

**UNIVERSITAS DIPONEGORO**

**PERANCANGAN SISTEM *DATABASE* DAN *SERVER* PADA SISTEM KEAMANAN KUNCI PINTU GEDUNG DENGAN *ACCESS CONTROL***

**TUGAS AKHIR**

**MUHAMMAD KHOIRIL WAFI**

**21060119140133**

**FAKULTAS TEKNIK**

**DEPARTEMEN TEKNIK ELEKTRO**

**PROGRAM STUDI SARJANA**

**SEMARANG**

**JUNI 2023**



**UNIVERSITAS DIPONEGORO**

**PERANCANGAN SISTEM *DATABASE* DAN *SERVER* PADA SISTEM KEAMANAN KUNCI PINTU GEDUNG DENGAN *ACCESS CONTROL***

**TUGAS AKHIR**

**Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik**

**MUHAMMAD KHOIRIL WAFI**

**21060119140133**

**FAKULTAS TEKNIK**

**DEPARTEMEN TEKNIK ELEKTRO**

**PROGRAM STUDI SARJANA**

**SEMARANG**

**JUNI 2023**

HALAMAN PERNYATAAN ORISINALITAS

**Tugas Akhir ini adalah karya saya sendiri,**

**dan semua sumber baik yang dikutip maupun yang dirujuk**

**telah saya nyatakan dengan benar**

|  |  |  |
| --- | --- | --- |
| NAMA | : | MUHAMMAD KHOIRIL WAFI |
| NIM | : | 21060119140133 |
| Tanda Tangan | : |  |
| Tanggal | : | 23 Juni 2023 |

HALAMAN PENGESAHAN

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI**

**TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

Sebagai sivitas akademika Universitas Diponegoro, saya yang bertanda tangan di bawah ini:

Nama : Muhammad Khoiril Wafi

NIM : 21060119140133

Program Studi : SARJANA (S1)

Departemen : TEKNIK ELEKTRO

Fakultas : TEKNIK

Jenis Karya : TUGAS AKHIR

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Diponegoro **Hak Bebas Royalti Noneksklusif** (*Non-exclusive Royalty Free Right)* atas karya ilmiah saya yang berjudul: PERANCANGAN SISTEM *DATABASE* DAN *SERVER* PADA SISTEM KEAMANAN KUNCI PINTU GEDUNG DENGAN *ACCESS* *CONTROL.* Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti/Noneksklusif Universitas Diponegoro berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Semarang

Pada Tanggal : 23 Juni 2023

Yang menyatakan,

(Muhammad Khoiril Wafi)

NIM. 21060119140133

ABSTRAK

*Diisin nanti setelah perancangan selesai. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed euismod posuere urna, in finibus nisi dignissim sed. Proin in justo eu turpis interdum condimentum. Quisque eu mauris sed massa feugiat dapibus vitae sit amet elit. Donec lacinia neque non magna pellentesque, sed venenatis enim accumsan. Fusce sit amet dui magna. Vivamus fringilla lobortis vulputate. Curabitur id consequat nisl. Aliquam erat volutpat. Nulla facilisi. Vestibulum scelerisque euismod tellus nec eleifend. Nulla eu quam elit. Vivamus non lacus ligula. Aenean eu congue tellus. In suscipit risus ac elit pulvinar, in congue ex pulvinar. Nullam in felis tincidunt, tincidunt ligula et, rutrum velit. Aenean iaculis lobortis nulla nec auctor. Sed luctus elementum purus id dictum.*

***Kata Kunci****: Lorem ipsum dolor sit amet, consectetur adipiscing elit*

ABSTRACT

*Diisin nanti setelah perancangan selesai. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed euismod posuere urna, in finibus nisi dignissim sed. Proin in justo eu turpis interdum condimentum. Quisque eu mauris sed massa feugiat dapibus vitae sit amet elit. Donec lacinia neque non magna pellentesque, sed venenatis enim accumsan. Fusce sit amet dui magna. Vivamus fringilla lobortis vulputate. Curabitur id consequat nisl. Aliquam erat volutpat. Nulla facilisi. Vestibulum scelerisque euismod tellus nec eleifend. Nulla eu quam elit. Vivamus non lacus ligula. Aenean eu congue tellus. In suscipit risus ac elit pulvinar, in congue ex pulvinar. Nullam in felis tincidunt, tincidunt ligula et, rutrum velit. Aenean iaculis lobortis nulla nec auctor. Sed luctus elementum purus id dictum.*

***Key Words****:*

.

KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Allah Subhanahu Wa Ta’ala atas rahmat, karunia, taufik, dan hidayah-Nya, sehingga penulis dapat menyelesaikan tugas akhir dan penyusunan laporan ini. Tugas Akhir berjudul “PERANCANGAN SISTEM *DATABASE* DAN *SERVER* PADA SISTEM KEAMANAN KUNCI PINTU GEDUNG DENGAN *ACCESS CONTROL*” ini diajukan sebagai syarat akhir untuk menyelesaikan program Sarjana di Departemen Teknik Elektro, Fakultas Teknik, Universitas Diponegoro, Semarang. Penyusunan dan penyelesaian Tugas Akhir ini tidak lepas dari bantuan dan dukungan dari semua pihak, baik secara langsung maupun tidak langsung. Oleh karena itu, pada kesempatan kali ini penulis mengucapkan terima kasih kepada:

1. Bapak Aghus Sofwan, S.T., M.T., Ph.D., selaku Ketua Departemen Teknik Elektro Fakultas Teknik Universitas Diponegoro Semarang.
2. Bapak Dr. Munawar Agus Riyadi, S.T., M.T., selaku Ketua Program Studi S1 Teknik Elektro Universitas Diponegoro Semarang.
3. Bapak Yuli Christiyono, S.T., M.T., selaku Sekretaris Program Studi S1 Teknik Elektro Universitas Diponegoro.
4. Bapak M. Arfan, S. Kom., M. Eng., selaku Dosen Pembimbing Utama yang selalu memberikan bimbingan, arahan, dan motivasi dalam menyelesaikan tugas akhir ini.
5. Bapak Imam Santoso, S.T., M.T., selaku Dosen Pendamping yang selalu memberikan motivasi, bimbingan, serta arahan yang membangun kepada penulis.
6. Bapak Yuli Christiyono, S.T., M.T., selaku dosen wali penulis yang selalu memberikan semangat dan mendengarkan dengan baik permasalahan penulis selama menempuh studi di Departemen Teknik Elektro Universitas Diponegoro.
7. Orang tua dan keluarga besar penulis yang selalu memberikan kasih sayang, doa, motivasi, dan semangat kepada penulis.
8. Sahabat kelompok tugas akhir Henric Dhiki Wicaksono dan Novi Dianasari yang banyak membantu penulis dalam menyelesaikan tugas akhir ini.
9. Keluarga Eternity S1 Teknik Elektro Universitas Diponegoro Angkatan 2019 khususnya konsentrasi Teknologi Informasi, terimakasih atas semuanya selama penulis menimba ilmu di S1 Teknik Elektro Universitas Diponegoro.
10. Dan semua pihak yang tidak dapat penulis sebutkan satu per satu yang telah membantu dari awal hingga akhir.

Penulis menyadari bahwa dalam penulisan laporan Tugas Akhir ini tentunya ada kekurangan. Oleh karena itu, kritik dan saran yang dapat membangun diperlukan demi kebaikan dan kesempurnaan penyusunan laporan di masa yang akan datang. Semoga Tugas Akhir ini dapat memberikan manfaat dan menambahkan pengetahuan bagi kita.

Semarang, 23 Juni 2023

(Muhammad Khoiril Wafi)

NIM. 21060119140133

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

# **BAB I**

**PENDAHULUAN**

1. Latar Belakang

Kemanan menjadi hal yang harus diperhatikan dalam sebuah gedung atau bangunan. Pada saat ini sistem penguncian masih banyak menggunakan penguncian tradisional dengan menggunakan kunci fisik yang tidak efisien mengingat jumlah ruangan yang banyak, kunci fisik juga mempunyai tingkat keaman yang kurang dikarenakan kunci rentan untuk dicuri atau diduplikasi[1].

Mengelola sejumlah besar ruangan menjadi tantangan tersendiri. Seiring bertambahnya ruangan yang harus dikelola maka bertambah juga konpleksitas tugas yang harus dikerjakan seperti mengatur siapa saja yang dapat mengakses ruangan tersebut, bertambahnya ruangan juga meningkatkan resiko kunci hilang atau tertukar sehingga ruangan tidak bisa diakses[2].

Masalah keamanan ruangan dalam sebuah gedung dan efektifitas dapat diselesaikan dengan menggunakan sebuah sistem penguncian cerdas yang terorganisasi dan kerkoneksi ke sebuah sistem manajemen kunci pintu (*access control*) yang memiliki tingkat keamana yang lebih tinggi, adaptif dan fleksibel[3], [4].

Untuk mendukung kinerja dari sistem penguncian yang terorganisasi maka diperlukan sebuah *server* dan penyimpanan data. sebuah *server* akan menjalankan kode program yang bertugas untuk mengatur dan mengawasi semua aktifitas sistem penguncian dan sebuah penyimpanan data digunakan untuk menyimpan data-data seperti data pengguna, kunci, dan *backup*[5].

Dengan demikian, perancangan sistem *database* dan *server* pada sistem keamanan kunci pintu gedung dengan *access control* diharapkan dapat memberikan solusi yang efektif dalam meningkatkan keamanan kunci pintu gedung dan memberikan kenyamanan serta kemudahan dalam pengelolaannya.

1. Tujuan

Tujuan tugas akhir ini adalah untuk merancang sistem *database* dan *server* pada sistem keamanan kunci pintu gedung dengan *access control.*

1. Batasan Masalah

Dalam tugas akhir ini, telah ditentukan batasan-batasan masalah sebagai berikut:

1. Perancangan sistem *database* dan *server* akan difokuskan pada aplikasi *access control* untuk kunci pintu gedung. Hal ini meliputi perancangan struktur *database* untuk menyimpan informasi pengguna, akses pintu gedung, dan sejarah aktivitas akses serta perancangan server untuk menjalankan sistem penguncian.
2. Perancangan sistem *server* akan berfokus pada pemrosesan data dan komunikasi antara sistem *access control* dan perangkat keras.
3. Sistem keamanan kunci pintu gedung dengan *access control* akan mencakup teknologi seperti perangkat lunak *access control* untuk mengatur akses pengguna.
4. Sistem keamanan akan diimplementasikan pada gedung yang memiliki pintu masuk terbatas dengan akses terbatas pada karyawan atau pihak tertentu.
5. Penyimpanan data pengguna dan aktivitas akses akan disimpan dengan cara yang aman dan terenkripsi untuk memastikan keamanan informasi.
6. Pengembangan sistem *access control* harus memenuhi standar keamanan dan regulasi yang berlaku dalam industri atau lingkungan operasional gedung.
7. Sistem keamanan *access control* harus mudah dioperasikan dan dikelola oleh administrator gedung dengan tingkat keamanan dan kontrol yang optimal.
8. Sistematika Penulisan

Sistematika penulisan dalam laporan Tugas Akhir ini adalah sebagai berikut:

**BAB I PENDAHULUAN**

Bab ini berisi latar belakang, tujuan, batasan masalah, dan sistematika penulisan.

**BAB II DASAR TEORI**

Bab ini berisi teori yang melandasi dalam perancangan.

**BAB III PERANCANGAN SISTEM**

Bab ini berisi tentang perancangan sistem *database* dan *server* serta sistem keamanan kunci pintu gedung dengan *access control*.

**BAB IV PENGUJIAN DAN ANALISIS**

Pada bab ini dilakukan pengujian dan analisis hasil pengujian.

**BAB V PENUTUP**

Bab ini berisi kesimpulan dan saran terhadap hasil pengujian dan analisis yang telah dibuat.

**BAB II**

**DASAR TEORI**

1. Kendali Akses (Access Control)

Kendali akses atau *access control* merupakan sebuah mekanisme kebijakan yang digunakan untuk membatasi dan mengatur hak akses pengguna terhadap suatu sumber dayaatau fasilitas tertentu. Kendali akses dapat diaplikasikan secara *hardware* maupun *software* yang akan menjadi pintu masuk pengguna sebelum mengakses suatu fasilitas. Dengan menggunakan kendali akses kita bisa mengatur dan membatasi akses pengguna sehingga hanya pengguna tertentu yang diijinkan yang bisa mengakses sumber daya yang dilindungi[6].

Terdapat beberapa pendekatan yang biasanya digunakan untuk membatasi akses pengguna dengan menggunakan kendali akses, diantaranya yaitu :

1. *Access Control List*

ACL atau *Access Control List*  merupakan sebuah daftar yang berisi aturan-aturan atau daftar pengguna yang diijinkan untuk mengakses sebuah sumber daya yang dilindungi[7]. Dengan menggunakan ACL kita dapat mambatasi hak akses dengan hanya mengijinkan pengguna yang terdapat didalam list untuk mengakses sumbser daya atau fasilitas tertentu.

1. *Role-based Access Control*

RBAC atau *Role-based Access Control* merupakan pendekatan kendali akses yang tidak memberikan hak akses langsung ke pengguna melainkan memberikan hak akses berdasarkan peran yang dimiliki oleh pengguna seperti *admin*, operator, *supervisor*, dan lain sebagaimnya.

1. Internet of Things

Internet of Things atau IoT merupakan sebuah konsep yang digunakan untuk mengambangkan koneksifitas internet, dengan menggunakan konsep IoT kita dapat menghubungkan peralatan seperti sensor dan aktuator yang terhubung ke internet menjadi sebuah kesatuan sistem yang dapat dikoordinasi atau dikendalikan secara efektif[8].

Terdapat tiga elemen utama didalam konsep IoT, yaitu :

1. Perangkat IoT

Perangkat IoT merupakan sebuah alat biasanya berupa sensor atau aktuator atau gabungan keduanya yang dilengkapi dengan modul IoT untuk terhubung ke server seperti modul WiFi atau lainnya[8].

1. Koneksi Internet

Untuk saling berkomunikasi dan bertukas data, Perangkat IoT memerlukan koneksi internet. Koneksi ini digunakan sebagai jalur pengiriman data untuk saling berkomunikasi andara perangkat IoT dengan server. untuk mendapatkan koneksi internet perngkat IoT bisasanya menggunakn jaringan WiFi atau jaringan seluler[8].

1. Sistem Server

Sistem server menjadi pusat kendali yang mengatur semua perangkat IoT. Server juga menyediakan penyimpanan data yang digunakan untuk menyimpan aplikasi maupun data pengukuran sensor atau lainnya[8].

1. Database MySQL

Database atau basi data merupakan sekumpulan data yang terintegrasi dan diatur sedemikian rupa sehingga data tersebut dapat dicari, diambil, ditambahkan, dan diolah dengan tepat[6].

Selain berisikan data, database juga berisikan informasi lainnya seperti tipe data, nama-nama kolom dan baris. Adapun fungsi lain dari database yaitu[9] :

1. Mempermudah identifikasi data dengan cara pengelompokkan data, salah satu contohnya dengan pembuatan beberapa tabel atau *field* yang berbeda-beda.
2. Meminimalisir suatu data ganda.
3. Penyimpanan secara digital.
4. Menjadi alternatif lain terkait masalah penyimpanan ruang dalam suatu aplikasi.

MySQL merupakan sistem database relasional yang berarti pada MySQL data akan disimpan dalam tabel-tabel yang terdiri dari beberapa kolom, data yang disimpan dapat berupa teks, angka, boolean, waktu, hari dan lain sebagainya tergantung dengan tipe data pada setiap kolomnya. Dalam MySQL juga terapat relasi antar tabel yang memberikan informasi hubungan antara tabel satu dengan tabel lainnya serta perilakunya terhadap perubahan yang terjadi pada tabel induknya.

1. Server

Server merupakan sebuah komputer yang memberikan layanan tertentu kepada pengguna atau client, server bertanggung jawab untuk menerima permintaan, mengolah dan memberikan respon atas permintaan tersebut. Sebuah server akan menyimpan kode program aplikasi yang akan diakses oleh pengguna[10]. Berdasarkan fungsi dan tugasnya terdapat beberapa jenis server, diantaranya :

1. Web Server

Web server digunakan untuk memberikan layanan dengan menggunakan protokol http, contoh : Apache, Nginx dan Microsoft IIS[10].

1. Database Server

Database server digunakan untuk memberikan layanan penyimpanan data secara terpusat dan mendistribusi kan data ke pengguna menggunakn jaringan wireless atau kabel, contoh : MySQL, PostgreSQL dan Oracle[10].

1. Mail Server

Mail server digunakan untuk memberikan layanan berupa pengiriman email, contoh : Mercury, Sendmail dan Gmail[10].

1. Application Programming Interface

Application Programming Interface atau API merupakan sebuah protokol yang digunakan untuk menampilkan layanan atau data yang disediakan oleh sebuah aplikasi melalui sumber daya yang telah ditentukan. Dengan menggunakan API, aplikasi lain dapat mengakses layanan atau data tanpa harus mengimplementasikan prosedur atau objek yang mendasarinya[11].

Sebuah API akan mengirimkan data dengan fromat JSON sebagai respon dari permintaan. JSON atau Java Script Object Notation merupakan sebuah format data yang yang ditemukan oleh Douglas Crockford pada tahun 2006, JSON menggunakan sintak objek javascript sebagai format data berbasis teks, JSON memiliki ukuran data yang lebih kecil serta waktu proses yang lebih cepat jika dibandingkan dengan format data lainnya[12].

Dengan menggunakan sebuah API maka pengguna tidak akan terhubung secara langsung dengan database, sehinggga akan meningkatkan keamanan dari sistem, jika sewaktu waktu terjadi hal-hal yang tidak di ingin kan seperti rusaknya data atau sistem di retas oleh pihak yang tidak bertanggung jawab data asli tetap aman tersimpan pada database tanpa mengalami gangguan sedikitpun[12].

1. Laravel

Laravel merupakan kerangka kerja pemrograman web dengan menggunakan bahasa PHP yang terbuka dan gratis, laravel diperuntukkan untuk pengembangan aplikasi berbasis web dengan menggunakan pendekatan pola MVC atau Model View Controller[13]. Pada laravel terdapat 3 bagian utama yang membangun kerangka kerja pemrograman, yaitu :

1. Routing

Routing bertugas untuk menerima dan mengolah permintaan yang dikirimkan oleh pengguna. Permintaan akan di teruskan ke controller untuk diolah sedemikian rupa sehingga menghasilkan keluaran yang akan dikembalikan ke pengguna.

1. Controller

Controller digunakan untuk mengolah data yang diminta oleh pengguna, sebuah kontroller tidak langsung menerima permintaan dari pengguna melainkan controller akan dipanggil oleh bagian routing pada saat ada permintaan.

1. Model

Model digunakan sebagai jembatan yang menghubungkan controller dengan database, model tertanggung jawab untuk meberikan koneksi ke database dengan aman.

1. Web Socket

Websocket merukan sebuah protokol komunikasi web berbasi client-server, keberadaan websocket dinilai dapat menggantikan teknilogi AJAX sebagai pendahulu komunikasi client-server. Websocket merupakan teknologi yan gmampu memberikan performa terbaik ketika diimplementasikan dalam sistem dengan rate-request tinggi,dibandingkan dengan teknologi komunikasi lain termasuk AJAX[14].

Websocket memungkinkan komunikasi dua arah antara client dan server dengan menggunakan koneksi yang sudah terjalin, hal ini dikarenakan pada websocket koneksi akan terus terjalin selama tidak terjasi error atau ada permintaan pemutusan koneksi.

1. Crontab

Crontab merupakan fitur yang disediakan oleh sistem operasi linux yang digunakan untuk menjalankan perintah atau script secara berkesinambungan[15]. Cron akan menjalankan perintah yang kita masukkan secara periodik sesuai dengan waktu yang kita masukkan sehingga kita bisa menjalankan perintah yang harus dijalankan secara periodik seperti untuk mengecek jadwal setiap satu menit dan lain sebagainya.

1. Penelitian Terdahulu

Beberapa penelitian terdahulu yang dijadikan referensi dalam perancangan sistem database dan server pada sistem keamanan kunci pintu gedung menggunakan access control adalah :

**Tabel 2.1** Ringkasan Penelitian Terdahulu

|  |  |  |  |
| --- | --- | --- | --- |
| **Judul** | **Tahun** | **Penulis** | **Pembahasan** |
| *Perancangan Database IoT Berbasi Cloud dengan Restful API* | 2021 | M. Kasyful Anwar | Dalam penelitian ini dibahas mengenai rancangan database berbasis cloud dengan Restful API untuk IoT agar data IoT aman dan memiliki *throughput* yang bagus dengan struktur data yang diatur pada database. |
| *Development of Intelligent Door Lock System for Room Management Using Multi Factor Authentication* | 2023 | I. Hermawan, D. Arnaldy, P. Oktivasari, dan D. A. Fachrudin | Penelitian ini menjelaskan mengenai permasalahan yang muncul pada manajemen kunci secara tradisional seperti jumlah kunci yang banyak, mudah hilang dan mudah diduplikasi. |
| *RFID and GSM Based Attendance Monitoring System using door locking / unlocking system and Its Hardware Implementation* | 2015 | A. Jain, V. L. Kalyani, dan B. Nogiya | Penelitian ini menjelaskan penggunaan RFID untuk melakukan absesnsi siswa sekaligus sebagai kunci untuk membuka pintu ruang kelas. |
| *Implementasi Teknologi Websocket dalam Pengembangan Sistem Berbagi Lokasi Berbasis Web* | 2017 | A. B. Al-Ilman dan T. A. Kurniawan | Penelitian ini menjelaskan tentang penggunaan websocket sebagai komunikasi antara client dan server. |

BAB III

**PERANCANGAN SISTEM**

1. Metode Perancangan

Metode perancangan perangkat lunak dalam pelaksanaan tugas akhir ini menggunakan pendekatan System Development Life Cycle Waterfall dengan diagram alur perancangan sebagai berikut :

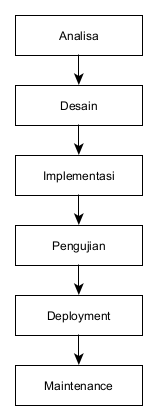


Diagram Alur Perancangan

1. Analisa

Pada tahap analisa, dilakukan pengumpulan informasi mengenai sistem yang akan dibuat. Analisa meliputi tujuan dan carakerja sistem, batasan-batasan sistem dan teknologi yang akan digunakan.

Analisa dilakukan bersama tim dengan melakukan penelusuran pustaka serta referensi yang relefan dan disertai dengan diskusi sehingga menghasilkan keluaran berupa gambaran umum mengenai bagian-bagian dari sistem yang akan digunakan pada tahap selanjutnya.

1. Desain

Setelah mendapatkan gambaran mengenai sistem yang akan dibuat, selanjutnya dilakukan pemodelan sistem dengan menggunakan diagram pemodelan. Diagram pemodelan digunakan untuk menggambarkan hubungan dan interaksi antar komponen didalam sistem, dengan menggunakan diagram maka akan menghasilkan gambar sistem yang lebih rinci yang digunakan untuk membangun sistem.

1. Implementasi

Pada tahap implementasi, gambar pemodelan dari sistem kemudian diterjemahkan atau diimplementasikan menggunakan kode program untuk membuat sistem dapat bekerja sesuai dengan dasain awal yang telah ditentukan.

Implementasi menggunakan pbahasa pemrograman PHP dengan menggunakan framework Laravel. Laravel mempunyai karakteristik RAD atau Rapid Application Development sehingga implementasi dapat dilaksanakan dengan cepat dan efisien, Laravel juga menyediakan komponen-komponen yang akan digunakan didalam sistem seperti autentikasi, antrian, notifikasi pesan, websocket, dan lain sebagainaya sehinggadapat mempermudah proses implementasi.

1. Pengujian

Pada tahap pengujian, sistem yang telah dibangun diuji kinerjanya dengan mengirimkan beberapa permintaan data dan menganalisa hasil keluaran yang diberikan apakah sesuai atau tidak.

Proses pengujian dilakukan dengan menggunakan software Postman. Postman merupakan aplikasi yang digunakan untuk melakukan pengujian API baik berbasis HTTP maupun Socket. Pengujian lebih lanjut dilakukan untuk menguji kemanan pada autentikasi serta koneksi websocket yang terjalin.

1. Deployment

Pada tahap deployment aplikasi yang sudah selesai dilakukan pengujian dan dinyatakan lolos uji selanjutkan akan dipasang atau diinstall pada server. Pada proses deployment juga akan dilakukan pemasangan aplikasi tambahan seperti basis data, penjadwalan, firewall, dan lain sebagainya.

1. Maintenance

Pada tahap maintenance atau perawatan dilakukan pengecekan secara berkala pada aplikasi yang sudah berjalan pada server. dengan adanya perawatan secara teratur jika ditemukan permasalahan atau komponen yang tidak bekerja maka akan langsung diperbaiki sehingga tidak mengganggu kinerja dari server.

1. Perancangan Database

Database digunakan untuk menyimpan data yang digunakan dalam sistem penguncian pintu gedung seperti data pengguna, data pintu, data riwayat akses dan lain sebagainya. Database yang digunakan harus dapat menunjang kinerja sistem dengan karakteristik respon yang cepat dan pengelolaan data terstruktur.

Berdasarkan penelitian [16] yang membandingkan kinerja dari berbagai tipe dan jenis database didapatkan hasil penggunaan MySQL menunjukkan hasil kinerja yang bagus dalam hal waktu eksekusi permintaan, dengan sistem penyimpanan data bersifat relasional dan terstruktur maka MySQL dapat diterapkan pada sistem penguncian pintu gedung ini.

1. Tabel Pengguna

Tabel pengguna (users) menyimpan data pengguna yang akan digunakan sebagai identitas pengguna untuk keperluan autentikasi dan digunakan juga untuk menyimpan riwayat akses. Struktur dari tabel pengguna dapat dilihat pada tabel x.x berikut :

Tabel Pengguna

| **No** | **Nama Kolom** | **Tipe Data** | **Keterangan** |
| --- | --- | --- | --- |
| 1 | id | char(36) | kunci primer tabel |
| 2 | added\_by | char(36) | referensi aktor |
| 3 | email | varchar(255) | alamat email pengguna |
| 4 | password | varchar(255) | password pengguna |
| 5 | remember\_token | varchar(255) | token untuk fitur "ingat saya" |
| 6 | name | varchar(255) | nama penggua |
| 7 | phone | varchar(255) | nomor hp pengguna |
| 8 | gender | enum | jenis kelamin pengguna |
| 9 | role | enum | jabatan pengguna |
| 10 | avatar | varchar(255) | foto profil pengguna |
| 11 | email\_verified\_at | timestamp | waktu verifikasi email |
| 12 | created\_at | timestamp | waktu data dibuat |
| 13 | updated\_at | timestamp | waktu data diedit |

1. Tabel Gedung

Tabel gedung (offices) digunakan untuk meyimpan data gedung. Pada sistem penguncian pintu gedung ini terdapat beberapa gedung dengan masing-masing operator yang mengatur pengelolaan kunci gedung. Struktur dari tabel gedung dapat dilihat pada tabel x.x berikut :

Tabel Gedung

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Kolom** | **Tipe Data** | **Keterangan** |
| 1 | id | char(36) | kunci primer tabel |
| 2 | user\_id | char(36) | referensi operator gedung |
| 3 | name | varchar(255) | nama gedung |
| 4 | created\_at | timestamp | waktu data dibuat |
| 5 | updated\_at | timestamp | waktu data diedit |

1. Tabel Pintu

Tabel pintu (doors) digunakan untuk menyimpan data pintu pada setiap gedung. Tabel pintu akan memberikan informasi terkait pintu seperti status koneksi dan status penguncian. Struktur dari tabel pintu dapat dilihat pada tabel x.x berikut :

Tabel Pintu

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Kolom** | **Tipe Data** | **Keterangan** |
| 1 | id | char(36) | kunci primer tabel |
| 2 | office\_id | char(36) | referensi gedung |
| 3 | socket\_id | char(36) | identitas websocket |
| 4 | name | varchar(36) | nama pintu |
| 5 | device\_name | varchar(36) | username pintu |
| 6 | device\_pass | varchar(36) | password pintu |
| 7 | key | varchar(36) | kunci pintu |
| 8 | is\_lock | tinyint(1) | status penguncia |
| 9 | created\_at | timestamp | waktu data dibuat |
| 10 | updated\_at | timestamp | waktu data diedit |

1. Tabel Jadwal

Tabel jadwal (schedules) digunakan untuk menyimpan jadwal operasional pintu secara otomatis. Pada sistem penguncian pintu gedung ini terdapat fitur yang memungkinkan operator untuk mengatur pintu terbuka dan terkunci secara otomatis pada rentang waktu tertentu. Struktur dari tabel jadwal dapat dilihat pada tabel x.x berikut :

Tabel Jadwal

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Kolom** | **Tipe Data** | **Keterangan** |
| 1 | id | char(36) | kunci primer tabel |
| 2 | office\_id | char(36) | referensi gedung |
| 3 | name | varchar(255) | nama jadwal |
| 4 | date\_begin | date | tanggal mulai |
| 5 | date\_end | date | tanggal berakhir |
| 6 | time\_begin | time | waktu mulai |
| 7 | time\_end | time | waktu berakhir |
| 8 | is\_repeating | tinyint(1) | status perulangan |
| 9 | day\_repeating | varchar(255) | perulangan hari |
| 10 | status | enum | status pelaksanaan jadwal |
| 11 | created\_at | timestamp | waktu data dibuat |
| 12 | updated\_at | timestamp | waktu data diedit |

1. Tabel Jadwal Pintu

Tabel jadwal pintu (door\_schedules) digunakan untuk menyimpan pintu-pintu yang akan dibuka/dikunci menggunakan jadwal yang telah dibuat. Setiap jadwal akan berisi beberapa pintu yang akan dibuka/dikunci sehingga pintu-pintu tersebut dapat dibuka/dikunci secara bersamaan menggunakan jadwal yang sudah dibuat. Struktur dari tabel jadwal pintu dapat dilihat pada tabel x.x berikut :

Tabel Jadwal Pintu

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Kolom** | **Tipe Data** | **Keterangan** |
| 1 | id | char(36) | kunci primer tabel |
| 2 | schedule\_id | char(36) | referensi ke jadwal |
| 3 | door\_id | char(36) | referensi ke pintu |
| 4 | created\_at | timestamp | waktu data dibuat |
| 5 | updated\_at | timestamp | waktu data diedit |

1. Tabel Akses

Tabel akses (access) digunakan untuk menyimpan data kartu akses pengguna ke masing-masing. Dengan menggunakan kartu akses yang tersimpan didalam database maka pengaturan akses pengguna dapat dilakukan dengan efisien dan realtime. Struktur dari tabel akses dapat dilihat pada tabel x.x berikut :

Tabel Akses

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Kolom** | **Tipe Data** | **Keterangan** |
| 1 | id | char(36) | kunci primer tabel |
| 2 | user\_id | char(36) | referensi ke pengguna |
| 3 | door\_id | char(36) | referensi ke pintu |
| 4 | time\_begin | time | waktu mulai |
| 5 | time\_end | time | waktu berakhir |
| 6 | date\_begin | date | tanggal mulai |
| 7 | date\_end | date | tanggal berakhir |
| 8 | is\_temporary | tinyint(1) | status akses sementara |
| 9 | is\_running | tinyint(1) | status akses sedang berjalan |
| 10 | created\_at | timestamp | waktu data dibuat |
| 11 | updated\_at | timestamp | waktu data diedit |

1. Tabel Riwayat Akses

Tabel riwayat akses (access\_logs) digunakan untuk menyimpan riwayat akses pengguna ke pintu. Dengan menggunakan riwayat akses pada aktifitas pengguna akan tercatat didalam database dan dapat dilihat oleh masing-masing operator gedung. Struktur dari tabel riwayat akses dapat dilihat pada tabel x.x berikut :

Tabel Riwayat Akses

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Kolom** | **Tipe Data** | **Keterangan** |
| 1 | id | char(36) | kunci primer tabel |
| 2 | user\_id | char(36) | referensi ke pengguna |
| 3 | office\_id | char(36) | referensi ke gedung |
| 4 | door\_id | char(36) | referensi ke pintu |
| 5 | log | text | pesan log |
| 6 | created\_at | timestamp | waktu data dibuat |
| 7 | updated\_at | timestamp | waktu data diedit |

1. Tabel OTP

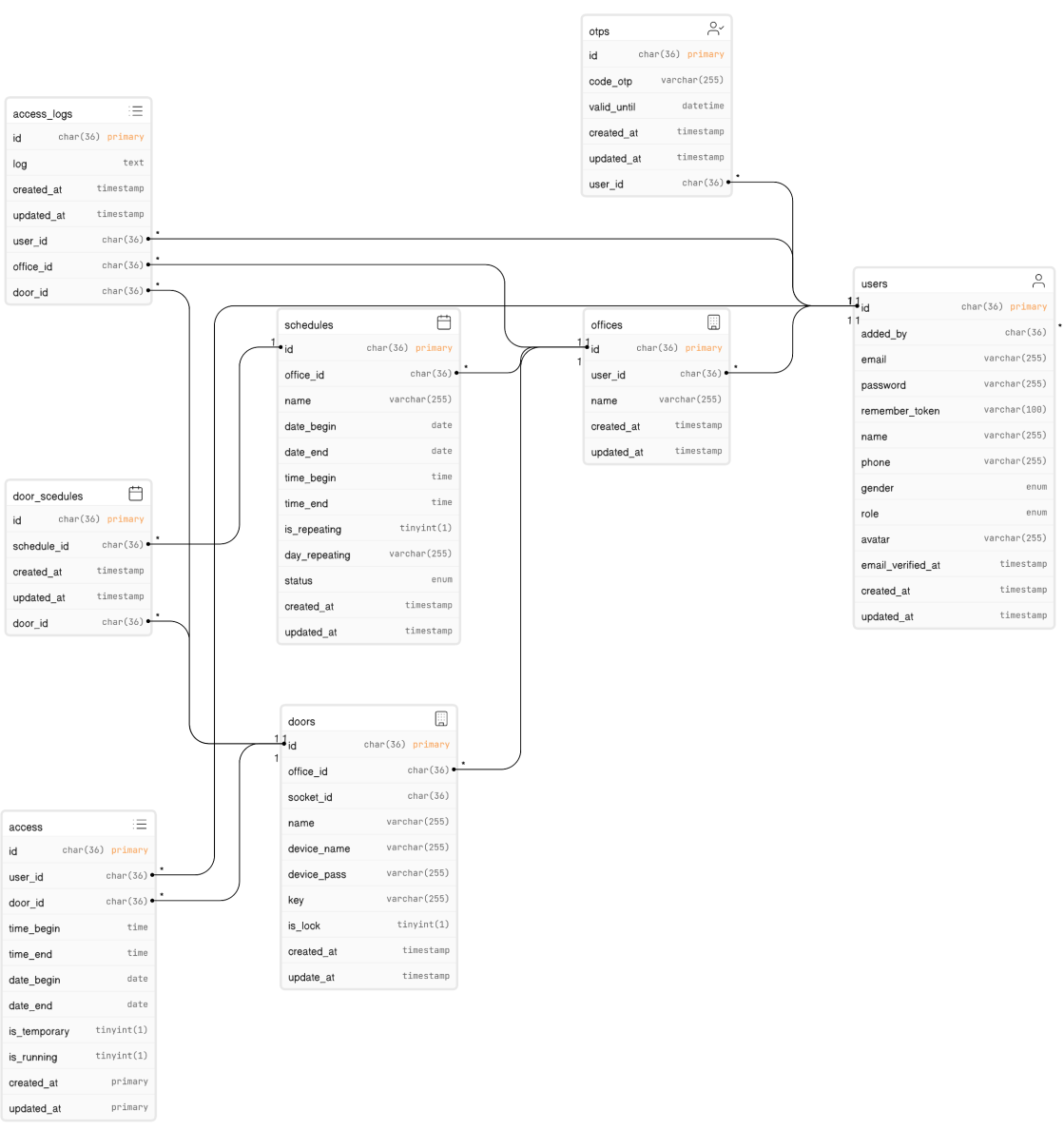
Tabel OTP (otps) atau One Time Password digunakan untuk menyimpan data verifikasi kode OTP. Dengan menggunakan verifikasi OTP maka hanya pengguna dengan email aktif yang akan diterima sehingga meningkatkan keamanan dan keabsahan data pengguna. Struktur dari tabel OTP dapat dilihat pada tabel x.x berikut :

Tabel OTP

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Kolom** | **Tipe Data** | **Keterangan** |
| 1 | id | char(36) | kunci primer tabel |
| 2 | user\_id | char(36) | referensi ke pengguna |
| 3 | code\_opt | varchar(255) | kode OTP |
| 4 | valid until | datetime | hari dan tanggal kadaluarsa |
| 5 | created\_at | timestamp | waktu data dibuat |
| 6 | updated\_at | timestamp | waktu data diedit |

1. Relasi Antar Tabel

Dengan menggunakan relational database tentunya akan terdapat hubungan atau relasi antara dua tabel atau lebih, relasi memberikan informasi mengenai hubungan antara dua tabel atau lebih yang saling berkaitan beserta dengan perilaku terhadap perubahan data pada tabel induknya.



Relasi Antar Tabel

Dalam skema database yang dirancang pada sistem penguncian pintu gedung ini terdapat 12 relasi antar tabel yaitu sesuai dengan tabel x.x berikut :

Tabel relasi

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No** | **Tabel Induk** | **Anak** | | **Jenis Hubungan** | **On Update** | **On Delete** |
| **Tabel** | **Kolom** |
| 1 | users | users | added\_by | one to many | cascade | restrict |
| 2 | users | access | user\_id | one to many | cascade | cascade |
| 3 | doors | access | door\_id | one to many | cascade | cascade |
| 4 | users | access\_logs | user\_id | one to many | cascade | cascade |
| 5 | doors | access\_logs | door\_id | one to many | cascade | cascade |
| 6 | offices | access\_logs | office\_id | one to many | cascade | cascade |
| 7 | offices | doors | office\_id | one to many | cascade | cascade |
| 8 | schedules | door\_schedule | schedule\_id | one to many | cascade | cascade |
| 9 | doors | door\_schedule | door\_id | one to many | cascade | cascade |
| 10 | users | offices | user\_id | one to one | cascade | restrict |
| 11 | users | otps | user\_id | one to one | cascade | cascade |
| 12 | offices | schedules | office\_id | one to many | cascade | cascade |

Dengan menggunakan relasi seperti yang tertulis pada tabel x.x diatas dapat memberikan struktur data yang sesuai dengan menghilangkan data ganda atau perulangan, sebagai contoh pada beberapa tabel yang membutuhkan data pengguna seperti tabel access, opts, offices dan access\_logs data pengguna tidak dicantumkan didalam tabel tersebut melainkan berada pada tabel users yang menjadi induk dari tabel-tabel tersebut.

Dengan menggunakan relasi seperti yang tertulis pada tabel x.x diatas juga dapat memberikan konsistensi data, seperti contoh pada tabel offices yang berelasi dengan tabel users dimana pada perilaku on delete restrict maka penghapusan pada data pengguna yang berkaitan dengan tabel offices akan dicegah sehingga menjamin bahwa setiap gedung memiliki seorang operator.

1. Perancangan Backend API

Backend API digunakan sebagai salah satu jalur komunikasi yang menghubungkan antara perangkat mobile dan perangkat kunci pintu berbasis IoT dengan server. Selain terdapat website yang digunakna sebagai antarmuka utama dalam mengelola sistem ini juga terdapat aplikasi berbasis mobile yang digunakan untuk menunjang kinerja dari sistem terutama pada bagian yang membutuhkan teknologi yang belum disediakan oleh browser seperti koneksi bluetooth dan pindai kode QR menggunakan kamera.

Backend API yang dikembangkan menggunakan bahasa pemrograman PHP dengan menggunakan kerangka kerja dari Laravel. Laravel menyediakan beberapa fitur yang siap digunakan seperti autentikasi login, reset password, notifikasi email dan lain sebagaimnya sehingga mempermudah dan mempercepat proses pengkodean.

1. Diagram Use Case API

Tidak semua fitur yang ada pada website diimplementasikan dalam bentuk API, hal ini berkaitan dengan kompleksitas dari sistem yang dikembangkan, fitur yang berkaitan dengan manajemen penguncian seperti menambah pintu baru, membuat jadwal, menambahakan pengguna baru, memberikan akses ke pengguna hanya bisa dilakukan melalui website hal ini dimaksudkan untuk mempertahankan kontrol yang lebih ketat terhadap akses dan perubahan yang dilakukan pada sistem. Dengan membatasi akses langsung melalui API, administrator sistem dapat memastikan bahwa hanya proses yang telah melalui lapisan keamanan dan validasi yang benar yang dapat mempengaruhi manajemen penguncian dan pengaturan lainnya. Diagram Use Case dari Backend API dapat dilihat pada gambar x.x dibawah.

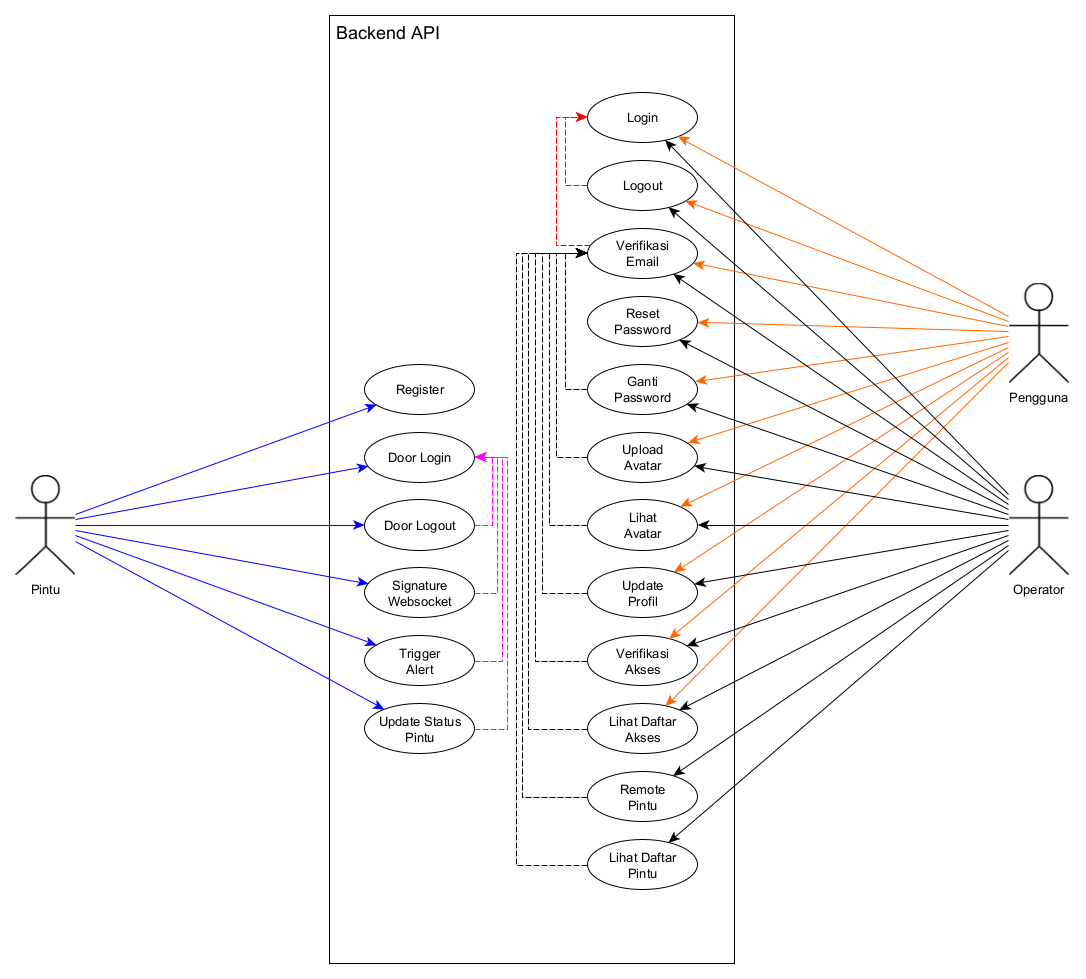


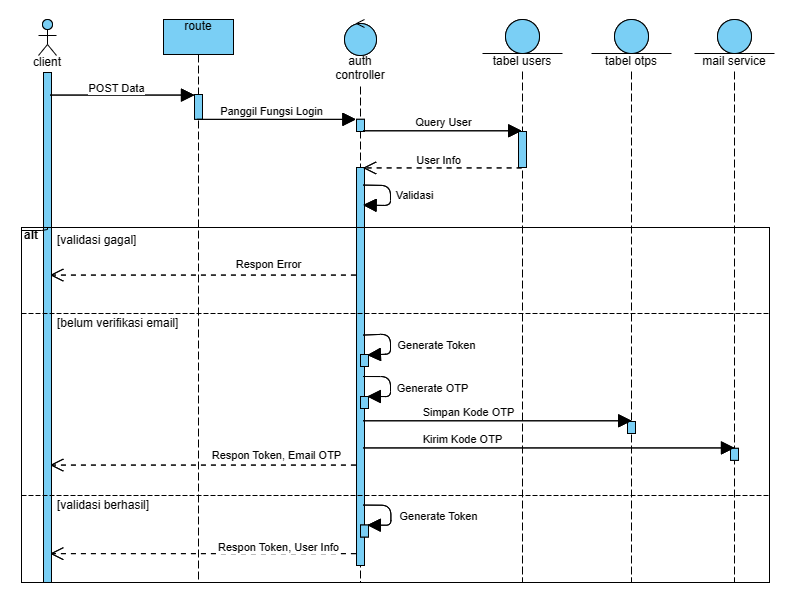
Diagram Use Case API

Dapat dilihat pada gambar x.x diatas, terdapat 3 aktor yang berinteraksi dengan sistem melalui API yaitu pintu, pengguna dan operator. Pintu merupakan perangkat IoT yang digunakan untuk melakukan penguncian pada gedung, sedangkan pengguna dan operator merupakan aplikasi mobile yang digunakan sebagai antarmuka sistem.

Pada gambar x.x diatas juga terlihat beberapa fungsi seperti update profil, lihat avatar, verifikasi akses dan lain sebagainya merupakan extends dari fungsi verifikasi email yang menandakan bahawa fungsi-fungsi tersebut hanya bisa diakses jike client sudah melakukan proses verifikasi email. Fungsi verifikasi email juga merupakan extends dari fungsi login sehingga fungsi verifikasi email hanya bisa diakses jika client sudah mellakukan autentikasi login. Hal ini dimaksudkan supaya hanya client yang terautentikasi dan terverifikasi yang dapat mengakses sumber daya yang disediakan.

1. Fungsi Login

Fungsi login digunakan untuk melakukan autentikasi client melalui username dan password yang dikirimkan, fungsi login juga mengecek apakan pengguna dan operator sudah melakukan verifiaksi email, jika belum maka login akan tertahan sampai pengguna melalukan verifikasi email. Diagram fungsi login dapat dilihat pada gambar x.x dibawah.

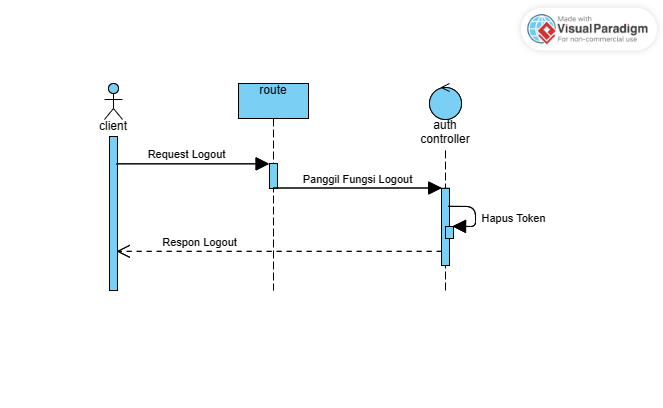


Sequence Diagram Login

Fungsi login dimulai oleh client, client akan mengirimkan username dan password mereka melalui endpoint “/login” kemudian route akan memanggil fungsi login didalam kontroler, kontroler akan memeriksa username dan password dengan melakukan query ke tabel users, jika cocok maka kontroler akan membuat token menggunakan modul sanctum. Sanctum adalah sebuah paket autentikasi dan autorisasi yang disediakan oleh Laravel yang dirancang untuk memudahkan implementasi otentikasi API yang sederhana namun aman pada aplikasi Laravel. Setelah mendapatkan token kemudian kontroler akan mengembalikan token tersebut disertai dengan detail client seperti nama, email, nomor hp dan lain sebagainya. Jika terdeteksi client belum malakukan verifikasi email maka kontroler akan membuat kode OTP atau One Time Password yang merupakan 6 digit angka acak dan mengirimakan kode terkesebut ke email client. Jika autentikasi yang dilakukan gagal, maka kontroler akan mengembalikan respon error ke client.

1. Fungsi Logout

Fungsi logout digunakan untuk mengakhiri sesi login client dengan cara menghapus semua token yang dimiliki oleh client. Diagram dari fungsi logout dapat dilihat pada gambar x.x dibawah.

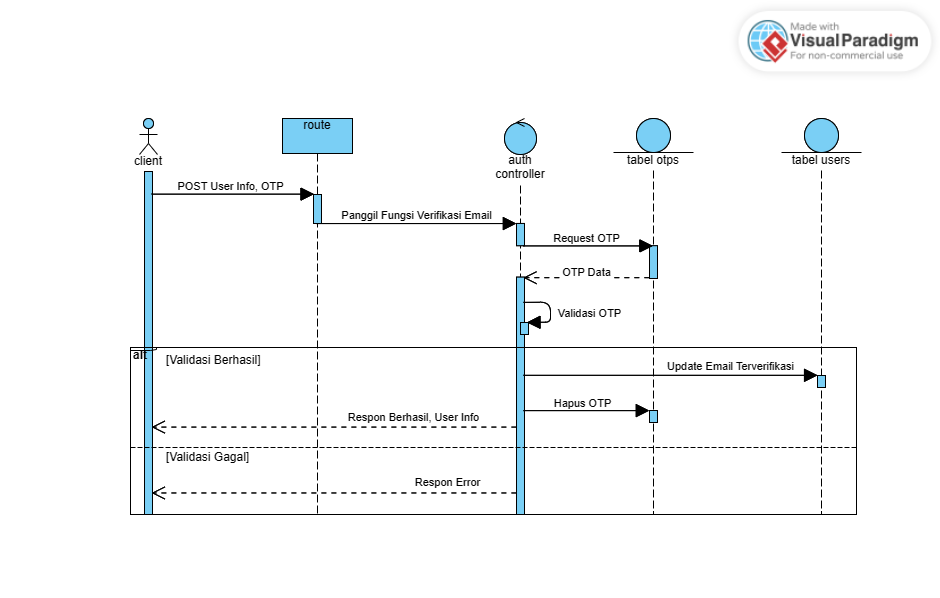


Sequence Diagram Logout

Pada gambar x.x diatas, fungsi logout didalam kontroler akan dipanggil oleh route jika ada client yang melakukan request ke endpoint “/logout”. Kemudian kontroler akan menghapus token dari client sesuai dengan token yang dilampirkan didalam header pada saat request diterima.

1. Fungsi Verifikasi Email

Fungsi verifikasi email digunakan untuk memastikan bahwa client memiliki email yang valid dan aktif hala ini akan meningkatkan keamanan dengan hanya mengijinkan client yang terpercaya untuk mengakses sumber daya yang ada. Diagram dari fungsi verifikasi email dapat dilihat pada gambar x.x dibawah.

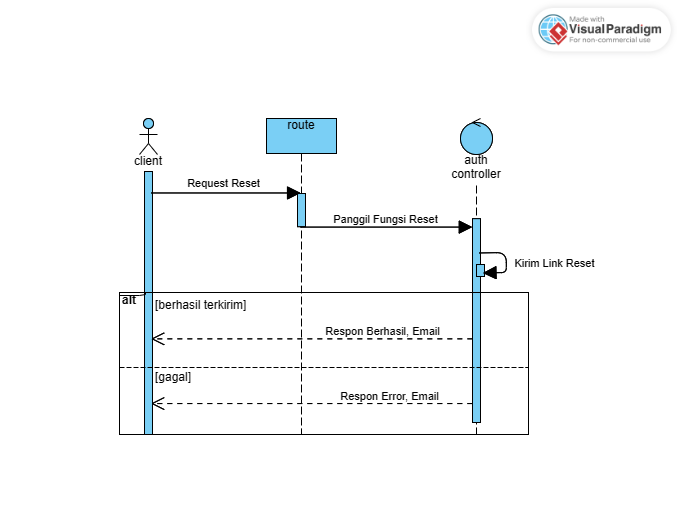


Sequence Diagram Verifikasi Email

Pada gambar x.x diatas dapat dilihat bahwa pengguna melakukan verifikasi email dengan mengirimkan kode OTP yang sudah diterima melalui email disertai dengan detail client seperti id, nama dan email ke endpoint “/verifikasi-email”, kemudian pada kontroler akan memeriksa kode yang diterima dengan kode yang tersimpan pada tabel otps, jika cocok dan masih aktif maka kontroler akan memperbarui status client menjadi terverifikasi dan menghapus kode otm yang lama kemudian mengembalikan respon berhasil. Jika kode salah tau sudah kadaluarsa maka kontroller akan mengembalikan respon error.

1. Fungsi Reset Password

Fungsi reset password digunakan oleh client untuk memperbarui password jika merakan lupa dengan cara memasukkan email yang sudah terdaftar dan menggunakan link yang telah dikirimkan ke email untuk memperbarui password. Diagram dari fungsi reset password dapat dilihat pada gambar x.x dibawah.



Sequence Diagram Reset Password

Pada gamabar x.x diatas dapat dilihat client memulai reset password dengan melakukan request ke endpoint “/reset-password” dengan mengirimkan email yang sudah terdaftar, kemudian kontroler akan memanggil fungsi kirim link reset yang sudah disediakan oleh Laravel dengan melampirkan email yang telah diterima, jika email berhasil terkirim maka kontroler akan mengembalikan respon berhasil dan jika email gagal terkirim atau email tidak ditemukan maka kontroler akan mengembarikan respon error.

1. Perancangan Komunikasi Websocket
2. Perancangan Server

BAB IV

**HASIL DAN PEMBAHASAN**

BAB V

**PENUTUP**

**5.1 Kesimpulan**

Berdasarkan hasil pengerjaan dan analisis yang telah dilakukan, dapat disimpulkan beberapa hal sebagai berikut:

## 5.2 Saran

Setelah dilakukan uji coba dan analisis didapatkan beberapa saran antara lain:

**DAFTAR PUSTAKA**

**BIODATA MAHASISWA**

|  |  |  |  |
| --- | --- | --- | --- |
|  | Nama | : | Muhammad Khoiril Wafi |
| NIM | : | 21060119140133 |
| Konsentrasi | : | Teknologi Informasi |
| Tempat Tanggal Lahir | : | Demak, 4 Maret 2001 |
| Alamat Sekarang | : | Jl. Baskoro Raya No. 61, RT.03 / RW.07, Tembalang, Kec. Tembalang, Kota Semarang, Jawa Tengah 50275 |
| Alamat E-mail | : | [khoirilwafi123@gmail.com](mailto:khoirilwafi123@gmail.com) |
| Nama Orang Tua | : | Muarifin |
| Alamat Orang Tua | : | Jl. Putrajaya No. 06 02/06 Desa Karangrejo Kec. Bonang Kab. Demak |
| IPK | : | - |

**Pengalaman dan Prestasi yang Pernah Diraih:**

|  |  |
| --- | --- |
| **Nama pengalaman / prestasi** | **Tahun** |
|  |  |
|  |  |
|  |  |

Semarang, 23 Juni 2023

(Muhammad Khoiril Wafi)

NIM. 21060119140133

**LAMPIRAN A**

**SENARAI PROGRAM**

**LAMPIRAN B**

**DATA TRAINNING**

**LAMPIRAN C**

**MAKALAH TUGAS AKHIR**