cable</i>	9
Gambar 3.1	Alur Interaksi Device.....	15
Gambar 3.2	<i>Flowchart</i> Aplikasi Hotel Cerdas.....	17
Gambar 3.3	<i>Flowchart</i> Alur Reservasi Hotel.....	18
Gambar 3.4	<i>Flowchart</i> Alur Check-in Kamar Hotel.....	19
Gambar 3.5	<i>Flowchart</i> Alur Mengontrol Perangkat.....	20
Gambar 3.6	<i>Flowchart</i> Aur Check Out.....	21
Gambar 3.7	Interaksi <i>Microcontroller</i> dengan Alat.....	29
Gambar 3.8	<i>Use Case Diagram</i> Sistem <i>Administrator</i>	31
Gambar 4.1	Hasil Eksekusi MySQL.....	42
Gambar 4.2	Rangkaian Wemos dengan Relay.....	43
Gambar 4.3	Rangkaian Relay dengan Solenoid.....	43
Gambar 4.4	Rangkaian Relay dengan Lampu.....	44
Gambar 4.5	Rangkaian Relay dengan Kipas.....	44
Gambar 4.6	Rangkaian Wemos dengan <i>Green LED</i>	45
Gambar 4.7	Rangkaian Sistem Hotel Cerdas.....	45
Gambar 4.8	<i>Pseudocode</i> Fungsi Mengirim <i>HTTP Request</i>	49
Gambar 4.9	<i>Pseudocode</i> Membuka dan Menutup Pintu.....	50
Gambar 4.10	<i>Pseudocode</i> Menghidupkan dan Mematikan Lampu.....	50
Gambar 4.11	<i>Pseudocode</i> Menghidupkan dan Mematikan Kipas.....	51
Gambar 4.12	<i>Pseudocode</i> Menghidupkan dan Mematikan Lift.....	52
Gambar 4.13	<i>Pseudocode</i> Kelas Connector.....	53
Gambar 4.14	<i>Pseudocode</i> Melakukan Pemesanan.....	54
Gambar 4.15	<i>Pseudocode</i> Melakukan Pembayaran <i>Online</i>	55
Gambar 4.16	<i>Pseudocode</i> <i>Check-in</i> dengan <i>QR Code</i>	56
Gambar 4.17	<i>Pseudocode</i> Mengontrol Kamar.....	56
Gambar 4.18	<i>Pseudocode</i> Implementasi <i>Room Sharing</i>	57
Gambar 4.19	<i>Pseudocode</i> Implementasi <i>Check-out</i>	58
Gambar 4.20	<i>Pseudocode</i> Implementasi Akses Lift.....	59
Gambar 4.21	Konfigurasi <i>Database</i>	59

Gambar 4.22	<i>Pseudocode</i> Mendapatkan Daftar Hotel.....	61
Gambar 4.23	<i>Pseudocode</i> Mendapatkan Kamar yang Tersedia.....	62
Gambar 4.24	<i>Pseudocode</i> Menambahkan Transaksi.....	62
Gambar 4.25	<i>Pseudocode</i> Fungsi Mendapatkan Daftar Kamar.....	63
Gambar 4.26	<i>Pseudocode</i> Mengubah Status Check-in.....	64
Gambar 4.27	<i>Pseudocode</i> Mendapatkan Kondisi Kamar.....	64
Gambar 4.28	<i>Pseudocode</i> Fungsi <i>Controlling</i> Perangkat.....	65
Gambar 4.29	<i>Pseudocode</i> Tambah atau Hapus <i>Shared User</i>	65
Gambar 4.30	<i>Pseudocode</i> Fungsi <i>Checkout</i>	66
Gambar 4.31	<i>Pseudocode</i> Fungsi Mendapatkan <i>Shared Room</i>	66
Gambar 4.32	<i>Pseudocode</i> Mendapatkan Status Perangkat.....	67
Gambar 4.33	<i>Pseudocode</i> Fungsi Menampilkan Data Kamar.....	69
Gambar 4.34	<i>Pseudocode</i> Mengontrol Kamar.....	70
Gambar 4.35	<i>Pseudocode</i> Mengontrol <i>Lift</i>	70
Gambar 4.36	<i>Pseudocode</i> Fungsi Konfirmasi Check-out.....	71
Gambar 4.37	<i>Pseudocode</i> Fungsi Menampilkan Riwayat Transaksi.....	72
Gambar 4.38	<i>Pseudocode</i> Fungsi Menampilkan QR Code.....	73
Gambar 5.1	Halaman Utama dan Navigasi.....	77
Gambar 5.2	Skenario Pengujian Pemesanan Hotel.....	78
Gambar 5.3	Skenario Pengujian Check-in.....	79
Gambar 5.4	Skenario Pengujian Mengontrol Perangkat Kamar.....	80
Gambar 5.5	Halaman <i>Room Control</i> untuk <i>Shared User</i>	81
Gambar 5.6	Skenario Pengujian <i>Check-out</i>	82
Gambar 5.7	Halaman Dashboard.....	83
Gambar 5.8	Halaman Data Kamar.....	83
Gambar 5.9	Halaman <i>Room Control Website</i>	84
Gambar 5.10	Halaman <i>Lift Control</i>	84
Gambar 5.11	Check-in Berhasil.....	85
Gambar 5.12	Check-out Berhasil.....	85
Gambar 5.13	Konfirmasi Check-out.....	86
Gambar 5.14	Riwayat Transaksi.....	86
Gambar 5.15	Halaman QR Code Check-in.....	87
Gambar 5.16	Mengontrol Pintu.....	87
Gambar 5.17	Mengontrol Lampu.....	88

Gambar 5.18 Mengontrol Kipas.....	88
Gambar 5.19 Mengontrol <i>Lift</i>	88
Gambar 5.20 Perangkat Tidak Aktif.....	89
Gambar 5.21 Perangkat Aktif.....	89

DAFTAR TABEL

Tabel 2.1	<i>Library</i> Arduino.....	10
Tabel 2.2	<i>Library</i> Android Studio.....	13
Tabel 3.1	Deskripsi Tabel <i>Database</i>	22
Tabel 3.2	Detail Tabel ht_user.....	22
Tabel 3.3	Detail Tabel ht_hotel.....	23
Tabel 3.4	Detail Tabel ht_kamar.....	24
Tabel 3.5	Detail Tabel ht_transaksi.....	24
Tabel 3.6	Detail Tabel ht_share.....	25
Tabel 3.7	Detail Tabel ht_lift.....	26
Tabel 3.8	Detail Tabel ht_admin.....	26
Tabel 3.9	Detail Tabel ht_log_transaksi.....	27
Tabel 4.1	Rangkaian Sistem Hotel Cerdas.....	46
Tabel 5.1	Lingkup Spesifikasi <i>Smartphone</i> Android.....	75
Tabel 5.2	Lingkup Spesifikasi Laptop.....	75
Tabel 5.3	Uji Respons Alat.....	90

[Halaman ini sengaja dikosongkan]

DAFTAR KODE SUMBER

Kode Sumber 4.1	MySQL ht_user.....	34
Kode Sumber 4.2	MySQL ht_hotel.....	35
Kode Sumber 4.3	MySQL ht_kamar.....	35
Kode Sumber 4.4	MySQL ht_transaksi.....	36
Kode Sumber 4.5	MySQL ht_share.....	37
Kode Sumber 4.6	MySQL ht_lift.....	37
Kode Sumber 4.7	MySQL ht_admin.....	38
Kode Sumber 4.8	MySQL ht_log_transaksi.....	39
Kode Sumber 4.9	MySQL Trigger After Insert.....	40
Kode Sumber 4.10	MySQL Trigger After Update.....	40
Kode Sumber 4.11	MySQL Trigger Before Delete.....	41
Kode Sumber 4.12	MySQL Event Checkout Handler.....	41
Kode Sumber 4.13	Library dan Konfigurasi WiFi.....	47
Kode Sumber 4.14	Deklarasi Pin.....	47
Kode Sumber 4.15	Setup Arduino.....	48
Kode Sumber 4.16	Fungsi <i>Connect WiFi</i>	48
Kode Sumber 4.17	Fungsi <i>Loop</i>	49

[Halaman ini sengaja dikosongkan]

BAB I

PENDAHULUAN

Pada bab ini akan dipaparkan mengenai garis besar Tugas Akhir yang meliputi latar belakang, tujuan, rumusan masalah, batasan permasalahan, metodologi pembuatan Tugas Akhir, dan sistematika penulisan buku Tugas Akhir ini.

1.1 Latar Belakang

Hotel merupakan salah satu bentuk pelayanan publik yang menawarkan suatu jasa dalam hal penyediaan tempat tinggal, yang sifatnya sementara dan dalam waktu-waktu tertentu bagi siapa saja yang memerlukannya. Hotel fungsi menyediakan layanan penginapan kepada pelanggan. Pelanggan yang dimaksud bisa musafir, orang asing, pengusaha, wisatawan, pengunjung, dll. Sebagian besar pelanggan akan kesulitan untuk mendapatkan kamar untuk melewati malam, karena biasanya mereka mencari hotel ketika telah tiba di lokasi tertentu, masuk dari hotel ke hotel dan mencari tahu apakah ada kamar kosong. Jika tidak ada kamar kosong, mereka harus pindah ke hotel terdekat berikutnya untuk bertanya sekali lagi. Mungkin ada yang beruntung memiliki nomor kontak hotel yang dapat dihubungi untuk memesan kamar. Tetapi, petugas hotel belum tentu dapat menyediakan kamar. Pada kasus lain, Mungkin pelanggan tersebut memiliki kerabat di daerah sana yang dapat memesan hotel. Namun, Tidak ada sistem di tempat yang mengikat hotel dan pelanggan bahwa pelanggan yang sebenarnya bukan kerabat dari pelanggan tersebut. Sehingga pelanggan tidak dapat dipastikan mendapatkan kamar. Hal ini dapat membuat pelanggan terlantar pada malam hari.[2]

Ketika pelanggan mendapatkan hotel, pelanggan juga harus melewati proses *check-in* dengan durasi yang cukup lama apabila antrian cukup panjang. Pelanggan tersebut juga belum tentu bisa mendapatkan kamar pada lantai yang diinginkan.

Sehingga dapat membuat kesal pelanggan yang sudah lelah melakukan perjalanan jauh.

Untuk itu, perlu adanya sistem yang dapat memudahkan pemesanan hotel dengan membuat sebuah aplikasi pemesanan hotel berbasis Android yang digunakan untuk memesan hotel, namun dapat memilih kamar secara spesifik, serta dapat melakukan pembayaran secara online. Dan memadukan konsep *Internet of Things (IoT)*, dimana pengunjung hotel dapat langsung melakukan *check in* dengan *scan QR-Code* yang terdapat di lobby dan Smart Lock yang akan digunakan pada pintu kamar[3]. Setelah melakukan *check in*, pengunjung dapat langsung ke kamar dan menekan tombol yang tersedia pada *room controlling*[8] yang terdapat pada aplikasi untuk membuka kunci pintu.

1.2 Rumusan Permasalahan

Rumusan masalah yang terdapat pada tugas akhir ini adalah sebagai berikut:

1. Bagaimana membuat aplikasi reservasi kamar hotel berbasis Android, dan melakukan pembayaran secara online?
2. Bagaimana konfirmasi kamar pada resepsionis dengan *scan QR-Code*?
3. Bagaimana cara membuka kunci kamar dengan menekan tombol yang ada pada *room controlling* dalam aplikasi?
4. Bagaimana mengaktifkan dan mematikan perangkat yang ada di kamar hotel?
5. Bagaimana merancang *prototype* Sistem Hotel Cerdas?
6. Bagaimana membuat *website monitoring* dan *controlling* hotel untuk digunakan oleh admin?

1.3 Batasan Permasalahan

Permasalahan pada Tugas Akhir ini, memiliki batasan masalah sebagai berikut:

1. Aplikasi hanya dapat digunakan menggunakan android.
2. Sistem yang dibangun berupa *prototype*.
3. Server menggunakan *localhost*.
4. Sistem Cerdas berupa mengontrol perangkat-perangkat pada hotel melalui *mobile apps*.

1.4 Tujuan Pembuatan Tugas Akhir

Tujuan dari pembuatan Tugas Akhir ini adalah untuk merancang Sistem Hotel Cerdas.

1.5 Manfaat Tugas Akhir

Manfaat dari Tugas Akhir ini diharapkan dapat membantu Sistem Perhotelan mengurangi lama proses check in, dan tidak membebani pengunjung dengan kunci pintu maupun sebuah kartu untuk membuka pintu kamar.

1.6 Metodologi

Langkah-langkah yang ditempuh dalam pengerjaan Tugas Akhir ini yaitu:

1.6.1 Penyusunan Proposal Tugas Akhir

Proposal Tugas Akhir ini merupakan pendahuluan dalam menyelesaikan Tugas Akhir. Proposal ini selain berisi ringkasan tugas akhir juga meliputi latar belakang dari masalah, rumusan masalah yang akan diselesaikan, batasan masalah, tujuan dan manfaat dalam tugas akhir yang dibuat, tinjauan pustaka dalam mengerjakan tugas akhir serta metodologi yang berisi tahapan-tahapan dalam menyusun tugas akhir.

1.6.2 Studi Literatur

Studi literatur yang dipakai adalah dengan pengumpulan informasi mengenai bahan-bahan yang digunakan dalam pembuatan prototipe yang dijadikan sebagai referensi dalam pengerjaan Tugas Akhir. Mengumpulkan informasi mengenai IoT (*Internet of Things*), solenoid, *wemos* dan *arduino*. Informasi didapatkan dari buku, paper, jurnal dan materi-materi kuliah yang berhubungan dengan topik Tugas Akhir.

1.6.3 Implementasi Perangkat Lunak

Implementasi yang akan dilakukan yaitu prototipe perancangan sistem berdasarkan studi literatur dan pengumpulan informasi yang telah dilakukan. Tahap ini suatu bentuk awal perancangan yang akan diimplementasikan. Yang dibuat adalah merupakan prototipe versi awal perancangan sistem hotel cerdas.

1.6.4 Pengujian dan Evaluasi

Pengujian dilakukan melalui dua cara yaitu:

1. Pengujian *hardware*
Pengujian *hardware* adalah pengujian yang mengutamakan spesifikasi fungsional dari perangkat keras. Pengujian ini dilakukan untuk menguji apakah sistem *room controlling* untuk mengendalikan perangkat-perangkat yang terdapat pada kamar hotel berjalan atau tidak.
2. Pengujian *software*
Pengujian *software* adalah pengujian yang berfokus pada spesifikasi fungsional dari perangkat lunak. Pengujian ini dilakukan untuk menguji apakah perangkat lunak sistem hotel cerdas berjalan atau tidak.

1.6.5 Penyusunan Buku Tugas Akhir

Pada tahap ini dilakukan penyusunan laporan yang menjelaskan dasar teori dan metode yang digunakan dalam tugas akhir ini serta hasil dari implementasi aplikasi perangkat lunak yang telah dibuat. Sistematika penulisan buku tugas akhir secara garis besar antara lain.

1.7 Sistematika Penulisan

Buku Tugas Akhir ini merupakan laporan secara lengkap mengenai Tugas Akhir yang telah dikerjakan baik dari sisi teori, rancangan, maupun implementasi sehingga memudahkan bagi pembaca dan juga pihak yang ingin mengembangkannya lebih lanjut. Sistematika penulisan buku Tugas Akhir secara garis besar antara lain:

Bab I Pendahuluan

Bab ini berisi penjelasan latar belakang, rumusan masalah, batasan masalah dan tujuan pembuatan Tugas Akhir. Selain itu, metodologi pengerjaan dan sistematika penulisan laporan Tugas Akhir juga dijelaskan di dalamnya.

Bab II Tinjauan Pustaka

Pada bagian ini berisi kajian teori atau penjelasan dari metode, algoritma, *library*, dan *tools* yang digunakan dalam penyusunan Tugas Akhir ini.

Bab III Perancangan Sistem

Bab ini berisi penjelasan tentang rancangan dari sistem yang akan dibangun.

Bab IV Implementasi

Pada bagian ini menjelaskan implementasi yang berbentuk kode program dari proses modifikasi, proses pengujian, serta hardware dan maket.

Bab V Pengujian dan Evaluasi

Bab ini berisi penjelasan mengenai data hasil percobaan dan pembahasan mengenai hasil percobaan yang telah dilakukan.

Bab VI Kesimpulan dan Saran

Bab ini merupakan bab terakhir yang menjelaskan kesimpulan dari hasil uji coba yang dilakukan dan saran untuk pengembangan perangkat lunak ke depannya.

BAB II

TINJAUAN PUSTAKA

Bab ini berisi penjelasan mengenai teori-teori dasar yang berkaitan dengan pengimplementasian perangkat lunak dan perangkat keras serta penunjangnya. Penjelasan ini bertujuan untuk memberikan gambaran umum terhadap *Internet of Things, tools*, serta definisi yang digunakan dalam pembuatan Tugas Akhir.

2.1 *Hardware Tools*

2.1.1 WeMos D1 R1

WeMos D1 merupakan module development board yang berbasis *WiFi* dari ESP8266 yang dimana dapat diprogram menggunakan *software* IDE Arduino, perangkat inilah yang nantinya akan menghubungkan perangkat dengan *internet* melalui *WiFi*. [13]



Gambar 2.1 Wemos D1 R1 [15]

2.1.2 *Solenoid Door Lock*

Solenoid ini berfungsi sebagai *aktuator*. Prinsip dari solenoid sendiri akan bekerja sebagai pengunci dan akan aktif ketika diberikan tegangan. Didalam solenoid terdapat kawat yang melingkar pada inti besi. Ketika arus listrik mengalir melalui

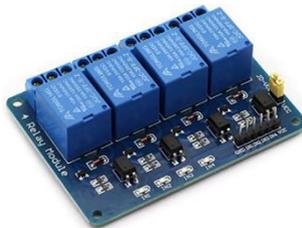
kawat ini, maka terjadi medan magnet untuk menghasilkan energi yang akan menarik inti besi ke dalam.



Gambar 2.2 *Solenoid door lock*[16]

2.1.3 **Relay 4 Channel 5V**

Relay adalah Saklar (Switch) yang dioperasikan secara listrik dan merupakan komponen Electromechanical (Elektromekanikal) yang terdiri dari 2 bagian utama yakni Elektromagnet (Coil) dan Mekanikal (seperangkat Kontak Saklar/Switch). Relay menggunakan Prinsip Elektromagnetik untuk menggerakkan Kontak Saklar sehingga dengan arus listrik yang kecil (low power) dapat menghantarkan listrik yang bertegangan lebih tinggi.[8]



Gambar 2.3 *Relay 4 Channel 5V*[17]

2.1.4 **Green LED**

Green LED adalah lampu yang dipasangkan pada Arduino yang mengubah sinyal listrik menjadi cahaya.



Gambar 2.4 *Green LED*[18]

2.1.5 *Jumper Cable*

Kabel *jumper* adalah kabel yang digunakan untuk menghubungkan Arduino dengan sensor atau modul lainnya, terdapat tiga jenis kabel jumper, Female to Male, Female to Female, Male to Male, dimana masing masing memiliki fungsi yang sama hanya berbeda pada ujung kabelnya saja.



Gambar 2.5 *Jumper Cable*[19]

2.2 *Software Tools*

2.2.1 *Arduino IDE*

Arduino IDE merupakan lingkungan terintegrasi yang digunakan untuk melakukan pengembangan. Melalui *software* inilah Arduino dilakukan pemrograman untuk melakukan fungsi-fungsi yang dibenamkan melalui sintaks pemrograman. Arduino

menggunakan bahasa pemrograman sendiri yang menyerupai bahasa C. Bahasa pemrograman Arduino sudah dilakukan perubahan untuk memudahkan pemula dalam melakukan pemrograman dari bahasa aslinya. Sebelum dijual ke pasaran, IC mikrokontroler Arduino telah ditanamkan suatu program bernama *Bootlader* yang berfungsi sebagai penengah antara *compiler* Arduino dengan *microcontroller*. [12]

Dalam pembuatan Tugas Akhir ini digunakan beberapa *Library* untuk mendukung berjalannya pemrograman pada Arduino, beberapa *Library* yang digunakan adalah:

Tabel 2.1 *Library* Arduino

Nama Library	Deskripsi
ESP8266WiFi.h	<i>Library</i> yang digunakan untuk menghubungkan ESP8266 pada Arduino dengan <i>internet</i> .
ESP8266HTTPClient.h WiFiClient.h	<i>Library</i> yang memungkinkan untuk ESP8266 pada Arduino mengakses sebuah <i>Hypertext Transfer Protocol</i> (HTTP).

2.2.2 XAMPP

XAMPP adalah sebuah paket perangkat lunak (software) komputer yang sistem penamaannya diambil dari akronim kata Apache, MySQL (dulu) / MariaDB (sekarang), PHP, dan Perl. Sementara imbuhan huruf “X” yang terdapat pada awal kata berasal dari istilah *cross platform* sebagai simbol bahwa aplikasi ini bisa dijalankan di empat sistem operasi berbeda, seperti OS Linux, OS Windows, Mac OS, dan juga Solaris. XAMPP pada Tugas Akhir ini akan digunakan sebagai *Web Server* serta penggunaan *database* MySQL yang telah tersedia pada XAMPP.

2.2.3 *Web Server*

Web Server adalah sebuah software yang berfungsi untuk menerima dan melayani permintaan yang dikirimkan user melalui browser kemudian ditampilkan kepada user sesuai dengan permintaan yang dikirimkan ke server. Bisa dikatakan *web server* merupakan pusat kontrol yang berfungsi untuk memproses permintaan yang diterima dari browser. Layanan ini juga bisa merujuk sebagai software atau hardware.

2.2.4 *HTTP*

HTTP(*Hypertext Transfer Protocol*) merupakan standar arsitektur komunikasi berbasis web yang sering diterapkan dalam pengembangan layanan berbasis web. Secara umum, ketika mengunjungi sebuah website, *browser* akan membuat sebuah *HTTP request* kepada sebuah server. Kemudian server tersebut akan merespons dengan sebuah *resource*, seperti foto, video, atau *HTML* dari halaman web yang kemudian akan ditampilkan oleh *browser*. Pada dasarnya, *HTTP* adalah *stateless* yang berarti semua *request* terpisah satu sama lain. Jadi setiap permintaan dari browser harus mengandung informasi yang cukup untuk server untuk memenuhi *request* tersebut.

Pada *HTTP*, setiap *request* harus memiliki alamat *URL*(*Uniform Resource Locator*) dan *request* harus memiliki sebuah *method* yang kemudian akan ditranslasikan menjadi sebuah pesan *request* sesuai protokol yang telah di spesifikasikan. *HTTP method* yang akan digunakan dalam pembuatan Tugas Akhir ini , yaitu:

1. *GET*

GET digunakan untuk meminta data dari *resource* dimana data tidak diubah. *Request* dengan menggunakan *GET* tidak mengubah kondisi dari *resource*.

2. *POST*

POST digunakan untuk mengirim data ke sebuah server untuk membuat sebuah *resource*.

Ketika *request* sampai ke server, server akan melakukan salah satu dari beberapa aksi dibawah:

1. Server menginterpretasikan request yang diterima, kemudian memetakan request kedalam file di bawah direktori dokumen server, dan mengembalikan file yang diminta kepada client.
2. Sama seperti point pertama, tetapi server akan memetakan request kedalam suatu program yang disimpan di server, menjalankan program, dan mengembalikan output dari program kepada client.
3. Permintaan tidak dapat dipenuhi, server mengembalikan pesan error.

Pesan atau file yang dikembalikan kepada client merupakan sebuah *HTTP response*. [10]

2.2.5 CodeIgniter 3

CodeIgniter 3 adalah sebuah *Framework* berbasis PHP yang digunakan pada pembuatan *website* untuk melakukan *Monitoring Smart Hotel*. [9]

2.2.6 Sublime Text 3

Sublime Text 3 adalah sebuah teks editor yang digunakan penulis dalam melakukan pemograman untuk membangun *Website Monitoring* dan *Controlling Hotel*, serta *web service*. [6]

2.2.7 SQLyog

Sebuah *database client* yang digunakan oleh penulis untuk mempermudah dalam berinteraksi dengan *database*. [7]

2.2.8 Android Studio

Android Studio adalah *Integrated Development Environment* (IDE) resmi untuk pengembangan aplikasi Android, berbasis IntelliJ IDEA. Selain sebagai *code editor* dan alat pengembang IntelliJ, Android Studio menawarkan lebih banyak fitur yang meningkatkan produktivitas saat membangun aplikasi Android.[14]

Dalam pembuatan Tugas Akhir ini digunakan beberapa *Library* untuk mendukung berjalannya pemograman pada Arduino, beberapa *Library* yang digunakan adalah:

Tabel 2.2 *Library* Android Studio

Nama Library	Deskripsi
com.midtrans:uikit:1.23.1-SANDBOX	<i>Library</i> yang digunakan untuk melakukan pembayaran <i>online</i> menggunakan Midtrans.
com.journeyapps:zxing-android-embedded:4.1.0	<i>Library</i> yang memungkinkan untuk melakukan pemindaian <i>QR Code</i> .

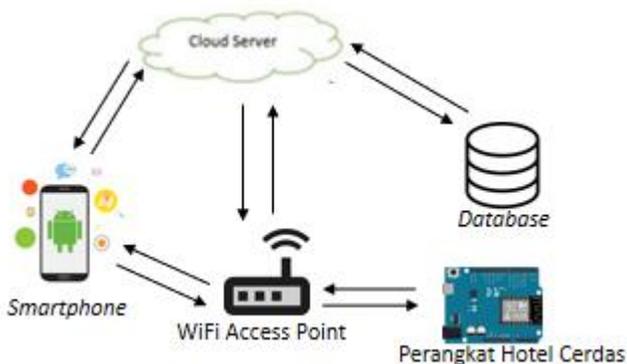
[Halaman ini sengaja dikosongkan]

BAB III PERANCANGAN

Pada bagian ini akan dijelaskan rancangan *hardware* serta sistem yang akan dibangun oleh penulis pada Tugas Akhir ini.

3.1 Arsitektur Sistem

Perancangan dan Implementasi Sistem Hotel Cerdas ini adalah sebuah *prototype* yang didesain untuk membantu sistem perhotelan mengurangi lama proses *check in*, dan tidak membebani pengunjung dengan kunci pintu maupun sebuah kartu untuk membuka pintu kamar dengan sebuah aplikasi *mobile* berbasis Android dan menggunakan *microcontroller* Wemos untuk mengontrol perangkat-perangkat yang berada pada kamar maupun pada hotel itu sendiri. Arsitektur jaringan seperti pada Gambar 3.1 Alur Interaksi Device.



Gambar 3.1 Alur Interaksi Device

Untuk dapat berfungsi, alat ini harus terhubung dengan koneksi internet melalui *Wi-Fi*, yang nantinya akan mencatat dan memvalidasi pengunjung yang melakukan *Scan QR-Code*. Untuk

melakukan implementasi *prototype*, dibutuhkan beberapa komponen penting, yaitu:

1. Aplikasi Hotel Cerdas

Aplikasi Hotel Cerdas atau *AHC* ini adalah sebuah aplikasi *mobile* berbasis android yang memiliki beberapa fungsi, yaitu:

- a. Melakukan pemesanan secara online dan melakukan pembayaran dengan *Go-Pay*.
- b. Melakukan konfirmasi pemesanan kamar atau *Check In* dengan *scan QR-Code*.
- c. Menyediakan sebuah *room controlling* yang dapat digunakan untuk mengendalikan perangkat-perangkat yang terdapat pada kamar hotel, seperti pintu, lampu, kipas, serta dapat mengaktifkan lift dengan *scan QR Code*.
- d. Menyediakan fitur *Room Sharing*, yaitu sebuah fitur yang dapat digunakan untuk memberikan akses kamar kepada user lain agar pengunjung lain dari kamar tersebut tidak perlu khawatir apabila pengunjung utama kamar sedang tidak berada dalam hotel atau kamar.
- e. Melakukan *check out*, baik secara otomatis maupun manual.

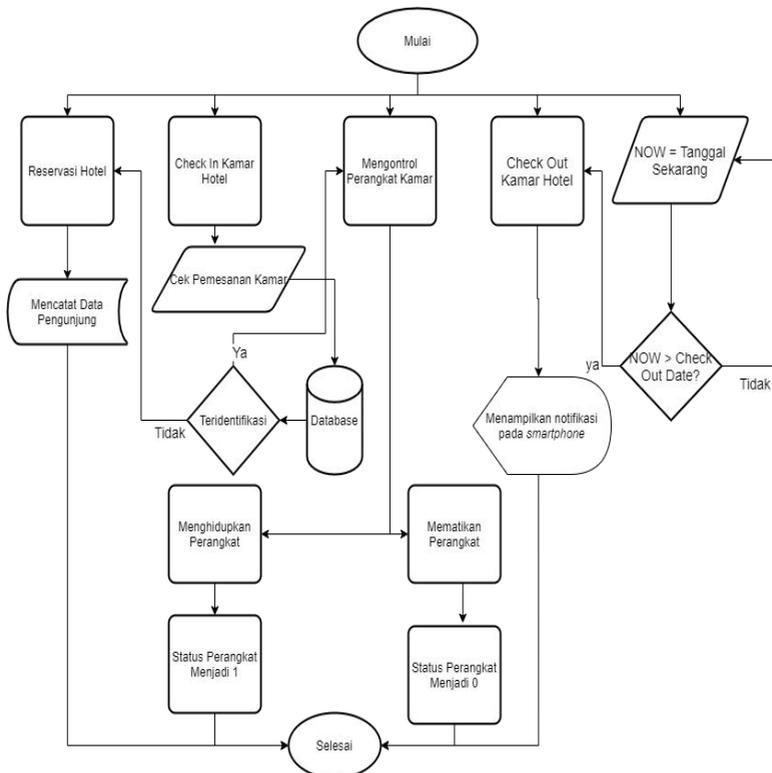
2. *Monitoring System*

Monitoring System adalah sistem berbasis *website* yang dibuat untuk admin atau petugas hotel dalam melakukan *monitoring* dan *controlling* terhadap Hotel Cerdas, dimana sistem monitor dapat melakukan :

- a. Melihat kamar hotel.
- b. Melakukan *check in* untuk pengunjung yang tidak dapat melakukan *scan QR-Code*.
- c. Mengonfirmasi *check out* saat pengunjung melakukan *check out* melalui aplikasi maupun saat

- masa tinggal pengunjung sudah melewati tanggal *check out*.
- d. Mengontrol perangkat-perangkat yang terdapat pada kamar.
 - e. Melihat riwayat transaksi.
 - f. Melihat *QR Code* untuk *check-in*.

Proses Aplikasi Hotel Cerdas dapat dilihat pada Gambar 3.2 *Flowchart* Aplikasi Hotel Cerdas.

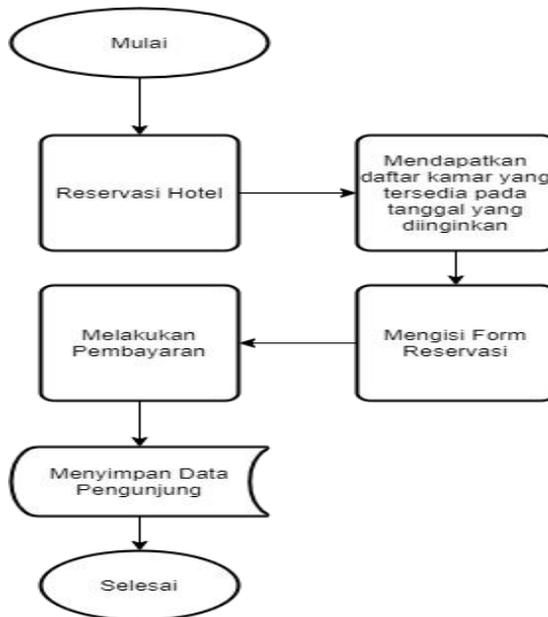


Gambar 3.2 *Flowchart* Aplikasi Hotel Cerdas

Pada *flowchart* Aplikasi Hotel Cerdas memiliki beberapa jenis alur, yaitu:

1. Reservasi Hotel

Pada proses ini pengunjung melakukan reservasi kamar melalui aplikasi. Kemudian pengunjung memasukkan data diri yang diminta aplikasi, dan melakukan pembayaran. Setelah itu data diri pengunjung akan disimpan dalam *database*.

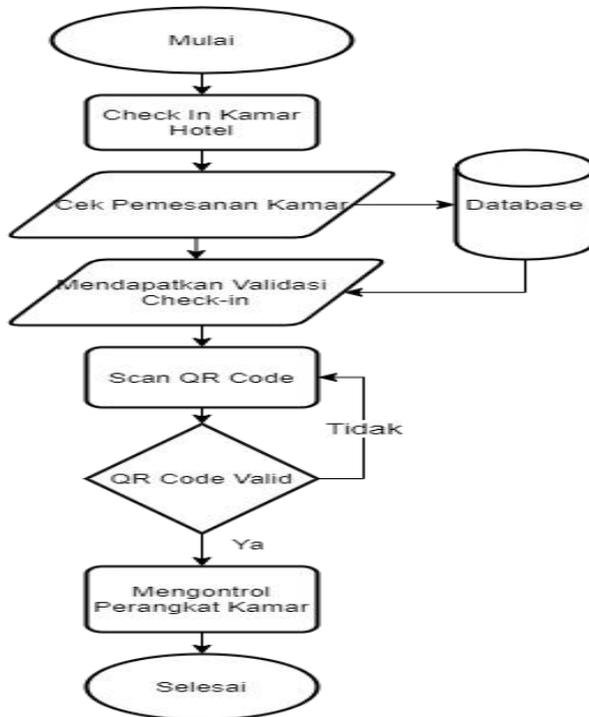


Gambar 3.3 *Flowchart* Alur Reservasi Hotel

2. Check-in Kamar Hotel

Pada proses ini pengunjung melakukan *Check-in* kamar hotel, dimana pengunjung diminta untuk melakukan *scan QR-Code* untuk konfirmasi pemesanan

hotel. Apabila pengunjung belum melakukan pemesanan, maka pengunjung diminta untuk melakukan *flowchart* alur 1(Reservasi Hotel). Sebaliknya, setelah konfirmasi berhasil, menu *Room Controlling* akan bisa diakses.



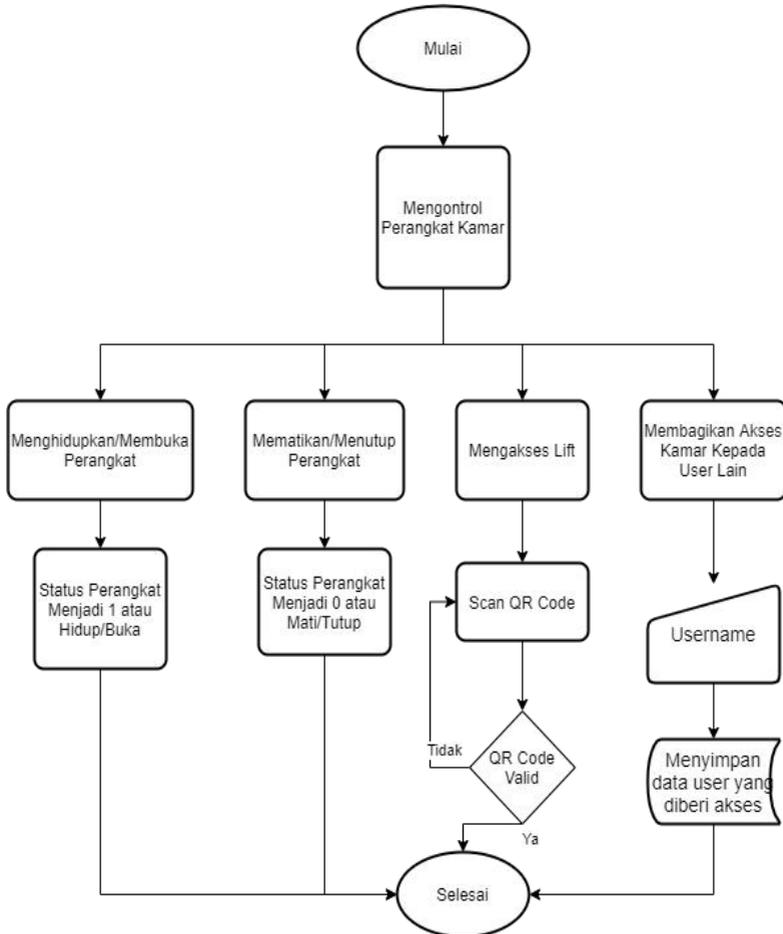
Gambar 3.4 *Flowchart* Alur Check-in Kamar Hotel

3. Mengontrol Perangkat Kamar

Pada proses ini pengunjung melakukan *controlling* kamar melalui akses kamar yang terdapat pada *APH*. Pada akses kamar memiliki beberapa fungsi, seperti:

- a. Mengaktifkan akses *lift*.

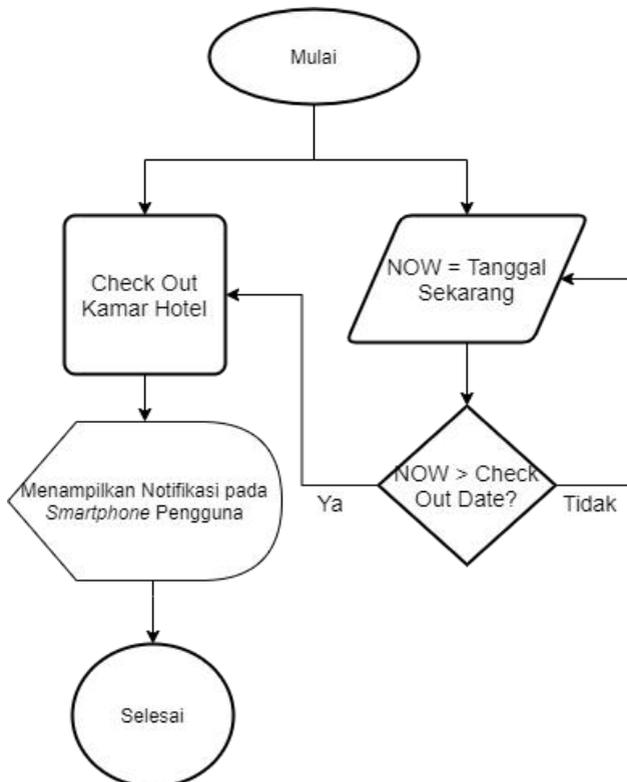
- b. Membuka dan Menutup pintu.
- c. Menghidupkan dan mematikan lampu.
- d. Mengontrol perangkat-perangkat yang terdapat pada kamar.
- e. Membagikan akses kamar kepada user lain.



Gambar 3.5 Flowchart Alur Mengontrol Perangkat

4. *Check-out* Kamar Hotel

Pada proses ini pengunjung melakukan *check-out* kamar hotel, dimana pengunjung akan otomatis *check-out* ketika masa singgah pengunjung sudah habis. Pengunjung juga dapat melakukan *check-out* sebelum masa singgah berakhir dengan cara menekan tombol *check-out* yang terdapat pada akses kamar.



Gambar 3.6 *Flowchart* Aur Check Out

3.2 Perancangan *Database*

Rancangan *Database* ini digunakan sebagai media penyimpanan data oleh Web Service yang nantinya data akan berubah sesuai data yang dikirim oleh Webs Service, yang akan ditampilkan pada Aplikasi Hotel Cerdas. *Database* akan berisi 7 tabel yaitu:

Tabel 3.1 Deskripsi Tabel *Database*

Atribut	Deskripsi Tabel
ht_user	Tabel yang berisi data pengguna.
ht_hotel	Tabel yang berisi data hotel.
ht_kamar	Tabel yang berisi data kamar beserta status-status perangkat.
ht_transaksi	Tabel yang berisi data transaksi pemesanan hotel.
ht_share	Tabel yang berisi data <i>user</i> yang di berikan akses kamar.
ht_lift	Tabel yang berisi lift yang berada di hotel.
ht_admin	Tabel yang berisi data admin hotel.

Penjelasan mengenai rincian tabel pada *database* adalah sebagai berikut:

Tabel 3.2 Detail Tabel ht_user

ht_user		
Atribut	Data Type	Deskripsi
u_id	Integer (11)	Atribut <i>primary key</i> pada tabel ini yang berfungsi sebagai ID <i>primary</i> pada tabel.

u_username	Varchar(100)	Atribut yang berisi username dari <i>user</i> .
u_password	Varchar (50)	Atribut yang berisi password dari <i>user</i> .
u_nama	Varchar(100)	Atribut yang berisi nama dari <i>user</i> .
u_email	Varchar(100)	Atribut yang berisi email dari <i>user</i> .
u_phonenumber	Varchar(50)	Atribut yang berisi nomor telfon dari <i>user</i> .

Tabel 3.3 Detail Tabel ht_hotel

ht_hotel		
Atribut	Data Type	Deskripsi
h_id	Integer (11)	Atribut <i>primary key</i> pada tabel ini yang berfungsi sebagai ID <i>primary</i> pada tabel.
h_nama	Varchar(100)	Atribut yang berisi nama dari hotel.
h_alamat	Varchar(200)	Atribut yang berisi alamat dari hotel.
h_price	Varchar(20)	Atribut yang berisi harga dari hotel.
h_image	Varchar(200)	Atribut yang berisi gambar dari hotel.

Tabel 3.4 Detail Tabel ht_kamar

ht_kamar		
Atribut	Data Type	Deskripsi
k_id	Integer (11)	Atribut <i>primary key</i> pada tabel ini yang berfungsi sebagai ID <i>primary</i> pada tabel.
h_id	Integer (11)	Atribut <i>foreign key</i> pada tabel yang didapat dari tabel ht_hotel.
k_nomor	Varchar(50)	Atribut yang berisi nomor dari kamar.
k_deviceid	Integer(11)	Atribut yang berisi id <i>microcontroller</i> pada kamar.
k_pintu	Varchar(1)	Atribut yang berisi status dari pintu.
k_listrik	Varchar(1)	Atribut yang berisi status dari lampu.
k_kipas	Varchar(1)	Atribut yang berisi status dari kipas.

Tabel 3.5 Detail Tabel ht_transaksi

ht_transaksi		
Atribut	Data Type	Deskripsi
t_id	Integer (11)	Atribut <i>primary key</i> pada tabel ini yang berfungsi sebagai ID <i>primary</i> pada tabel.
u_id	Integer (11)	Atribut <i>foreign key</i> pada tabel yang didapat dari

tabel ht_user.		
h_id	Integer (11)	Atribut <i>foreign key</i> pada tabel yang didapat dari tabel ht_hotel.
k_id	Integer (11)	Atribut <i>foreign key</i> pada tabel yang didapat dari tabel ht_kamar.
t_tanggal	datetime	Atribut yang berisi tanggal transaksi.
m_checkin	datetime	Atribut yang berisi tanggal <i>check-in</i> hotel.
m_checkout	datetime	Atribut yang berisi tanggal <i>check-out</i> hotel.
m_status	Varchar(1)	Atribut yang berisi status dari transaksi, dimana 0 adalah belum <i>check-in</i> , 1 adalah sudah <i>check-in</i> , 2 adalah <i>check-out</i> .
t_totalbayar	Varchar(20)	Atribut yang berisi total pembayaran transaksi.

Tabel 3.6 Detail Tabel ht_share

ht_share		
Atribut	Data Type	Deskripsi
s_id	Integer(11)	Atribut <i>primary key</i> pada tabel ini yang berfungsi sebagai ID <i>primary</i> pada tabel.
t_id	Integer(11)	Atribut <i>foreign key</i> pada tabel yang didapat dari tabel ht_transaksi.

u_id	Integer(11)	Atribut <i>foreign key</i> pada tabel yang didapat dari tabel ht_user.
------	-------------	--

Tabel 3.7 Detail Tabel ht_lift

ht_lift		
Atribut	Data Type	Deskripsi
l_id	Integer(11)	Atribut <i>primary key</i> pada tabel ini yang berfungsi sebagai ID <i>primary</i> pada tabel.
h_id	Integer(11)	Atribut <i>foreign key</i> pada tabel yang didapat dari tabel ht_hotel.
l_lokasi	Varchar(50)	Atribut yang berisi lokasi dari <i>lift</i> pada hotel.
l_status	Integer(1)	Atribut yang berisi status dari <i>lift</i> .

Tabel 3.8 Detail Tabel ht_admin

ht_admin		
Atribut	Data Type	Deskripsi
a_id	Integer (11)	Atribut <i>primary key</i> pada tabel ini yang berfungsi sebagai ID <i>primary</i> pada tabel.
a_username	Varchar(50)	Atribut yang berisi username dari admin.
a_password	Varchar(50)	Atribut yang berisi

		password dari admin.
h_id	Integer(11)	Atribut <i>foreign key</i> pada tabel yang didapat dari tabel ht_hotel.

Tabel 3.9 Detail Tabel ht_log_transaksi

ht_share		
Atribut	Data Type	Deskripsi
t_id	Integer (11)	Atribut <i>primary key</i> pada table ini yang berfungsi sebagai ID <i>primary</i> pada tabel ht_log_transaksi yang berfungsi sebagai penyimpanan riwayat transaksi.
u_id	Integer (11)	Atribut <i>foreign key</i> pada tabel yang didapat dari tabel ht_user.
h_id	Integer (11)	Atribut <i>foreign key</i> pada tabel yang didapat dari tabel ht_hotel.
k_id	Integer (11)	Atribut <i>foreign key</i> pada tabel yang didapat dari tabel ht_kamar.
t_tanggal	datetime	Atribut yang

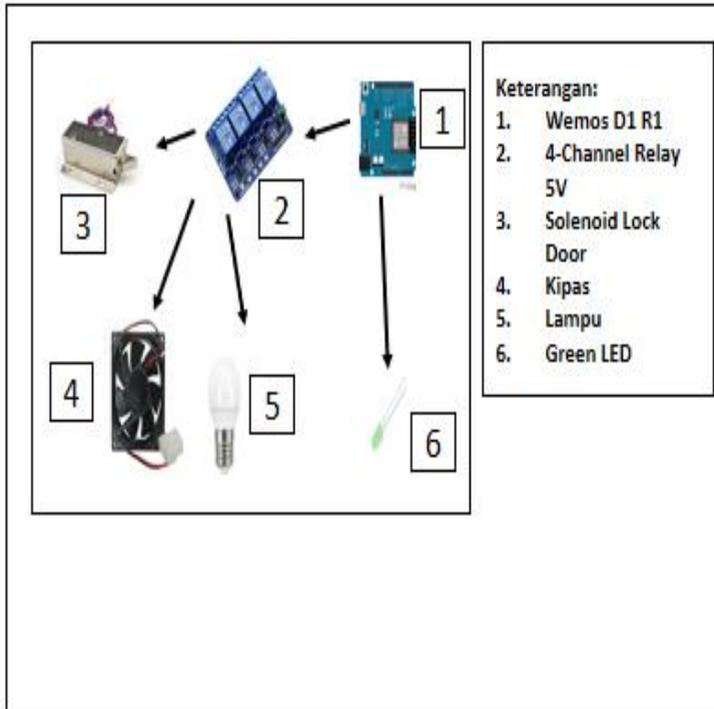
		berisi tanggal transaksi.
m_checkin	datetime	Atribut yang berisi tanggal <i>check-in</i> hotel.
m_checkout	datetime	Atribut yang berisi nomor dari kamar.
m_status	Varchar(1)	Atribut yang berisi status dari transaksi, dimana 0 adalah belum <i>check-in</i> , 1 adalah sudah <i>check-in</i> , 2 adalah <i>check-out</i> .
t_totalbayar	Varchar(20)	Atribut yang berisi total pembayaran transaksi.

3.3 Perancangan Alat

Perancangan prototipe *Smart Hotel* akan menggunakan beberapa komponen *hardware*, dengan WeMos D1 R1 sebagai *microcontroller*, diantaranya :

1. 4-Channel Relay 5V
2. Solenoid Lock Door 12V
3. Lampu
4. Kipas
5. Green LED

Microcontroller akan dihubungkan dengan masing masing komponen melalui 4-Channel Relay 5V dengan mengirimkan arus listrik kepada relay seperti pada Gambar 3.7 Interaksi *Microcontroller* dengan Alat.



Gambar 3.7 Interaksi *Microcontroller* dengan Alat

Microcontroller akan mengirimkan sinyal *High* dan *Low* kepada relay. Ketika *microcontroller* mengirimkan sinyal *Low* maka relay akan aktif dan saklar akan tertutup sehingga terminal NO(*Normally Open*) akan mendapatkan arus, sedangkan saat *microcontroller* mengirimkan sinyal *High* maka relay akan mati dan saklar akan terbuka sehingga terminal NC(*Normally Closed*) akan mendapatkan arus. dan membuka saklar. Komponen-komponen *hardware* akan terhubung dengan relay dengan beberapa ketentuan sebagai berikut:

1. Lampu terhubung pada relay IN 1 pada terminal NO.
2. Kipas terhubung pada relay IN 3 pada terminal NO.

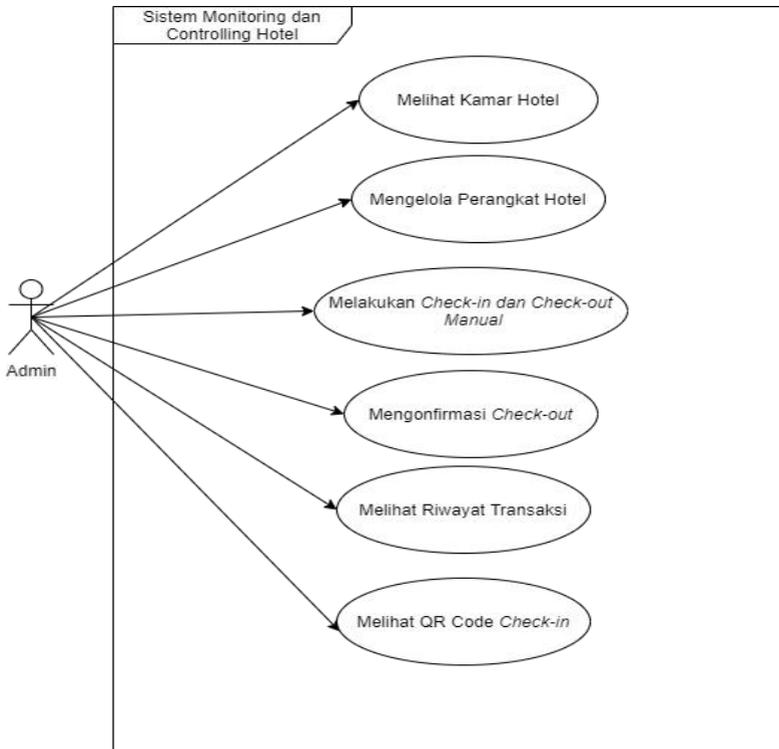
3. Solenoid Lock Door 12V terhubung pada relay IN 4 pada terminal NO.

Microcontroller yang terhubung pada 4-Channel Relay 5V akan berfungsi sebagai berikut:

1. Membuka pintu dengan mengirimkan sinyal *Low* saat $k_pintu = 1$ pada tabel *ht_kamar* dan menutup pintu dengan mengirimkan sinyal *High* saat $k_pintu = 0$ pada tabel *ht_kamar* kepada relay.
2. Menghidupkan lampu dengan mengirimkan sinyal *Low* saat $k_lampu = 1$ pada tabel *ht_kamar* dan mematikan lampu dengan mengirimkan sinyal *High* saat $k_lampu = 0$ pada tabel *ht_kamar* kepada relay.
3. Menghidupkan kipas dengan mengirimkan sinyal *Low* saat $k_kipas = 1$ pada tabel *ht_kamar* dan mematikan kipas dengan mengirimkan sinyal *High* saat $k_kipas = 0$ pada tabel *ht_kamar* kepada relay.
4. Mengaktifkan Lift dengan mengirimkan sinyal *Low* saat $l_status = 1$ pada tabel *ht_lift* dan mematikan kipas dengan mengirimkan sinyal *High* saat $l_status = 0$ pada tabel *ht_lift* kepada relay.

3.4 Perancangan Sistem *Administrator*

Perancangan Sistem *Administrator* yang akan digunakan admin hotel *monitoring* dan *controlling* Hotel Cerdas digambarkan melalui *Use Case Diagram* pada Gambar 3.7 *Use Case Diagram* Sistem Administrator.



Gambar 3.8 Use Case Diagram Sistem Administrator

Sistem Monitor dibangun menggunakan *framework* CodeIgniter 3 akan digunakan oleh admin hotel untuk melakukan *monitoring* dan *controlling* Sistem Hotel Cerdas, yang dapat melakukan:

1. Melihat kamar hotel.
2. Mengelola perangkat hotel.
3. Melakukan *check-in* dan *check-out* manual bagi pengunjung hotel yang memiliki kendala pada *smartphone* Android mereka.

4. Mengonfirmasi *check-out* ketika masa tinggal pengunjung hotel sudah habis atau pengunjung melakukan *check-out* secara manual.
5. Melihat riwayat transaksi
6. Melihat *QR Code* untuk *check-in*.

[Halaman ini sengaja dikosongkan]

BAB IV IMPLEMENTASI

Pada bab ini akan dijelaskan tentang implementasi yang dilakukan berdasarkan perancangan pada bab sebelumnya

4.1 Implementasi *Database*

Pada implementasi *database* ini, penulis menggunakan *tool SQLyog* untuk mengakses *database*, dan XAMPP sebagai penyimpanan *database* MySQL serta *Web Server*, *database* akan dibuatkan tabel seperti yang telah dijelaskan pada bab perancangan.

4.1.1 Implementasi Tabel

4.1.1.1 Implementasi Tabel ht_user

Pada implementasi tabel ht_user akan menjalankan *query* seperti Kode Sumber 4.1 pada *SQLyog*

```
1. CREATE TABLE `ht_user` (  
2. `u_id` int(11) NOT NULL AUTO_INCREMENT,  
3. `u_username` varchar(100) DEFAULT NULL,  
4. `u_password` varchar(50) DEFAULT NULL,  
5. `u_nama` varchar(100) DEFAULT NULL,  
6. `u_email` varchar(100) DEFAULT NULL,  
7. `u_phonenumber` varchar(50) DEFAULT NULL,  
8. PRIMARY KEY (`u_id`)  
9. ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

Kode Sumber 4.1 MySQL ht_user

4.1.1.2 Implementasi Tabel ht_hotel

Pada implementasi tabel ht_hotel akan menjalankan *query* seperti Kode Sumber 4.2 pada *SQLyog*

```

1. CREATE TABLE `ht_hotel` (
2.   `h_id` int(11) NOT NULL AUTO_INCREMENT,
3.   `h_nama` varchar(100) DEFAULT NULL,
4.   `h_alamat` varchar(200) DEFAULT NULL,
5.   `h_price` varchar(20) DEFAULT NULL,
6.   `h_image` varchar(200) DEFAULT NULL,
7.   PRIMARY KEY (`h_id`)
8. ) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

Kode Sumber 4.2 MySQL ht_hotel

4.1.1.3 Implementasi Tabel ht_kamar

Pada implementasi tabel ht_kamar akan menjalankan *query* seperti Kode Sumber 4.3 pada *SQLyog*.

```

1. CREATE TABLE `ht_kamar` (
2.   `k_id` int(11) NOT NULL AUTO_INCREMENT,
3.   `h_id` int(11) DEFAULT NULL,
4.   `k_nomor` varchar(50) DEFAULT NULL,
5.   `k_deviceid` int(11) DEFAULT NULL,
6.   `k_pintu` varchar(1) NOT NULL DEFAULT '0',
7.   `k_listrik` varchar(1) NOT NULL DEFAULT '0',
8.   `k_kipas` varchar(1) DEFAULT '0',
9.   PRIMARY KEY (`k_id`),
10.  KEY `fk_kamar_hotel` (`h_id`),
11.  CONSTRAINT `fk_kamar_hotel` FOREIGN KEY(`h_id`) REFERENCES `ht_hotel` (`h_id`) ON UPDATE CASCADE
12. ) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

Kode Sumber 4.3 MySQL ht_kamar

4.1.1.4 Implementasi Tabel ht_transaksi

Pada implementasi tabel ht_transaksi akan menjalankan *query* seperti Kode Sumber 4.4 pada *SQLyog*.

```

1. CREATE TABLE `ht_transaksi` (
2.   `t_id` int(11) NOT NULL AUTO_INCREMENT,
3.   `u_id` int(11) DEFAULT NULL,
4.   `h_id` int(11) DEFAULT NULL,
5.   `k_id` int(11) DEFAULT NULL,
6.   `t_tanggal` datetime NOT NULL DEFAULT CURRENT_TIMESTAMP,
7.   `m_checkin` datetime DEFAULT NULL,
8.   `m_checkout` datetime DEFAULT NULL,
9.   `m_status` varchar(1) NOT NULL DEFAULT '0',
10.  `t_totalbayar` varchar(20) DEFAULT NULL,
11.  PRIMARY KEY (`t_id`),
12.  KEY `FK_Transaksi_User` (`u_id`),
13.  KEY `FK_Transaksi_Hotel` (`h_id`),
14.  KEY `FK_Transaksi_Kamar` (`k_id`),
15.  CONSTRAINT `FK_Transaksi_Hotel` FOREIGN KEY (`h_id`)
    REFERENCES `ht_hotel` (`h_id`) ON UPDATE CASCADE,
16.  CONSTRAINT `FK_Transaksi_Kamar` FOREIGN KEY (`k_id`)
    REFERENCES `ht_kamar` (`k_id`) ON UPDATE CASCADE,
17.  CONSTRAINT `FK_Transaksi_User` FOREIGN KEY (`u_id`)
    REFERENCES `ht_user` (`u_id`) ON UPDATE CASCADE
18. ) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

Kode Sumber 4.4 MySQL ht_transaksi

4.1.1.5 Implementasi Tabel ht_share

Pada implementasi tabel ht_share akan menjalankan query seperti Gambar 4.5 pada *SQLyog*.

```

1. CREATE TABLE `ht_share` (
2.   `s_id` int(11) NOT NULL AUTO_INCREMENT,
3.   `u_id` int(11) DEFAULT NULL,
4.   `t_id` int(11) DEFAULT NULL,
5.   PRIMARY KEY (`s_id`),

```

```

6.   KEY `FK_Share_Transaksi` (`t_id`),
7.   KEY `FK_Share_User` (`u_id`),
8.
9.   CONSTRAINT `FK_Share_Transaksi` FOREIGN KEY (`t_id`
REFERENCES `ht_transaksi` (`t_id`) ON DELETE
CASCADE ON UPDATE CASCADE,
10.
11.  CONSTRAINT `FK_Share_User` FOREIGN KEY (`u_id`)
REFERENCES `ht_user` (`u_id`) ON DELETE NO ACTION
ON UPDATE CASCADE
12. ) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

Kode Sumber 4.5 MySQL ht_share

4.1.1.6 Implementasi Tabel ht_lift

Pada implementasi tabel ht_lift akan menjalankan *query* seperti Kode Sumber 4.6 pada *SQLyog*.

```

1.  CREATE TABLE `ht_lift` (
2.  `l_id` int(11) NOT NULL AUTO_INCREMENT,
3.  `h_id` int(11) DEFAULT NULL,
4.  `l_lokasi` varchar(50) DEFAULT NULL,
5.  `l_status` int(1) DEFAULT '0',
6.  PRIMARY KEY (`l_id`),
7.  KEY `FK_Lift_Hotel` (`h_id`),
8.
9.  CONSTRAINT `FK_Lift_Hotel` FOREIGN KEY (`h_id`)
REFERENCES `ht_hotel` (`h_id`) ON DELETE CASCADE
ON UPDATE CASCADE
10. ) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

Kode Sumber 4.6 MySQL ht_lift

4.1.1.7 Implementasi Tabel ht_admin

Pada implementasi tabel ht_admin akan menjalankan *query* seperti Kode Sumber 4.7 pada *SQLyog*.

```

1. CREATE TABLE `ht_lift` (
2.   `l_id` int(11) NOT NULL AUTO_INCREMENT,
3.   `h_id` int(11) DEFAULT NULL,
4.   `l_lokasi` varchar(50) DEFAULT NULL,
5.   `l_status` int(1) DEFAULT '0',
6.   PRIMARY KEY (`l_id`),
7.   KEY `FK_Lift_Hotel` (`h_id`),
8.
9.   CONSTRAINT `FK_Lift_Hotel` FOREIGN KEY (`h_id`)
REFERENCES `ht_hotel` (`h_id`) ON
DELETE CASCADE ON UPDATE CASCADE
10. ) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

Kode Sumber 4.7 MySQL ht_admin

4.1.1.8 Implementasi Tabel ht_log_transaksi

Pada implementasi tabel ht_log_transaksi akan menjalankan *query* seperti Kode Sumber 4.8 pada *SQLyog*.

```

1. CREATE TABLE `ht_log_transaksi` (
2.   `t_id` int(11) NOT NULL AUTO_INCREMENT,
3.   `u_id` int(11) DEFAULT NULL,
4.   `h_id` int(11) DEFAULT NULL,
5.   `k_id` int(11) DEFAULT NULL,
6.   `t_tanggal` datetime NOT NULL DEFAULT CURRENT_TIMESTAMP,
7.   `m_checkin` datetime DEFAULT NULL,
8.   `m_checkout` datetime DEFAULT NULL,
9.   `m_status` varchar(1) NOT NULL DEFAULT '0',
10.  `t_totalbayar` varchar(20) DEFAULT NULL,
11.  PRIMARY KEY (`t_id`),
12.
13.  KEY `FK_Transaksi_User` (`u_id`),

```

```

14.
15.   KEY `FK_Transaksi_Hotel` (`h_id`),
16.
17.   KEY `FK_Transaksi_Kamar` (`k_id`)
18. ) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

Kode Sumber 4.8 MySQL ht_log_transaksi

4.1.2 Implementasi *Trigger* dan *Event*

Kemudian dibuatkan *trigger* dan *event* pada *database* yang berfungsi untuk melakukan *back up* dari tabel ht_transaksi ke tabel ht_log_transaksi saat terjadi *Insert*, *Update* atau *Delete* dan melakukan pengecekan terhadap ht_transaksi saat tanggal hari ini lebih besar dari pada tanggal m_checkout pada ht_transaksi, maka *event* akan melakukan *update* m_status menjadi 2 pada ht_transaksi.

4.1.2.1 Implementasi *Trigger* update_log_after_insert

Pada implementasi *Trigger* update_log_after_insert akan menjalankan *query* seperti Kode Sumber 4.9 pada *SQLyog*.

```

1. CREATE TRIGGER `update_log_after_insert` AFTER INSE
RT ON `ht_transaksi` FOR EACH ROW BEGIN
2.   INSERT INTO ht_log_transaksi SELECT * FROM ht_t
ransaksi ON DUPLICATE KEY UPDATE
3.     t_id = VALUES(t_id),
4.     u_id = VALUES(u_id),
5.     h_id = VALUES(h_id),
6.     k_id = VALUES(k_id),
7.     t_tanggal = VALUES(t_tanggal),
8.     m_checkin = VALUES(m_checkin),
9.     m_checkout = VALUES(m_checkout),
10.    m_status = VALUES(m_status);
11.

```

12. **END**

Kode Sumber 4.9 MySQL Trigger After Insert

4.1.2.2 Implementasi *Trigger* update_log_after_update

Pada pengimplementasian *Trigger* update_log_after_update akan menjalankan *query* seperti Kode Sumber 4.10 pada *SQLyog*.

```

1. CREATE TRIGGER `update_log_after_update` AFTER UPDA
TE ON `ht_transaksi` FOR EACH ROW BEGIN
2.     INSERT INTO ht_log_transaksi SELECT * FROM
ht_transaksi ON DUPLICATE KEY UPDATE
3.         t_id = VALUES(t_id),
4.         u_id = VALUES(u_id),
5.         h_id = VALUES(h_id),
6.         k_id = VALUES(k_id),
7.         t_tanggal = VALUES(t_tanggal),
8.         m_checkin = VALUES(m_checkin),
9.         m_checkout = VALUES(m_checkout),
10.        m_status = VALUES(m_status);
11.    END

```

Kode Sumber 4.10 MySQL Trigger After Update

4.1.2.3 Implementasi *Trigger* update_log_before_delete

Pada pengimplementasian *Trigger* update_log_before_delete akan menjalankan *query* seperti Kode Sumber 4.11 pada *SQLyog*.

```

1. CREATE TRIGGER `update_log_before_delete` BEFORE DE
LETE ON `ht_transaksi` FOR EACH ROW BEGIN
2.     INSERT INTO ht_log_transaksi SELECT * FROM ht_t
ransaksi ON DUPLICATE KEY UPDATE
3.         t_id = VALUES(t_id),
4.         u_id = VALUES(u_id),

```

```

5.         h_id = VALUES(h_id),
6.         k_id = VALUES(k_id),
7.         t_tanggal = VALUES(t_tanggal),
8.         m_checkin = VALUES(m_checkin),
9.         m_checkout = VALUES(m_checkout),
10.        m_status = VALUES(m_status);
11.     END

```

Kode Sumber 4.11 MySQL Trigger Before Delete

4.1.2.4 Implementasi *Event* checkout_handler

Pada pengimplementasian *Event* checkout_handler akan menjalankan *query* seperti Kode Sumber 4.12 pada *SQLyog*.

```

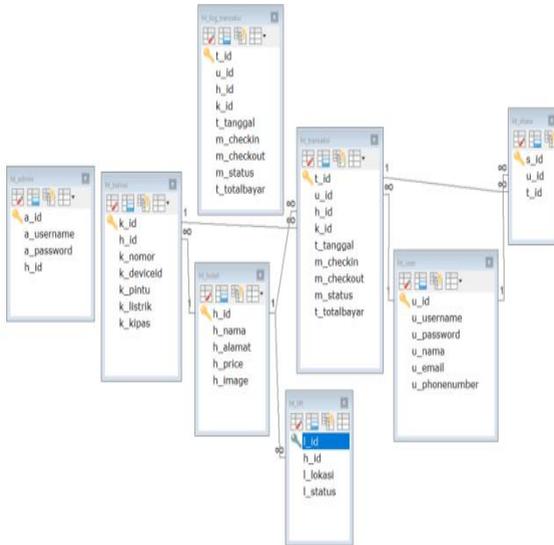
1.  CREATE EVENT `backup_transaksi` ON SCHEDULE EVERY 5
    SECOND STARTS '2020-05-
    28 04:23:11' ON COMPLETION PRESERVE ENABLE DO BEGIN
2.      delete from ht_transaksi WHERE NOW() > m_ch
    eckout and m_status = '0';
3.      UPDATE ht_transaksi SET m_status='2' WHERE
    NOW() > m_checkout;
4.  END

```

Kode Sumber 4.12 MySQL Event Checkout Handler

4.1.3 Hasil *Query* MySQL

Hasil dari implementasi *query* diatas akan menghasilkan skema *database* seperti pada Gambar 4.1.



Gambar 4.1 Hasil Eksekusi MySQL

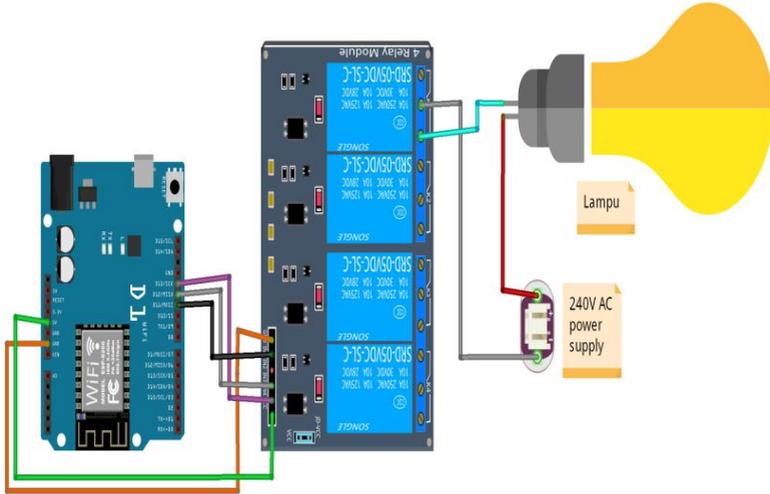
4.2 Implementasi Alat

Implementasi alat sistem Hotel Cerdas diperlukan beberapa rangkaian alat yang dihubungkan pada Wemos seperti penjelasan pada bab perancangan, *Microcontroller* akan dihubungkan dengan masing masing komponen melalui Relay 4 *Channel 5V* dengan mengirimkan arus listrik kepada relay, di mana microcontroller akan mengirimkan sinyal Low dan High untuk menghidupkan dan mematikan relay. Kemudian saat relay hidup saklar akan tertutup dan arus akan mengalir ke terminal NO(Normally Open).

4.2.1 Rangkaian Alat

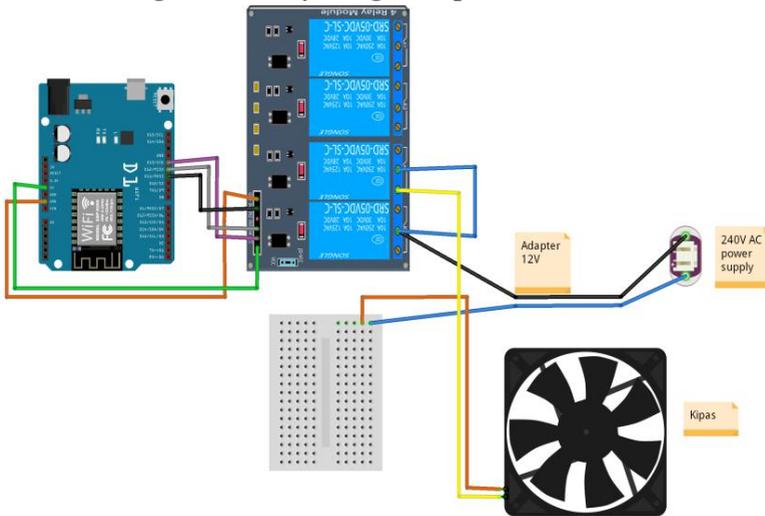
Seluruh rangkaian alat pada sistem Hotel Cerdas yang terhubung dengan relay akan menggunakan terminal NO. Rangkaian sistem Hotel Cerdas adalah sebagai berikut:

4.2.1.3 Rangkaian Relay dengan Lampu



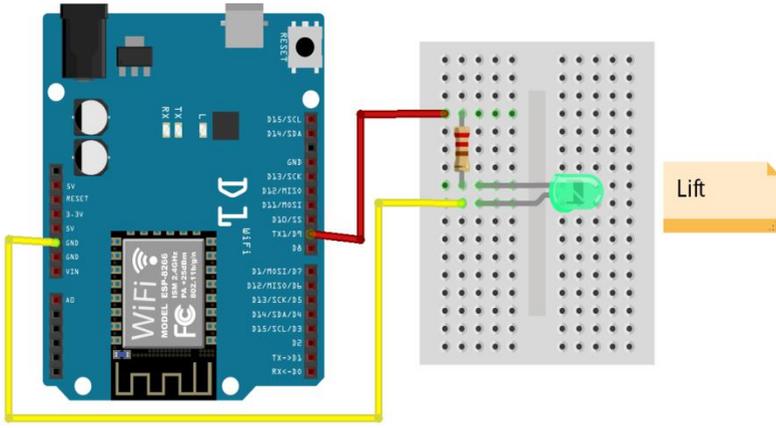
Gambar 4.4 Rangkaian Relay dengan Lampu

4.2.1.4 Rangkaian Relay dengan Kipas



Gambar 4.5 Rangkaian Relay dengan Kipas

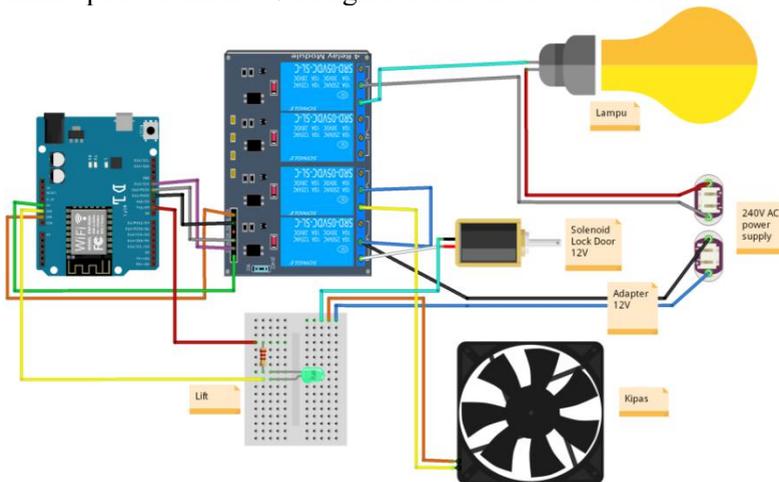
4.2.1.5 Rangkaian Wemos dengan *Green LED*



Gambar 4.6 Rangkaian Wemos dengan *Green LED*

4.2.1.6 Rangkaian Sistem Hotel Cerdas

Rangkaian sistem Hotel Cerdas secara keseluruhan dapat dilihat pada Gambar 4.7 Rangkaian Sistem Hotel Cerdas.



Gambar 4.7 Rangkaian Sistem Hotel Cerdas

Tabel 4.1 Rangkaian Sistem Hotel Cerdas

PIN		Perangkat Terhubung	
4-Channel Relay 5V	Wemos D1 R1	Jenis Terminal	Nama Perangkat
GND	GND	-	-
IN 1	D11	NO	Lampu
IN 3	D12	NO	Kipas
IN 4	D13	NO	<i>Solenoid Lock Door</i>
VCC	5V	-	-
-	D9	-	<i>Green LED</i>

Rangkaian sistem memiliki komponen-komponen yang terhubung pada *microcontroller* Wemos D1 R1, yaitu:

1. *4-Channel Relay 5V*

Pada relay akan dihubungkan perangkat-perangkat pada kamar, yaitu *Solenoid Lock Door*, Lampu, Kipa. Perangkat akan dihubungkan pada relay melalui terminal NO(Normally Open) dan relay akan diberikan aliran listrik pada pin IN 1, IN 2, IN 4 melalui pin D11, D12, D13 pada Wemos D1 R1 untuk menutup saklar pada relay dan mengaktifkan perangkat.

2. *Green LED*

LED akan dihubungkan langsung pada Wemos D1 R1 melalui pin D9.

4.2.2 Implementasi Arduino

Rangkaian sistem dapat dijalankan dengan memasukan kode program ke dalam Arduino melalui Arduino IDE. Implementasi fungsi-fungsi pada Arduino akan dijelaskan dan

ditampilkan dalam bentuk *pseudocode*. Kode program lengkap akan dilampirkan sepenuhnya pada bagian Lampiran.

4.2.2.1 Library dan Konfigurasi WiFi

Bagian ini berisi semua *library* yang digunakan pada *microcontroller* serta konfigurasi untuk koneksi WiFi dan alamat host.

```

1. #include <ESP8266WiFi.h>
2. #include <WiFiClient.h>
3. #include <ESP8266WebServer.h>
4. #include <ESP8266HTTPClient.h>
5. const char *ssid = "wifirumah";
6. const char *password = "rumahwifi";
7. const char *host= "192.168.1.103";
8. const String URL_ADDRESS="http://"+host+"/Hotel/Myroom/getStsArd?devid="+deviceid";

```

Kode Sumber 4.13 Library dan Konfigurasi WiFi

4.2.2.2 Deklarasi Pin

Bagian ini berisi deklarasi pin yang dipakai relay, dan konfigurasi untuk mendapatkan respon yang diinginkan dari web service.

```

1. const int in0 = D11; // Relay pin IN 1
2. const int in1 = D9; // Relay pin IN 2
3. const int in2 = D12; // Relay pin IN 3
4. const int in3 = D13; // Relay pin IN 4

```

Kode Sumber 4.14 Deklarasi Pin

4.2.2.3 Setup Arduino

Bagian ini adalah fungsi setup Arduino, fungsi ini akan dijalankan secara otomatis oleh Arduino ketika terhubung dengan arus listrik, fungsi ini akan menjalankan fungsi connectWifi yang

berisi kode program untuk menghubungkan Arduino dengan *WiFi* dan *Web Server*.

```

1. void setup() {
2.     Serial.begin(9600);
3.     connectWiFi(); // Fungsi connect WiFi
4.     pinMode(in0, OUTPUT); //Pin pintu
5.     pinMode(in1, OUTPUT); //Pin lampu
6.     pinMode(in2, OUTPUT); //Pin kipas
7.     pinMode(in3, OUTPUT); //Pin lift
8. }

```

Kode Sumber 4.15 *Setup* Arduino

```

1. // ----- Function Connect to WiFi
2. Void connectWiFi(){
3.     WiFi.mode(WIFI_OFF);
4.     delay(1000);
5.     WiFi.mode(WIFI_STA);
6.     WiFi.begin(ssid, password);
7.     Serial.println(""); Serial.print("Connecting");
8.     while (WiFi.status() != WL_CONNECTED) {
9.         delay(500);
10.    Serial.print(".");
11.    }
12. }

```

Kode Sumber 4.16 Fungsi *Connect WiFi*

4.2.2.4 Implementasi Loop Program

Setelah menjalankan fungsi void setup, Arduino akan melanjutkan nya dengan membaca fungsi void loop, fungsi ini akan terus melakukan pengulangan selama Arduino hidup, pada fungsi loop ini Arduino akan melakukan *HTTP(Hypertext Transfer Protocol) request* kepada *web server*, dan *web server* akan mengembalikan sebuah *response*. Arduino melakukan *HTTP*

request akan menggunakan metode GET dengan parameter *id* dari Arduino yang mengirimkan *request*. Kemudian, *Web Server* akan mengembalikan sebuah *response* berisi status dari pintu, lampu, dan kipas.

```

1. // ----- Function Loop
2. void loop() {
3.     getData(); //HTTP Request data
4.     packData(); //Menyimpan response
5.     cekPintu(); //Buka atau Tutup pintu
6.     cekLampu(); //Hidupkan atau Matikan lampu
7.     cekKipas(); //Hidupkan atau Matikan kipas
8.     cekLift(); //Hidupkan atau Matikan lift
9. }

```

Kode Sumber 4.17 Fungsi *Loop*

```

function getData(){
    IF client melakukan request THEN
        Mencoba mendapatkan data;
        IF data ada
            return data
        ELSE
            return error
    END
}

```

Gambar 4.8 *Pseudocode* Fungsi Mengirim *HTTP Request*

Pseudocode fungsi `getData()` ini digunakan oleh *client* untuk mengirimkan *request* ke *web server*, jika *server* mengirimkan *response* dengan data yang diminta maka *client* akan mendapatkan data tersebut.

4.2.2.5 Implementasi Pintu Kamar Hotel

Bagian ini adalah implementasi dari pintu kamar hotel yang menggunakan *Solenoid Lock Door*. Implementasi fungsi membuka dan menutup pintu ditampilkan dengan nama fungsi cekPintu() dalam bentuk *pseudocode* sebagai berikut.

```
function cekPintu(){
  IF status pintu = 1 THEN
    Buka pintu
  ELSE
    Tutup pintu
  END
}
```

Gambar 4.9 *Pseudocode* Membuka dan Menutup Pintu

Pseudocode fungsi cekPintu() ini digunakan untuk membuka dan menutup pintu. Jika status pintu adalah 1, maka pintu terbuka dan jika status pintu adalah 0, maka pintu tertutup. Kode lengkap program akan dilampirkan pada lampiran.

4.2.2.6 Implementasi Lampu Kamar Hotel

Bagian ini adalah implementasi dari lampu kamar hotel. Implementasi fungsi menghidupkan dan mematikan lampu ditampilkan dengan nama fungsi cekLampu() dalam bentuk *pseudocode* sebagai berikut.

```
function cekLampu(){
  IF status lampu= 1 THEN
    Lampu hidup
  ELSE
    Lampu mati
  END
}
```

Gambar 4.10 *Pseudocode* Menghidupkan dan Mematikan Lampu

Pseudocode fungsi cekLampu() ini digunakan untuk menghidupkan dan mematikan lampu. Jika status lampu adalah 1, maka lampu hidup dan jika status lampu adalah 0, maka lampu mati. Kode lengkap program akan dilampirkan pada lampiran.

4.2.2.7 Implementasi Kipas Kamar Hotel

Bagian ini adalah implementasi dari kipas kamar hotel. Implementasi fungsi menghidupkan dan mematikan kipas ditampilkan dengan nama fungsi cekKipas() dalam bentuk *pseudocode* sebagai berikut.

```
function cekKipas(){
  IF status kipas = 1 THEN
    Kipas hidup
  ELSE
    Kipas mati
  END
}
```

Gambar 4.11 *Pseudocode* Menghidupkan dan Mematikan Kipas

Pseudocode fungsi cekKipas() ini digunakan untuk menghidupkan dan mematikan kipas. Jika status kipas adalah 1, maka kipas hidup dan jika status kipas adalah 0, maka kipas mati. Kode lengkap program akan dilampirkan pada lampiran.

4.2.2.8 Implementasi Lift Hotel

Bagian ini adalah implementasi dari lampu kamar hotel. Implementasi fungsi menghidupkan dan mematikan lift ditampilkan dengan nama fungsi cekLift() dalam bentuk *pseudocode* sebagai berikut.

```

function cekKipas(){
  IF status lift= 1 THEN
    Lift hidup
  ELSE
    Lift mati
  END
}

```

Gambar 4.12 *Pseudocode* Menghidupkan dan Mematikan Lift

Pseudocode fungsi cekLift() ini digunakan untuk menghidupkan dan mematikan lift. Jika status lift adalah 1, maka lift hidup dan jika status lift adalah 0, maka lift mati. Kode lengkap program akan dilampirkan pada lampiran.

4.3 Implementasi Sistem Berbasis Android

Aplikasi Hotel Cerdas akan memiliki beberapa fungsi yang akan digunakan untuk memesan kamar dengan pembayaran online menggunakan *Go-Pay*, *monitoring* dan *controlling* kamar, serta mengimplementasikan fitur membagi kamar dengan pengguna lain yang menyewa kamar tersebut bersama pengunjung.

Aplikasi akan diimplementasikan menggunakan Android Studio. Pada aplikasi ini, proses penyimpanan data sama seperti Arduino dengan mengirimkan *HTTP Request* kepada *Web Server* dan menerima *HTTP Response*. Untuk itu diperlukan sebuah class untuk melakukan koneksi ke *Web Server* serta fungsi untuk mengirimkan. Untuk melakukan koneksi akan menggunakan sebuah kelas *controller* yang bernama *Connector.java* yang akan menghubungkan Aplikasi ke *Web Server*. Kelas ini akan memiliki 2 fungsi, yaitu fungsi *connect* yang digunakan untuk menghubungkan koneksi dengan *web server* dan fungsi *send* yang digunakan untuk mengirim *request* dan menerima *response*

dari *web server*. Implementasi kelas *Connector.java* akan di tampilkan dalam bentuk *pseudocode* sebagai berikut.

```

function connect(urlAddress){
  Mencoba koneksi ke urlAddress
  IF Koneksi berhasil THEN
    return koneksi
  ELSE
    return null
}

function send(urlAdress, postdata){
  Koneksi = connect
  IF tidak ada koneksi THEN
    return null

  Mengirimkan data

  IF mendapatkan response HTTP_OK THEN
    return response
  ELSE
    return error
END
}

```

Gambar 4.13 Pseudocode Kelas Connector

Pseudocode diatas adalah fungsi-fungsi yang ada di kelas *Connector.java* , ketika fungsi *send()* dijalankan maka *send()* akan memanggil fungsi *connect()* untuk mendapatkan koneksi. Jika koneksi ada, maka *request* akan dikirimkan dan *send()* akan mengembalikan

4.3.1 Implementasi Fitur Pemesanan Kamar Hotel

Implementasi fitur pemesanan Hotel Cerdas memerlukan beberapa kelas yang berfungsi untuk menampilkan daftar hotel, menampilkan detail dan form pemesanan hotel dan detail

pemesanan hotel. Implementasi pada fitur pemesanan ini akan ditampilkan menggunakan *pseudocode* sebagai berikut.

```
Mendapatkan data hotel
Memilih hotel
Memasukkan tanggal dan durasi pada form pemesanan

IF tanggal kosong THEN
    Menampilkan peringatan tanggal tidak boleh kosong
ELSE
    Masuk ke detail pemesanan
    Memilih nomor kamar yang tersedia
    Melanjutkan ke pembayaran
END
```

Gambar 4.14 *Pseudocode* Melakukan Pemesanan

Ketika pengguna ingin melakukan pemesanan dengan menekan salah satu daftar pada hotel, pengguna akan masuk ke form pemesanan untuk mengisi tanggal *check-in* dan durasi. Jika tanggal tidak diisi, maka aplikasi akan menampilkan form peringatan untuk mengisi tanggal. Setelah melengkapi form pemesanan, pengguna melanjutkan ke detail pemesanan dan memilih nomor kamar yang diinginkan. Kode sumber lengkap akan dilampirkan pada lampiran..

4.3.2 Implementasi Pembayaran Online Menggunakan Midtrans

Pada bagian ini adalah implementasi dari pembayaran online menggunakan sebuah SDK (*Software Development Kit*) Android yang bernama Midtrans. Untuk pembayaran online pada aplikasi ini akan menggunakan Go-Pay. Implementasi pada fitur pembayaran online ini akan ditampilkan dengan nama fungsi bayar() menggunakan *pseudocode* yang akan dijalankan saat melakukan ke pembayaran. Implementasi *pseudocode* adalah sebagai berikut.

```

function bayar(){
    Memilih metode pembayaran,
        contohnya menggunakan gopay

    Menyelesaikan pembayaran
    IF pembayaran sukses THEN
        Menyimpan transaksi pada database
        Menampilkan peringatan
        pembayaran sukses
    ELSE
        Menampilkan peringatan
        pembayaran gagal
    END
}

```

Gambar 4.15 *Pseudocode* Melakukan Pembayaran *Online*

Pseudocode fungsi bayar() dijalankan saat pengguna menekan tombol bayar, pengguna akan diberikan pilihan metode pembayaran, contohnya adalah *Go-Pay*. Jika pembayaran berhasil, maka transaksi akan disimpan pada *database*.

4.3.3 Implementasi *Monitoring* dan *Controlling* Kamar Hotel

Implementasi fitur *monitoring* dan *controlling* kamar pada Sistem Hotel Cerdas memerlukan beberapa *class* yang berfungsi untuk menampilkan daftar kamar yang dipesan, *check-in* dengan melakukan *scan QR Code* dan menampilkan halaman untuk mengontrol perangkat pada kamar hotel.

4.3.3.1 Implementasi *Check-in* dengan *Scan QR Code*

Bagian ini adalah implementasi melakukan *check-in* dengan *scan QR Code*. Pengguna dapat melakukan *check-in* setelah menekan salah satu dari daftar kamar untuk melakukan *check-in* dan masuk ke dalam halaman *monitoring* dan *controlling*

kamar. *Check-in* hanya dapat dilakukan ketika tanggal pada hari ini sudah sesuai dengan tanggal *check-in*. Implementasi pada fitur ini akan ditampilkan menggunakan *pseudocode* sebagai berikut..

```

IF belum check-in THEN
    Menghidupkan QR Code Scanner
    Scan QR
    IF QR Code sesuai THEN
        Check-in berhasil
ELSE
    Masuk ke halaman room control
END

```

Gambar 4.16 *Pseudocode Check-in dengan QR Code*

Ketika pengguna ingin masuk ke halaman *room control*, pengguna akan diminta untuk melakukan *check-in*. Jika belum melakukan *check-in*, kamera akan aktif dan dapat digunakan untuk melakukan *scan QR Code*. Jika *QR Code* cocok, maka *check-in* berhasil dan langsung masuk ke halaman *room control*. Kode sumber lengkap akan dilampirkan pada lampiran.

4.3.3.2 Implementasi *Controlling Perangkat Kamar*

Bagian ini adalah implementasi mengontrol perangkat yang terdapat pada kamar pada yang digunakan oleh pengguna pada halaman *room control* seperti pintu, lampu, dan kipas. Implementasi pada fitur ini akan ditampilkan menggunakan *pseudocode* sebagai berikut.

```

IF pengguna menghidupkan perangkat THEN
    Status perangkat menjadi hidup/buka
ELSE IF pengguna mematikan perangkat THEN
    Status Perangkat menjadi hidup/tutup
    Mengirim status ke database
END

```

Gambar 4.17 *Pseudocode Mengontrol Kamar*

Jika pengguna menghidupkan atau membuka sebuah perangkat, maka status perangkat tersebut menjadi hidup atau buka. Sebaliknya, jika mematikan perangkat tersebut, maka status perangkat menjadi mati atau tutup. Setelah itu status perangkat akan dikirimkan ke *database*. Kode sumber lengkap dilampirkan pada lampiran.

4.3.3.3 Implementasi Fitur *Room Sharing*

Bagian ini adalah implementasi *Room Sharing* atau pembagian akses kamar. Pada halaman *Room Control* yang dimiliki pemilik kamar akan terdapat form input *username* pengguna yang akan dibagikan akses kamar. *Username* harus sudah memiliki akun pada Aplikasi Smart Hotel. Implementasi pada fitur ini akan ditampilkan menggunakan *pseudocode* sebagai berikut.

Pengguna mengisi form room sharing dengan username milik pengguna lain untuk dibagikan akses kamar

```

IF username tidak ada THEN
    Menampilkan pemberitahuan pengguna tidak ada
ELSE IF pengguna sudah pernah ditambahkan THEN
    Menampilkan pemberitahuan pengguna sudah ada
ELSE
    Menambahkan pengguna
END

```

Gambar 4.18 *Pseudocode* Implementasi *Room Sharing*

Ketika pengguna ingin memberikan akses kamar kepada pengguna lainnya, pengguna mengisi form dengan *username* pengguna yang ingin ditambahkan. Jika *username* ada dan belum pernah ditambahkan, maka pembagian akses kamar berhasil. *Shared user* akan dapat mengakses kamar tersebut dengan masuk ke halaman daftar *shared room*. Dan akan masuk ke halaman

room control ketika memilih kamar dengan status sebagai *shared user*. Kode sumber lengkap dilampirkan pada lampiran.

4.3.3.4 Implementasi *Check-out*

Bagian ini adalah implementasi *check-out* dari kamar hotel. *Check-out* dapat dilakukan secara otomatis maupun manual. Implementasi *check-out* akan ditampilkan menggunakan *pseudocode* sebagai berikut.

```

day = tanggal hari ini

WHILE day <= tanggal check-out
    IF pengguna melakukan checkout manual THEN
        Check-out berhasil
        Break
    ENDWHILE

Check-out otomatis
Menunggu konfirmasi check-out dari admin atau petugas hotel
Mengirimkan notifikasi pada smartphone pengguna

END

```

Gambar 4.19 *Pseudocode* Implementasi *Check-out*

Check-out otomatis akan terjadi ketika masa tinggal pengguna sudah habis. Pengguna juga dapat langsung melakukan *check-out* secara manual dengan menekan tombol *check-out*. Pengguna akan mendapatkan notifikasi bahwa status telah berubah menjadi *check-out*. Setelah *check-out*, pengguna tidak dapat mengakses perangkat lain pada kamar, selain pintu. Akses *Room Control* akan tetap diberikan sampai *check-out* sudah di konfirmasi oleh petugas hotel. Kode sumber lengkap dilampirkan pada lampiran.

4.3.3.5 Implementasi Akses Lift

Bagian ini adalah implementasi untuk mengaktifkan lift dengan *scan QR Code* pada lift. Implementasi akan ditampilkan menggunakan *pseudocode* sebagai berikut.

```

IF pengguna mengaktifkan lift THEN
    Mengaktifkan QR Code scanner
    Pengguna melakukan scan
    IF QR Code cocok THEN
        Lift aktif
    END IF
END
  
```

Gambar 4.20 *Pseudocode* Implementasi Akses Lift

Saat pengguna ingin mengaktifkan lift pada halaman *room control*, *QR Code scanner* akan aktif dan pengguna dapat melakukan *scan* pada *QR Code*. Apabila *QR Code* sesuai, lift akan aktif. Kode sumber lengkap akan dilampirkan pada lampiran.

4.4 Codeigniter

Pada implementasi sistem *monitoring* dan *controlling* hotel untuk admin atau petugas hotel, serta *web service* untuk Arduino dan Aplikasi Hotel Cerdas, penulis menggunakan *framework* CodeIgniter 3 dalam pembuatannya. Perlu dilakukan konfigurasi *database* agar *framework* dapat terhubung dengan *MySQL* seperti pada Gambar 4.21 Konfigurasi *Database*.

```

$db['default'] = array(
    'dsn' => '',
    'hostname' => 'localhost',
    'username' => 'root',
    'password' => '',
    'database' => 'db_hotel',
  );
  
```

Gambar 4.21 Konfigurasi *Database*

Konfigurasi *database* berada pada folder `application/config/database.php`, *database* pada CodeIgniter 3 dihubungkan dengan tabel yang sama dengan yang di akses oleh Arduino dan Aplikasi Hotel Cerdas agar nantinya *framework* dapat menampilkan data yang telah di olah.

Pada implementasi sistem monitor ini penulis menggunakan lima *controller* dan satu *model*. *Controller* tersebut akan memiliki fungsi yang terbagi menjadi dua, yaitu:

1. Sebagai Web Service
2. Sebagai Sistem Monitoring dan Controlling Hotel

4.4.1 Implementasi *Web Service*

Sebagai web service, controller akan menangani semua *HTTP request* baik dari Aplikasi Hotel Cerdas, maupun dari Arduino untuk menambah, mengubah, melihat, dan menghapus data pada *database*. Setelah menjalankan *request*, server akan mengirimkan *response*. Pada implementasi *web service* ini penulis menggunakan dua *controller*. *Controller* yang digunakan untuk implementasi *web service* adalah sebagai berikut:

1. **Controller Login.php**

Controller login ini digunakan untuk melakukan login dan register untuk Aplikasi Hotel Cerdas.

2. **Controller Service.php**

Controller service ini digunakan berisi *query* yang dijalankan untuk melakukan interaksi antara *framework* dengan *database* untuk melakukan pengelolaan data dari *request* yang diminta oleh Aplikasi Hotel Cerdas dan *request* dari Arduino.

4.4.1.1 Implementasi Web Service Pemesanan Kamar Hotel

Untuk melakukan implementasi fitur pemesanan kamar hotel pada Aplikasi Hotel Cerdas, diperlukan beberapa fungsi

pada *controller* Service.php yang dibutuhkan untuk menjalankan *query* ketika mendapatkan *request* dari Aplikasi Hotel Cerdas. Pada implementasi Web Service ini, semua *request* akan menggunakan *method* POST. Implementasi bagian ini akan ditampilkan menggunakan *pseudocode* sebagai berikut.

```
function getHotel(){  
    Select tabel ht_hotel  
  
    FOR setiap baris pada tabel  
        Memasukkan baris tabel pada sebuah array  
    ENDFOR  
  
    Memasukkan array pada objek "listhotel"  
    Mengembalikan response dengan format JSON  
END  
}
```

Gambar 4.22 *Pseudocode* Mendapatkan Daftar Hotel

Pseudocode di atas merupakan fungsi untuk mendapatkan daftar hotel dari tabel ht_hotel yang ada di *database*. Hasil dari *query* akan diformat dalam format JSON (JavaScript Object Notation) yang kemudian dikirimkan kembali kepada Aplikasi Hotel Cerdas sebagai *response*. Kode sumber lengkap dilampirkan pada lampiran.

```

function getAvailableRoom(){
  Select tabel ht_kamar yang tidak digunakan
  pada tanggal check-in sampai tanggal
  check-out pengguna pada hotel yang dipilih

  FOR setiap baris pada tabel
    Masukkan data pada sebuah array
  ENDFOR

  Memasukkan array pada objek “kamar tersedia”
  Mengembalikan response dengan format JSON
END
}

```

Gambar 4.23 *Pseudocode* Mendapatkan Kamar yang Tersedia

Pseudocode di atas adalah fungsi untuk mendapatkan kamar yang tersedia pada tanggal yang dimasukkan oleh pengguna Aplikasi Hotel Cerdas. Hasil dari *query* akan dikirimkan kembali sebagai *response* dengan format JSON. Kode sumber lengkap dilampirkan pada lampiran.

```

Function rentRoom(){
  Mendapatkan data transaksi yang akan ditambahkan

  Memasukkan data transaksi pada ht_transaksi
END
}

```

Gambar 4.24 *Pseudocode* Menambahkan Transaksi

Pseudocode di atas adalah fungsi untuk menyimpan data pemesanan kamar hotel ke tabel ht_transaksi pada *database* setelah pengguna Aplikasi Hotel Cerdas menyelesaikan pembayaran. Kode sumber lengkap dilampirkan pada lampiran.

4.4.1.2 Implementasi Web Service *Monitoring dan Controlling Kamar*

Untuk melakukan implementasi fitur *monitoring* dan *controlling* kamar hotel pada Aplikasi Hotel Cerdas, diperlukan beberapa fungsi pada *controller* Service.php yang dibutuhkan untuk menjalankan *query* ketika mendapatkan *request* dari Aplikasi Hotel Cerdas. Implementasi bagian ini akan ditampilkan menggunakan *pseudocode* sebagai berikut.

```
function getMyroom(){  
  
    Select tabel ht_kamar yang dipesan pengguna  
  
    FOR setiap baris pada tabel  
        Memasukkan baris tabel pada sebuah array  
    ENDFOR  
  
    Memasukkan array pada objek “myroom”  
    Mengembalikan response dengan format JSON  
END  
}
```

Gambar 4.25 *Pseudocode* Fungsi Mendapatkan Daftar Kamar

Pseudocode di atas adalah fungsi untuk mendapatkan kamar yang sudah dipesan. Kamar yang akan ditampilkan adalah yang masa tinggalnya belum habis. Untuk kamar yang sudah bisa *check-in* akan memberikan status N atau “Not Available” dan memberikan status X atau “Available” untuk yang sudah bisa *check-in*. Kode sumber lengkap dilampirkan pada lampiran.

```

function checkin(){
    Mendapatkan ID transaksi

    Mengubah status pada tabel ht_transaksi menjadi
    1 atau check-in pada ID transaksi
END
}

```

Gambar 4.26 *Pseudocode* Mengubah Status Check-in

Pseudocode di atas adalah fungsi untuk mengubah *m_status* dan tanggal *check-in* pada *ht_transaksi* ketika berhasil melakukan *check-in*. Kode sumber lengkap dilampirkan pada lampiran.

```

function roomcond(){
    Mendapatkan ID kamar

    Mendapatkan status pintu, lampu, kipas pada
    ht_kamar pada ID kamar

    Mengembalikan response status perangkat kamar
END
}

```

Gambar 4.27 *Pseudocode* Mendapatkan Kondisi Kamar

Pseudocode di atas memiliki dua fungsi untuk mendapatkan kondisi perangkat kamar, serta mendapatkan daftar pengguna yang dibagikan akses kamar. Kedua fungsi tersebut akan dijalankan ketika pengguna memasuki halaman *Room Control*. Kode sumber lengkap dilampirkan pada lampiran. Kode sumber lengkap dilampirkan pada lampiran.

```

function setPerangkatKamar(){
  IF perintah== buka THEN
    Status perangkat menjadi 1
  ELSE IF perintah== tutup THEN
    Status perangkat menjadi 0
  END
}

function setLift(){
  IF lift sedang tidak aktif THEN
    Hidupkan lift dengan durasi 60 detik
  END
}

```

Gambar 4.28 *Pseudocode* Fungsi *Controlling* Perangkat

Pseudocode diatas merupakan fungsi-fungsi yang akan dijalankan ketika pengguna melakukan *controlling* pintu, lampu, kipas, dan lift pada halaman *Room Control*. Kode sumber lengkap dilampirkan pada lampiran.

```

function addShare(){
  Mendapatkan ID transaksi dan username pengguna
  yang ingin ditambahkan
  IF pengguna ada dan belum pernah ditambahkan THEN
    Menambahkan pengguna pada tabel ht_share
  END
}

function deleteShare(){
  Mendapatkan ID pengguna
  Menghapus pengguna pada tabel ht_share
  END
}

```

Gambar 4.29 *Pseudocode* Tambah atau Hapus *Shared User*

Pseudocode di atas merupakan fungsi-fungsi yang digunakan untuk menambahkan dan menghapus akses kamar

pada *shared user*. Kode sumber lengkap dilampirkan pada lampiran.

```
function checkout(){
    Mendapatkan ID transaksi
    Update status pada ht_transaksi menjadi 2 atau
    check-out
END
}
```

Gambar 4.30 *Pseudocode* Fungsi Checkout

Pseudocode di atas berisi fungsi *checkout*. Fungsi ini akan dijalankan ketika pengguna melakukan *check-out*.

4.4.1.3 Implementasi Web Service Menampilkan Daftar *Shared Room*

Untuk melakukan implementasi menampilkan daftar *shared room* untuk *shared user* pada Aplikasi Hotel Cerdas, diperlukan fungsi *getSharedroom* pada *controller* Service.php yang dibutuhkan untuk menjalankan *query* ketika mendapatkan *request* dari Aplikasi Hotel Cerdas.

```
function getSharedRoom(){
    Mendapatkan ID user
    Select ht_kamar yang ada pada idtransaksi yang dimiliki user
    pada ht_share.

    FOR setiap baris pada tabel
        Memasukkan baris tabel pada sebuah array
    ENDFOR
    Memasukkan array pada objek "myroom"
    Mengembalikan response dengan format JSON
END
}
```

Gambar 4.31 *Pseudocode* Fungsi Mendapatkan Shared Room

Pseudocode di atas merupakan fungsi yang digunakan untuk mendapatkan daftar kamar yang dibagikan akses nya kepada pengguna. Kode sumber lengkap dilampirkan pada lampiran.

4.4.1.4 Implementasi Web Service Arduino

Untuk menjalankan *request* yang diminta Arduino dibuatkan sebuah fungsi yang akan digunakan untuk mendapatkan status perangkat kamar dan status lift hotel. Yang nantinya akan digunakan oleh *microcontroller* untuk menentukan apakah perangkat akan dihidupkan atau dimatikan.

```
function geStsArd(){
    Mendapatkan ID device
    Select tabel ht_kamar yang memiliki device tersebut
    Mendapatkan status perangkat
    Mengembalikan response dengan status perangkat
END
}
```

Gambar 4.32 *Pseudocode* Mendapatkan Status Perangkat

Pseudocode diatas adalah fungsi untuk yang akan digunakan untuk mendapatkan status perangkat kamar dan status lift hotel. Kode sumber lengkap akan dilampirkan pada lampiran.

4.4.2 Implementasi Website Sistem *Monitoring* dan *Controlling* Hotel

Website Sistem *Monitoring* dan *Controlling* Hotel akan diimplementasikan sesuai Use Case Diagram yang telah dibuat pada bab perancangan, untuk itu dibuatkan controller untuk menampilkan website *monitoring* dan *controlling* yang akan digunakan admin hotel. Pada implementasi sistem ini penulis

menggunakan lima *controller* dan satu *model*. *Controller* dan *model* yang akan dibuat adalah sebagai berikut:

1. Controller Login.php

Controller login ini digunakan untuk melakukan login untuk website *monitoring* dan *controlling* untuk admin.

2. Controller Hotel.php

Controller hotel ini digunakan untuk menampilkan halaman *monitoring* dan *controlling* kamar yang sedang digunakan hari ini, akses lift, dan menampilkan *QR Code* yang akan digunakan untuk *check-in*.

3. Controller Myroom.php

Controller myroom ini digunakan untuk menampilkan halaman *room control*.

4. Controller Transaksi.php

Controller transaksi ini digunakan untuk menampilkan halaman riwayat transaksi.

5. Controller Service.php

Controller service ini digunakan berisi *query* yang dijalankan untuk melakukan interaksi antara *framework* dengan *database* untuk mengelola data dari *request* yang diminta oleh Sistem *Monitoring* dan *Controlling* Hotel.

6. Model M_CallSQL.php

Model *M_CallSQL* berisi beberapa fungsi yang akan dijalankan pada *controller*.

4.4.2.1 Use Case Admin Melihat Kamar Hotel

Use Case ini diimplementasikan menggunakan *controller* Hotel.php. Pada kasus ini admin dapat melihat daftar kamar yang sedang digunakan hari ini pada halaman *dashboard* dan melihat seluruh kamar dalam hotel yang dikelola admin tersebut dalam

bentuk tabel. Implementasi akan ditampilkan menggunakan *pseudocode*.

```

function getKamarToday(){
    Mendapatkan kamar yang sedang digunakan
    hari ini pada database
    FOR setiap baris pada tabel
        Menampilkan kamar
    ENDFOR
END
}

Function getKamarAll(){
    Mendapatkan semua kamar milik hotel yang
    sedang dikelola admin pada database
    FOR setiap baris pada tabel
        Menampilkan baris
    END FOR
END
}

```

Gambar 4.33 *Pseudocode* Fungsi Menampilkan Data Kamar

Pseudocode diatas adalah fungsi-fungsi yang digunakan untuk menampilkan kamar. Fungsi `getKamarToday()` digunakan untuk menampilkan daftar kamar yang sedang digunakan, sedangkan fungsi `getKamarAll()` menampilkan seluruh kamar milik hotel yang dikelola admin. Kode sumber lengkap dilampirkan pada lampiran

4.4.2.2 Use Case Admin Mengelola Perangkat Hotel

Use Case ini diimplementasikan menggunakan *controller* `Myroom.php` untuk menampilkan halaman yang menampilkan daftar kamar yang berada pada hotel dalam bentuk tabel, menampilkan halaman untuk mengontrol perangkat pada kamar dan halaman untuk mengontrol lift pada hotel. Kemudian menggunakan *controller* `Service.php` yang akan menjalankan

query untuk mengubah status perangkat pada *database*. Implementasi akan ditampilkan menggunakan *pseudocode*.

```

IF admin memilih tombol buka atau hidupkan dan tutup
atau matikan pada daftar perangkat THEN
    Menjalankan fungsi setPerangkat() pada
    Gambar 4.28 Pseudocode Fungsi Controlling
    Perangkat
END
}

```

Gambar 4.34 *Pseudocode* Mengontrol Kamar

Pseudocode di atas adalah ketika admin masuk pada halaman Room Control dan melakukan controlling perangkat kamar. Jika admin memilih tombol untuk mengaktifkan perangkat, maka akan menjalankan fungsi `setPerangkat()` pada Gambar 4.28 *Pseudocode* Fungsi Controlling Perangkat. Kode sumber lengkap akan dilampirkan dalam lampiran.

Fungsi untuk mengontrol *lift* hotel akan menggunakan *controller* `Service.php` dan memiliki proses yang sama dengan Gambar 4.34 *Pseudocode* Mengontrol Kamar. Untuk melakukan *controlling lift* admin harus masuk ke halaman Akses *Lift* pada website, setelah itu akan muncul halaman *Lift Control*. Admin dapat mengaktifkan *lift* pada hotel melalui tombol yang tersedia pada halaman tersebut. Implementasi fungsi akan ditampilkan menggunakan *pseudocode*.

```

IF admin memilih tombol untuk mengaktifkan lift THEN
    Menjalankan fungsi setLift() pada
    Gambar 4.28 Pseudocode Fungsi Controlling
    Perangkat
END

```

Gambar 4.35 *Pseudocode* Mengontrol Lift

Pseudocode di atas adalah ketika admin masuk pada halaman *Lift Control* dan melakukan *controlling lift*. Jika admin memilih tombol untuk mengaktifkan perangkat, maka akan menjalankan fungsi *setLift()* pada Gambar 4.38 *Pseudocode Fungsi Controlling Perangkat*. Kode sumber lengkap akan dilampirkan dalam lampiran.

4.4.2.3 Use Case Admin Melakukan *Check-in* dan *Check-out* Manual

Use Case ini diimplementasikan menggunakan *controller Service.php*. Pada kasus ini admin dapat melihat daftar kamar yang sedang digunakan hari ini dalam bentuk tabel, kemudian admin dapat menekan tombol “CHECK IN” atau “CHECK OUT” pada kolom aksi yang terdapat pada tabel. Ketika admin menekan tombol “CHECK IN”, maka akan menjalankan fungsi *checkin()* seperti pada Gambar 4.26 *Pseudocode Mengubah Status Check-in*. Kemudian jika admin menekan tombol “CHECK OUT”, maka akan menjalankan fungsi *checkout()*.

4.4.2.4 Use Case Admin Mengonfirmasi *Check-out*

Use Case ini diimplementasikan menggunakan *controller Service.php*. Pada kasus ini admin dapat melihat daftar kamar yang sedang digunakan hari ini dalam bentuk tabel, kemudian admin dapat menekan tombol konfirmasi *check-out* pada kolom aksi yang terdapat pada tabel. Implementasi fungsi akan ditampilkan menggunakan *pseudocode*.

```
function confirmchout(){
    Mendapatkan ID transaksi
    Menghapus transaksi pada tabel ht_transaksi
    dimana database akan menjalankan trigger
    update_log_before_delete untuk memindahkan
    transaksi yang dihapus ke tabel ht_log_transaksi
END
}
```

Gambar 4.36 *Pseudocode* Fungsi Konfirmasi *Check-out*

Pseudocode di atas adalah fungsi yang akan dijalankan ketika admin melakukan konfirmasi *check-out*. Ketika admin melakukan konfirmasi, fungsi tersebut akan dijalankan untuk menghapus data transaksi tersebut. *Trigger* pada *database* akan aktif untuk menyimpan data sebelum dihapus pada *ht_log_transaksi*. Tabel *ht_log_transaksi* ini akan digunakan untuk melihat riwayat transaksi. Kode sumber lengkap akan dilampirkan pada lampiran

4.4.2.5 Use Case Admin Melihat Riwayat Transaksi

Use Case ini diimplementasikan menggunakan *controller* *Transaksi.php* untuk menampilkan riwayat transaksi. Untuk melihat riwayat transaksi admin dapat memilih menu “Riwayat Transaksi” pada navigasi website. Implementasi ini akan ditampilkan menggunakan *pseudocode*.

```
function getRiwayat(){
    Select tabel ht_log_transaksi
    FOR setiap baris pada tabel
        Menampilkan baris
    END FOR
END
}
```

Gambar 4.37 Pseudocode Fungsi Menampilkan Riwayat Transaksi

4.4.2.6 Use Case Admin Melihat *QR Code* untuk *Check-in*

Use Case ini diimplementasikan menggunakan *controller* *Hotel.php* untuk menampilkan *QR Code* yang akan digunakan oleh pengunjung hotel untuk melakukan *check-in*. Untuk menampilkan *QR Code*, admin dapat memilih menu “QR CODE” pada menu navigasi. Implementasi akan ditampilkan menggunakan *pseudocode*.

```
function showQR(){  
    Mendapatkan ID hotel yang sedang dikelola admin  
    Select kolom h_qrstring pada tabel ht_hotel  
    FOR setiap baris pada tabel  
        Membuat QR Code berdasarkan  
        isi dari kolom tersebut  
        Menyimpan QR Code dalam bentuk  
        foto pada folder  
    ENDFOR  
  
    Menampilkan QR Code pada halaman “QR CODE”  
END  
}
```

Gambar 4.38 Pseudocode Fungsi Menampilkan QR Code

Pseudocode diatas adalah fungsi yang digunakan untuk membuat *QR Code* dari isi yang di dapatkan dari kolom *h_qrstring* tabel *ht_hotel*. Kemudian, hasilnya akan disimpan pada sebuah folder yang nantinya akan ditampilkan pada halaman web. Kode sumber lengkap akan dilampirkan pada lampiran.

[Halaman ini sengaja dikosongkan]

BAB V UJICOBA DAN EVALUASI

Pada bab ini dijelaskan tentang uji coba dan analisis dari implementasi yang telah dilakukan pada Tugas Akhir ini.

5.1 Lingkup Uji Coba

Pada lingkup uji coba, penulis menggunakan konektivitas jaringan rumah “*Home Network*” untuk melakukan uji coba Sistem Hotel Cerdas dijalankan pada *smartphone* Android dengan spesifikasi sebagai berikut:

Tabel 5.1 Lingkup Spesifikasi *Smartphone* Android

Komponen	Spesifikasi
CPU	1.8GHz Snapdragon 450 Octa-core
Sistem Operasi	Funtouch OS_4.0
Memori	4GB RAM
Android Version	Oreo/8.1.0
API level/NDK release	API level 27
Penyimpanan	1 TB

Untuk melakukan uji coba Sistem *Monitoring* dan *Controlling* Hotel dijalankan pada laptop dengan spesifikasi sebagai berikut:

Tabel 5.2 Lingkup Spesifikasi Laptop

Komponen	Spesifikasi
CPU	Intel(R) Core(TM) i7-8750H CPU @ 2.20GHz (12 CPUs), ~2.2GHz
Sistem Operasi	Windows 10 Home Single Language 64-bit (10.0, Build 17763)

Memori	16384MB RAM
Penyimpanan	1 TB
Browser	Chrome Version 83.0.4103.97 (Official Build) (64-bit)

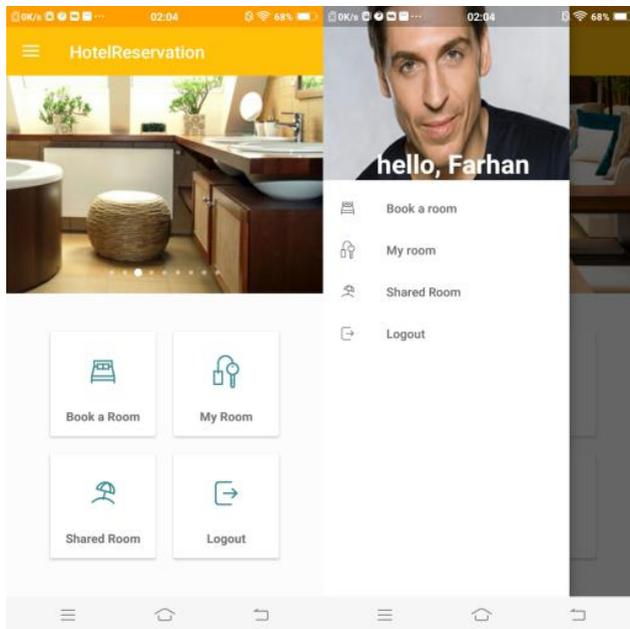
5.2 Skenario Pengujian

Pada bagian ini akan dibahas mengenai proses uji coba yang digunakan. Pada proses uji coba, pengujian dilakukan dengan menjalankan serangkaian perintah terhadap sistem dimana pengujian dilakukan pada prototipe yang telah dibuat. Kasus pengujian ini berkorelasi dengan *flowchart* dan kasus-kasus penggunaan yang sebelumnya sudah dirancang dan dijelaskan pada Bab III. Pengujian akan dibagi menjadi tiga kategori, yaitu:

1. Pengujian Aplikasi Hotel Cerdas yang hasilnya dapat dilihat dalam bentuk monitoring melalui website *Monitoring* dan *Controlling*.
2. Pengujian Website *Monitoring* dan *Controlling* untuk admin akan dapat dilihat langsung hasilnya pada Aplikasi Hotel Cerdas.
3. Pengujian alat pada miniatur dari hasil *controlling* melalui Website *Monitoring* dan *Controlling* atau Aplikasi Hotel Cerdas.

5.2.1 Aplikasi Hotel Cerdas

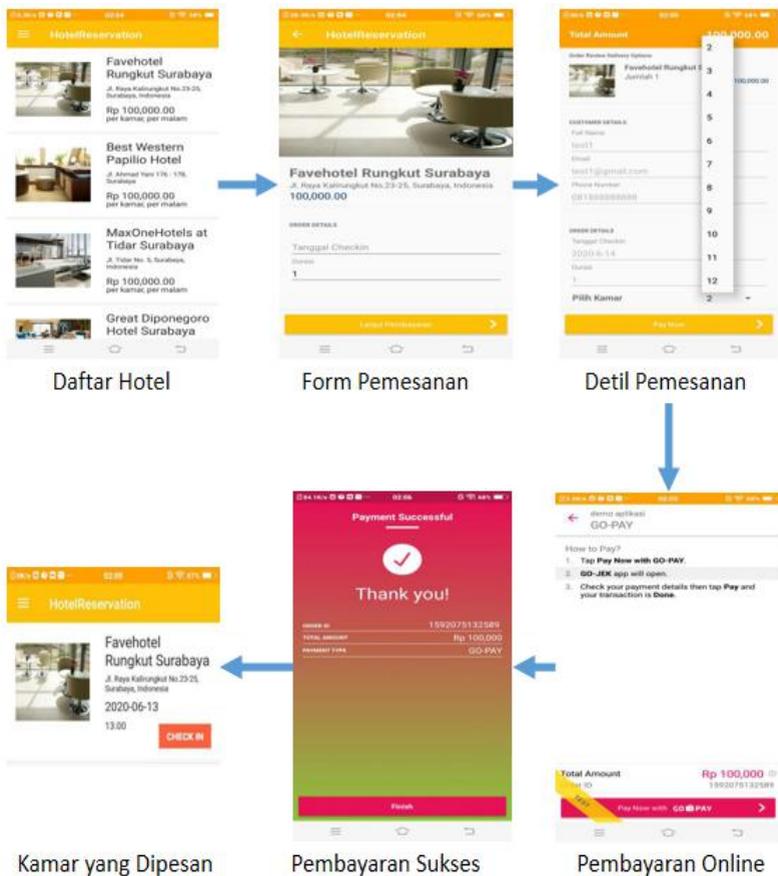
Pada bagian ini akan dilakukan pengujian untuk Aplikasi Hotel Cerdas seperti *flowchart* yang telah jelaskan pada perancangan sistem.



Gambar 5.1 Halaman Utama dan Navigasi

5.2.1.1 Skenario Pengujian Pemesanan Hotel

Pada skenario pengujian ini dilakukan untuk menguji apakah pengguna dapat melakukan pemesanan hotel. Untuk melakukan pengujian ini pengguna masuk pada menu “Book a Room”.



Gambar 5.2 Skenario Pengujian Pemesanan Hotel

Pada gambar di atas pengguna dapat memilih hotel yang akan dipesan, dan mengisi form pemesanan berupa tanggal dan durasi menginap. Setelah mengisi form pengguna dapat melanjutkan ke detail pemesanan dengan menekan tombol “Lanjut Pembayaran”, lalu memilih nomor kamar yang tersedia.

Kamar yang dipesan akan muncul pada menu “My room” pada navigasi setelah berhasil melakukan pembayaran.

Transaksi Data Transaksi Home > Transaksi

Data Transaksi Tambah Transaksi

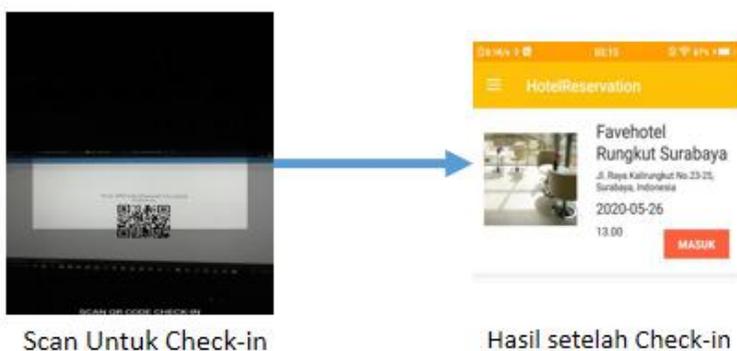
Show entries Search:

Transaksi Id	Nomor Kamar	Nama User	Tanggal Pesan	Check In	Check Out
6	2	test1	2020-06-14 02:06:46	2020-06-14 00:00:00	2020-06-15 00:00:00

Gambar 5.3 Hasil Pada Riwayat Transaksi

5.2.1.2 Skenario Pengujian *Check-in* Kamar Hotel

Pada skenario pengujian ini dilakukan untuk menguji apakah pengguna dapat melakukan *check-in* kamar dengan *scan QR Code*. Untuk melakukan pengujian ini pengguna masuk pada menu “My room”. Dan menekan tombol “CHECK IN” pada daftar kamar yang telah dipesan seperti pada Gambar 5.2 untuk memulai *scan QR Code*.

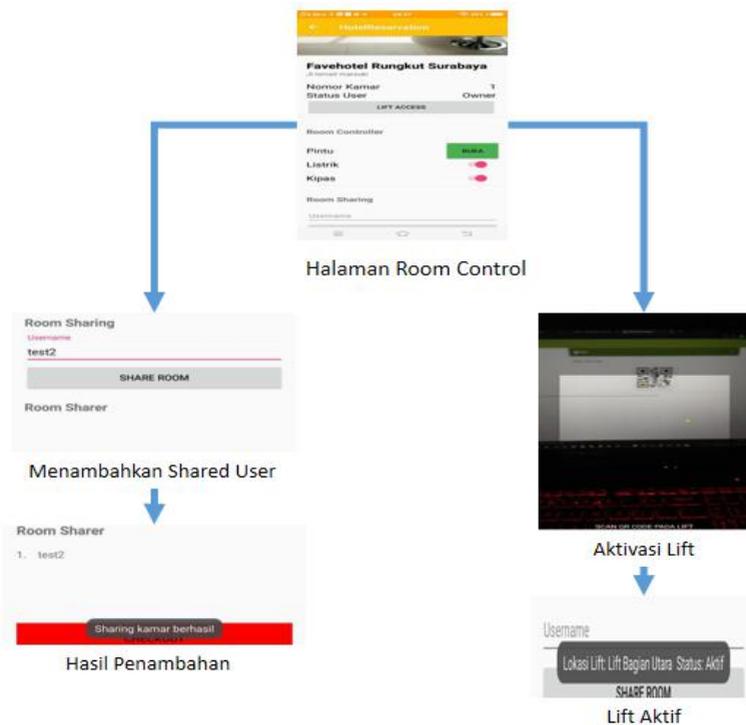


Gambar 5.3 Skenario Pengujian Check-in

Apabila *scan* berhasil dan *QR Code* cocok, maka *check-in* berhasil dan tombol “CHECK IN” akan berubah menjadi “MASUK”.

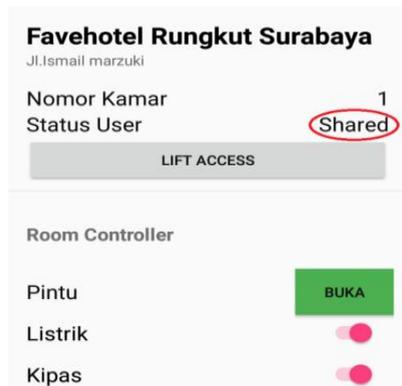
5.2.1.3 Skenario Pengujian Mengontrol Perangkat Kamar

Pada skenario pengujian ini dilakukan untuk menguji apakah pengguna dapat mengontrol perangkat kamar. Untuk melakukan pengujian ini pengguna dapat menekan tombol “MASUK” pada alur halaman daftar kamar setelah melakukan *check-in*.



Gambar 5.4 Skenario Pengujian Mengontrol Perangkat Kamar

Gambar di atas adalah Skenario Pengujian Mengontrol Perangkat Kamar dimana perangkat yang terdiri dari pintu, lampu, dan kipas. Pengguna dapat melakukan *controlling* dengan menekan tombol atau *switch* yang terdapat pada halaman ini. Pada bagian ini juga terdapat tombol “LIFT ACCESS” yang dapat digunakan untuk mengaktifkan *lift* dengan melakukan *scan QR Code*. Kemudian, pada halaman ini juga terdapat fitur *Room Sharing* yang berfungsi untuk membagikan akses kamar. Untuk pengujiannya penulis akan mencoba membagikan akses kepada pengguna dengan *username* "test2". Setelah dibagikan akses, pengguna “test2” dapat mengakses kamar ini dengan masuk ke menu “Shared Room” yang berisi daftar kamar yang aksesnya dibagikan kepada pengguna “test2”.



Gambar 5.5 Halaman *Room Control* untuk *Shared User*

Tampilan halaman untuk mengontrol perangkat bagi *shared user* akan sama, namun *shared user* tidak mendapatkan fitur untuk membagikan akses kamar dan tidak dapat melakukan *check-out*.

5.2.1.4 Skenario Pengujian *Check-out* Kamar Hotel

Pada skenario pengujian ini dilakukan untuk menguji apakah pengguna dapat melakukan *check-out*. Untuk melakukan pengujian ini pengguna dapat menekan tombol “CHECK OUT” pada halaman *room control*. Jika tidak melakukan *check-out* secara manual, maka *check-out* akan terjadi secara otomatis setelah masa tinggal habis dan aplikasi akan mengirimkan notifikasi untuk segera keluar dari kamar hotel.



Gambar 5.6 Skenario Pengujian *Check-out*

5.2.2 Sistem *Administrator Monitoring dan Controlling Hotel*

Pada bagian ini akan dilakukan pengujian untuk Sistem *Administrator Monitoring dan Controlling Hotel* yang akan digunakan oleh admin atau petugas hotel. Pengujian akan dilakukan sesuai kasus penggunaan yang telah jelaskan pada perancangan Sistem *Monitoring dan Controlling Hotel*.

5.2.2.1 Skenario Pengujian Melihat Kamar Hotel

Pada skenario pengujian ini dilakukan untuk menguji apakah admin dapat melihat daftar kamar yang sedang digunakan hari ini dan daftar seluruh kamar pada hotel.

Monitor Hotel 1

Dashboard

Data Kamar Hotel

Show 10 entries

ID Transaksi	Nomor Kamar	DeviceId	Check In	Check Out	Status	Action
2	1	Room Control	2020-06-14 04:36:00	2020-06-27 13:00:00	BELUM CHECK IN	CHECK IN

Showing 1 to 1 of 1 entries

Gambar 5.7 Halaman Dashboard

Kamar Data Kamar Hotel

Show 10 entries

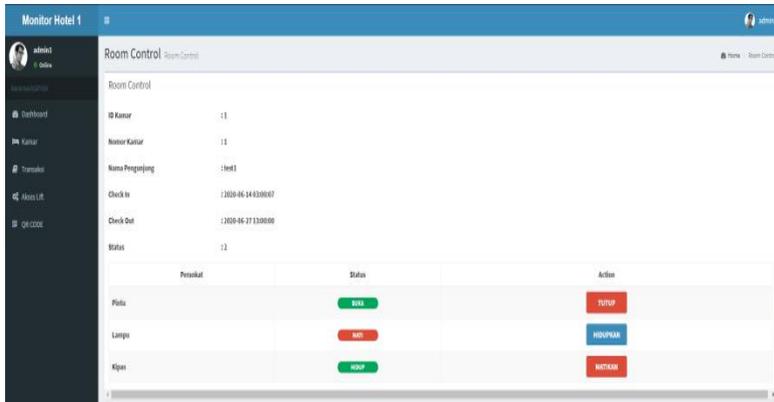
Id	Nomor Kamar	DeviceId	Check In	Check Out	Status	Action
1	1	Room Control	2020-06-14 04:36:00	2020-06-27 13:00:00	BELUM CHECK IN	CHECK IN
2	2	Room Control			Kamar Tersedia	
3	3	Room Control			Kamar Tersedia	
4	4	Room Control			Kamar Tersedia	
5	5	Room Control			Kamar Tersedia	
6	6	Room Control			Kamar Tersedia	
7	7	Room Control			Kamar Tersedia	
8	8	Room Control			Kamar Tersedia	
9	9	Room Control			Kamar Tersedia	
10	10	Room Control			Kamar Tersedia	

Showing 1 to 10 of 10 entries

Gambar 5.8 Halaman Data Kamar

5.2.2.2 Skenario Pengujian Mengelola Perangkat Hotel

Pada skenario pengujian ini dilakukan untuk menguji apakah admin dapat mengelola perangkat-perangkat yang ada di hotel.

Gambar 5.9 Halaman *Room Control Website*Gambar 5.10 Halaman *Lift Control*

5.2.2.3 Skenario Pengujian Melakukan *Check-in* dan *Check-out Manual*

Pada skenario pengujian ini dilakukan untuk menguji apakah admin dapat melakukan *check-in* atau *check-out* secara manual. Untuk melakukan *check-in* atau *check-out*, admin menekan tombol “CHECK IN” dan tombol “CHECK OUT” pada halaman dashboard atau halaman data kamar.

CHECK IN sukses!

Data Kamar Hotel

Show 10 entries Search:

Id Transaksi	Nomor Kamar	Deviceld	Check In	Check Out	Status	Action
2	1	Room Control	2020-06-14 05:58:46	2020-06-27 13:00:00	SUDAH CHECK IN	CHECK OUT

Showing 1 to 1 of 1 entries Previous 1 Next

Gambar 5.11 Check-in Berhasil

CHECK OUT sukses!

Data Kamar Hotel

Show 10 entries Search:

Id Transaksi	Nomor Kamar	Deviceld	Check In	Check Out	Status	Action
2	1	Room Control	2020-06-14 05:58:46	2020-06-27 13:00:00	REQUEST CHECK OUT	Konfirmasi Checkout

Showing 1 to 1 of 1 entries Previous 1 Next

Gambar 5.12 Check-out Berhasil

5.2.2.4 Skenario Pengujian Mengonfirmasi *Check-out*

Pada skenario pengujian ini dilakukan untuk menguji apakah admin dapat melakukan *check-in* atau *check-out* secara manual. Untuk melakukan *check-in* atau *check-out*, admin menekan tombol “Konfirmasi Checkout” pada halaman dashboard atau halaman data kamar.

Dashboard Dashboard

Data Kamar Hotel

Show 10 entries Search:

Id Transaksi	Nomor Kamar	Deviceld	Check In	Check Out	Status	Action
2	1	Room Control	2020-06-14 05:00:07	2020-06-27 13:00:00	REQUEST CHECK OUT	Konfirmasi Checkout

Showing 1 to 1 of 1 entries Previous 1 Next

Gambar 5.13 Konfirmasi Check-out

5.2.2.5 Skenario Pengujian Melihat Riwayat Transaksi

Pada skenario pengujian ini dilakukan untuk menguji apakah admin dapat melihat riwayat transaksi.

Transaksi Data Transaksi

Data Transaksi

Show 10 entries Search: [Tambah Transaksi](#)

Transaksi Id	Nomor Kamar	Nama User	Tanggal Pesan	Check In	Check Out
9	3	test1	2020-05-31 22:25:39	2020-05-31 13:00:00	2020-06-01 13:00:00
8	2	test1	2020-05-31 22:17:38	2020-05-30 13:00:00	2020-06-01 13:00:00
7	1	test1	2020-05-26 06:14:11	2020-05-28 13:00:00	2020-05-31 17:48:47
6	2	test1	2020-06-14 02:06:46	2020-06-14 00:00:00	2020-06-15 00:00:00
5	2	test1	2020-06-09 18:13:50	2020-06-24 00:00:00	2020-06-25 00:00:00
4	4	test1	2020-06-09 18:10:54	2020-05-11 13:00:00	2020-06-08 13:00:00
3	4	test1	2020-06-09 17:49:57	2020-05-11 13:00:00	2020-06-08 13:00:00
2	1	test1	2020-05-26 06:14:11	2020-06-14 05:58:46	2020-06-27 13:00:00

Showing 1 to 8 of 8 entries Previous 1 Next

Gambar 5.14 Riwayat Transaksi

5.2.2.6 Skenario Pengujian Melihat QR Code Check-in

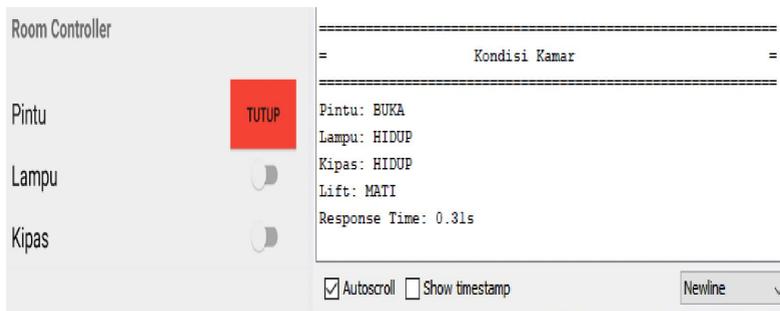
Pada skenario pengujian ini dilakukan untuk menguji apakah admin dapat melihat QR Code yang akan digunakan oleh pengunjung hotel untuk melakukan check-in.



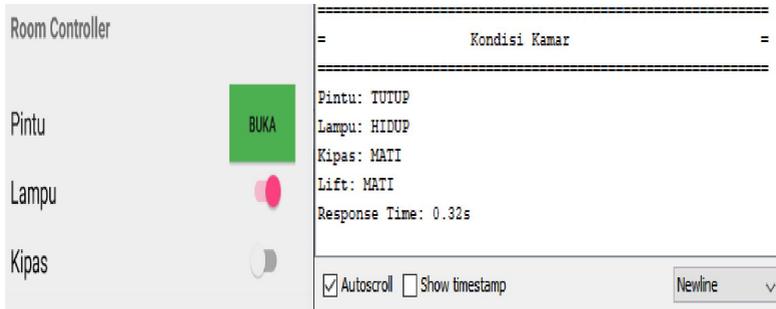
Gambar 5.15 Halaman QR Code Check-in

5.2.3 *Controlling* Alat Pada Maket atau Miniatur

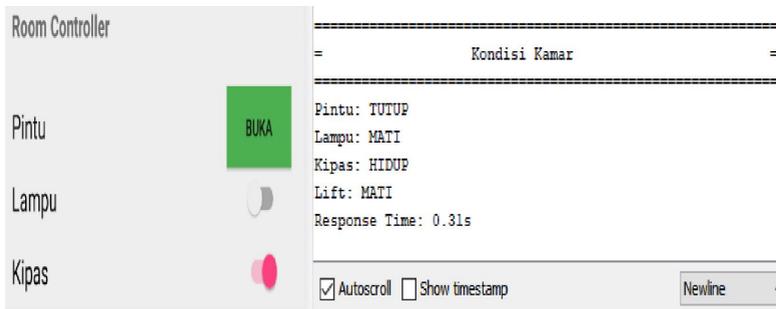
Pada bagian ini adalah hasil uji coba dari skenario mengontrol perangkat pada Aplikasi Hotel Cerdas dan *Website Monitoring* dan *Controlling*.



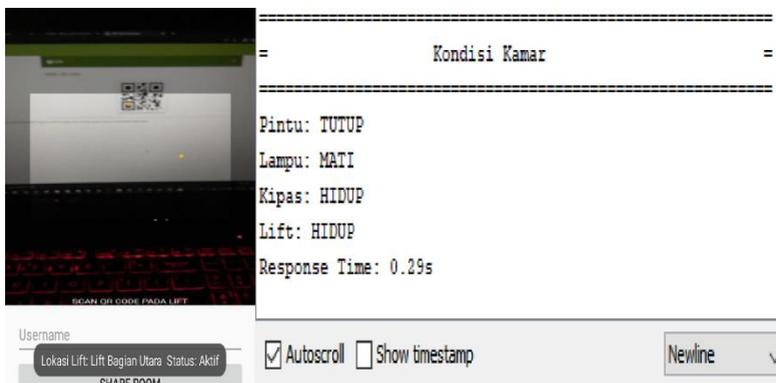
Gambar 5.16 Mengontrol Pintu



Gambar 5.17 Mengontrol Lampu

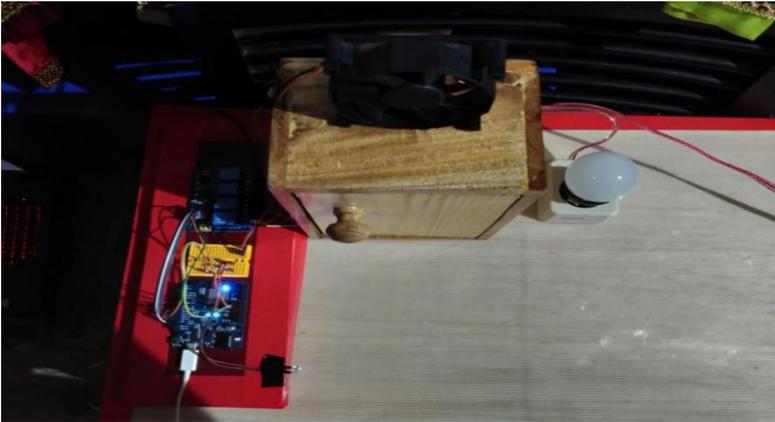


Gambar 5.18 Mengontrol Kipas



Gambar 5.19 Mengontrol Lift

Gambar diatas adalah serial monitor dari Arduino IDE yang akan menampilkan kondisi kamar ketika pengguna atau admin melakukan *controlling* perangkat. Di bawah ini akan dilampirkan gambar contoh ketika kondisi semua perangkat dihidupkan dan dimatikan. *Solenoid* pada pintu akan masuk ke dalam selama 2 detik yang menyebabkan pintu terbuka, kemudian *solenoid* akan keluar kembali setelah 2 detik.



Gambar 5.20 Perangkat Tidak Aktif



Gambar 5.21 Perangkat Aktif

Kemudian melakukan pengujian performa respons alat ketika melakukan *controlling*. Waktu respons alat akan dihitung dengan menghitung rata-rata waktu ketika alat hidup atau mati dengan melakukan 10 kali uji coba. Waktu respons termasuk waktu melakukan *HTTP request*. Hasil pengujian dapat dilihat pada Tabel 5.3 di bawah ini:

Tabel 5.3 Uji Respons Alat

No.	Waktu Respons(s)			
	Pintu	Lampu	Kipas	Lift
1	0.31	0.54	0.37	0.38
2	0.31	0.33	0.5	0.31
3	0.28	0.31	0.36	0.3
4	0.31	0.29	0.4	0.3
5	0.3	0.31	0.36	0.29
6	0.3	0.3	0.39	0.52
7	0.31	0.29	0.3	0.3
8	0.28	0.32	0.5	0.33
9	0.33	0.3	0.4	0.29
10	0.35	0.4	0.38	0.32
Jumlah	3.08	3.39	3.96	3.34
Rata-rata	0.308	0.339	0.396	0.334

Dari hasil pengujian respons alat dapat disimpulkan bahwa setidaknya membutuhkan waktu sekitar 0,3 – 0,4 detik untuk email bisa diterima oleh pemilik rumah. Dengan diadakannya pengujian tadi terlihat bahwa alat ini dapat dihidupkan dan dimatikan dengan waktu tunggu yang sangat singkat.

BAB VI

KESIMPULAN

Pada bab ini akan dijelaskan kesimpulan dari hasil ujicoba yang telah dilakukan.

6.1 Kesimpulan

Kesimpulan yang diperoleh dari Tugas Akhir ini didasarkan pada hasil uji coba dan evaluasi. Kesimpulan Tugas Akhir ini adalah sebagai berikut:

1. Pemesanan kamar hotel berhasil diimplementasikan pada *mobile apps* dan dapat melakukan pembayaran secara *online*.
2. Melakukan *check-in* dengan *scan QR Code* sudah berhasil diimplementasikan.
3. Kunci pintu dapat dibuka dengan menekan tombol buka pada halaman *controlling* yang terdapat di *mobile apps*.
4. Perangkat-perangkat pada kamar atau hotel seperti lampu, kipas, dan *lift* sudah dapat dikendalikan melalui halaman *controlling* yang terdapat pada *mobile apps*.
5. Prototipe dari Sistem Hotel Cerdas sudah berhasil dirancang dan berfungsi sesuai dengan yang diinginkan.
6. Website *Monitoring* dan *Controlling* Hotel sudah berhasil diimplementasikan dan berfungsi sesuai dengan yang diinginkan.
7. Waktu respons pintu, lampu, kipas dan lift adalah 0.308, 0.339, 0.396, 0.334. Dapat disimpulkan bahwa alat ini dapat dihidupkan dan dimatikan dengan waktu tunggu yang sangat singkat.

6.2 Saran

Saran yang dapat diberikan dari hasil uji coba dan evaluasi adalah sebagai berikut:

1. Melakukan uji coba aplikasi Android pada Android device dengan versi yang lebih rendah agar aplikasi dapat digunakan oleh beragam jenis versi.
2. Mengimplementasikan sistem keamanan untuk mendeteksi *user* yang mengakses kamar.

[Halaman ini sengaja dikosongkan]

DAFTAR PUSTAKA

- [1] William Stallings, “Operating Systems Internals and Design Principles,” in *Operating Systems Internals and Design Principles*, 7th ed., Prentice Hall, Ed. 2012, hal. 574–604.
- [2] Bemile, Richard & Achampong, Akwasi & Danquah, Emmanuel.2014. Online Hotel Reservation System. International Journal of Innovative Science, Engineering & Technology.
- [3] Mukherjee, Suprakash & Mondal, Subhendu. 2017. A Scheme for QR Code Based Smart Door Locks Security System Using an ARM Computer. 10.1007/978-981-10-2035-3_62.
- [4] N. M. Carl Hamacher, Zvonko G. Vranesic, Safwat G. Zaky, “Computer Organization and Embedded Systems,” in *Computer Organization and Embedded Systems*, 6th ed., McGraw-Hill, Ed. 2012, hal. 385–420.
- [5] Janaki Venukumar, Naveen.S. Arduino Based Door Access Control. International Journal of Research in Advent Technology, Vol.4, No.8, August 2016 E-ISSN: 2321-9637
- [6] Sublime Text, “Sublime Text 3 Tutorial.” <https://www.sublimetext.com/support> (diakses Okt 18, 2019).
- [7] SQLyog, “SQLyog Manual.” <https://sqlyogkb.webyog.com/article/23-what-is-sqlyog> (diakses Jan 05, 2020).
- [8] Arduino, “Relay Tutorial” <https://arduinogetstarted.com/tutorials/arduino-relay>

- (diakses Feb 25, 2019).
- [8] Aqib, Muhammad , “Use an Arduino and an ESP8266 module to Control a Servo Motor through a Webpage.” <https://maker.pro/arduino/tutorial/how-to-make-a-web-controlled-servo-with-arduino-and-esp8266> (diakses Des 17, 2019).
 - [9] K. V. Kai “Oswald” Seidler, “XAMPP.” <https://www.apachefriends.org/index.html> (diakses Mei 22, 2020).
 - [10] Aviani, Goran. 2019. “An introduction to HTTP: everything you need to know” <https://www.freecodecamp.org/news/http-and-everything-you-need-to-know-about-it/> (diakses Mei 22, 2020).
 - [11] Codeigniter, “CodeIgniter 3.” <https://codeigniter.com/> (diakses Mei 22, 2020).
 - [12] Arduino, “Arduino Tutorial.” <https://www.arduino.cc/en/Tutorial/HomePage?from=Main.Tutorials> (diakses Des 30, 2019).
 - [13] Alselectro, “WEMOS D1 R1.” <https://alselectro.wordpress.com/2018/04/14/wifi-esp8266-development-board-wemos-d1/> (diakses Mei 22, 2020).
 - [14] Android, “Meet Android Studio” <https://developer.android.com/studio/intro>. (diakses Des 17, 2020).
 - [15] WeMos D1 R1. <https://ns-electric.com/product/wemos-d1-r1-arduino-compatible-esp8266/>. (diakses Des 17, 2020)
 - [16] Solenoid Door Lock. <https://www.tokopedia.com/tokunik/solenoid-door-lock-dc-12v>. (diakses Des 17, 2020).

- [17] Relay 4-Channel 5V. <https://www.amazon.com/DZS-Elec-Channel-Protection-Expansion/dp/B06XRYGM2C>. (diakses Des 17, 2020).
- [18] Green LED. <https://www.sparkfun.com/products/9592>. (diakses Des 17, 2020).
- [19] Jumper Cable. <https://myduino.com/product/key-106/>. (diakses Des 17, 2020).

LAMPIRAN

Lampiran 1. Kode Sumber Tugas Akhir

Kode Sumber	Link GitHub
Arduino	https://github.com/trus25/Tugas_Akhir
Android	
Midtrans Payment Handler (folder charge)	
Web Service	https://github.com/trus25/Tugas-Akhir-Web

[Halaman ini sengaja dikosongkan]

BIODATA PENULIS



Farhan Zuhdi, lahir di Jakarta tanggal 25 November 1998. Penulis merupakan anak pertama dari tiga bersaudara. Penulis telah menempuh pendidikan formal di SDN 14 Pagi (2004-2010), SMPN 255 Jakarta (2010-2013), SMA Negeri 12 Jakarta (2013-2016). Penulis melanjutkan studi dengan berkuliah pada program sarjana (S1) di Departemen Informatika ITS. Sejak kecil, penulis suka dengan hal-hal terkait perkembangan teknologi dan perkembangan komputer. Hal tersebut juga yang mendasari penulis melanjutkan pendidikan sarjana S1 di Jurusan Teknik Informatika, Fakultas Teknologi Elektro dan Informatika Cerdas, Institut Teknologi Sepuluh Nopember, Surabaya. Selama menempuh perkuliahan, penulis aktif dalam kegiatan kepanitiaan Schematics, yaitu sebagai staff Keamanan dan Perizinan pada tahun 2017/2018 dan tahun 2018/2019.

Selama kuliah di Informatika ITS, penulis mengambil bidang minat Arsitektur dan Jaringan Komputer (AJK). Penulis dapat dihubungi melalui surel di fzuhdi50@gmail.com.