A red and black rectangular object with text

Description automatically generatedA picture containing icon

Description automatically generated

**JAVASCRIPT DAN SOLANA WEB3**

Tantangan Membangun Aplikasi di Era Disrupsi 4.0

Idam Fadilah

**Rolly Maulana Awangga**

**PANDUAN GOLANG DAN JAVASCRIPT DALAM PENGEMBANGAN WEB SERVICE PRESENSI**

Memahami cara panduan GoLang dengan Javascript dalam

pembuatan Web Service

**Farhan Rizki Maulana**

**Fahad Abdul Aziz**

**Nizar Abdul Kholiq**

**Ryaas Ishlah Ramadhan**

**M Iqbal Ramadhan**

|  |  |
| --- | --- |
| 2023 | **PT. Penerbit Buku Pedia**  **2023** |

**PANDUAN GOLANG DAN JAVASCRIPT DALAM PENGEMBANGAN WEB SERVICE PRESENSI**

Memahami cara panduan GoLang dengan Javascript dalam

pembuatan Web Service

***Penulis:***

Farhan Rizki Maulana

Fahad Abdul Aziz

Nizar Abdul Kholiq

Ryaas Ishlah Ramadhan

M Iqbal Ramadhan

***ISBN***: 978-623-8374-00-7

***Editor:***

Rolly Maulana Awangga

***Penyunting:***

Rolly Maulana Awangga

***Desain sampul dan Tata letak:***

Farhan Rizki Maulana

***Font:***

Calibri

***Penerbit:***

PT. Penerbit Buku Pedia

***Redaksi:***

Athena Residence Blok. E No. 1, Desa Ciwaruga,

Kec. Parongpong, Kab. Bandung Barat 40559

Tel. 628-775-2000-300

Email : penerbit@bukupedia.co.id

***Distributor:***

Informatics Research Center

Jl. Sariasih No. 54

Bandung 40151

Email : irc@ulbi.ac.id

Cetakan Pertama, 2023

Hak cipta dilindungi undang-undang

Dilarang memperbanyak karya tulis ini dalam bentuk dan

dengan cara apa pun tanpa ijin tertulis dari penerbit

**KATA PENGANTAR**

P

uji dan syukur ke hadirat Allah SWT, mari kita berdo’a diawali basmalah dan diakhiri dengan hamdalah berkat rahmat dan karunia-Nya. Penulis dapat menyelesaikan buku ini berjudul “Panduan Golang dan JavaScript dalam Pengembangan Web Service Presensi”. Selamat datang dalam panduan ini yang akan membahas pengembangan web service presensi menggunakan Golang dan JavaScript. Integrasi kedua bahasa pemrograman ini menjadi kunci untuk menciptakan solusi yang efektif dalam menghadirkan layanan presensi yang handal dan responsif.

Dalam panduan ini, kami akan membahas langkah-langkah praktis dalam mengintegrasikan Golang dan JavaScript. Golang akan digunakan sebagai bahasa pemrograman backend, mengatur logika bisnis dan menghubungkan dengan database MongoDB. Sementara itu, JavaScript akan digunakan untuk mengembangkan frontend yang interaktif.

Anda akan mempelajari konsep dasar Golang dan JavaScript, serta bagaimana menggabungkannya dengan baik dalam pembangunan web service presensi. Kami akan membahas pengaturan routing, manajemen database, pembuatan antarmuka pengguna yang menarik, dan komunikasi antara frontend dan backend.

Panduan ini ditujukan untuk pengembang yang ingin memperluas pengetahuan dan keterampilan mereka dalam mengembangkan web service presensi. Dengan memahami integrasi Golang dan JavaScript, Anda akan mampu menciptakan solusi presensi yang efisien dan sesuai dengan kebutuhan.

|  |
| --- |
| Bandung, Juli 2023 |
|  |
| Penulis |

**PRAKATA**

B

uku ini berisi panduan lengkap tentang pengembangan web service presensi menggunakan bahasa pemrograman Golang dan JavaScript. Buku ini ditujukan untuk para pengembang web yang ingin mempelajari cara mengembangkan web service presensi dengan menggunakan bahasa pemrograman Golang dan JavaScript.

Buku ini berisi data faktual dan statistik terpercaya tentang bahasa pemrograman Golang dan JavaScript, serta pentingnya penggunaan kedua bahasa pemrograman ini dalam pengembangan web service presensi. Selain itu, buku ini juga memberikan tujuan penulisan, pembaca sasaran, keunggulan buku, amanat/pesan untuk pembaca sasaran, tata cara membaca buku atau metode penyampaian setiap bab.

Buku ini membahas secara detail tentang penggunaan bahasa pemrograman Golang dan JavaScript dalam pengembangan web service presensi. Pembaca akan mempelajari cara menginstalasi Golang dan JavaScript, penggunaan variabel dan konstanta, fungsi print, tipe data dan operator pada Golang, serta struktur if, else, dan switch. Selain itu, pembaca juga akan mempelajari pengenalan array, slice, dan map, pengenalan perulangan for, penggunaan fungsi input, dan pembuatan fungsi sendiri.

Buku ini juga memberikan contoh-contoh penggunaan bahasa pemrograman Golang dan JavaScript dalam pengembangan web service presensi. Pembaca akan mempelajari cara mengembangkan aplikasi web menggunakan bahasa pemrograman Golang dan JavaScript, serta cara menggunakan bahasa pemrograman Golang dan JavaScript dalam membangun web service REST.

Buku ini ditulis dengan bahasa yang mudah dipahami dan disajikan dengan tata cara yang jelas dan terstruktur. Buku ini cocok untuk para pengembang web yang ingin mempelajari cara mengembangkan web service presensi dengan menggunakan bahasa pemrograman Golang dan JavaScript.

Setiap chapter atau bab yang disajikan dalam buku ini disertakan juga kode program yang bisa diakses melalui link: <https://github.com/Bukped/BukuPresensi>

**DAFTAR ISI**

**KATA PENGANTAR i**

**PRAKATA ii**

**DAFTAR ISI iii**

**BAB 1 PENGENALAN GOLANG, JAVASCRIPT, DAN MONGODB 1**

1. Pengenalan Golang 1
2. Fungsi Golang 2
3. Kelebihan Golang 2
4. Kekurangan Golang 3
5. Manfaat Golang 3
6. Pengenalan Javascript 5
7. Fungsi Javascript 5
8. Kelebihan Javascript 6
9. Kekurangan Javascript 7
10. Manfaat Javascript 8
11. Pengenalan MongoDB 9
12. Fungsi MongoDB 10
13. Kelebihan MongoDB 10
14. Kekurangan MongoDB 11
15. Manfaat MongoDB 12

**BAB 2 PETUNJUK INSTALASI GOLANG, JAVASCRIPT, DAN MONGODB 14**

1. Petunjuk Instalasi Golang 14
2. Instalasi Golang pada Sistem 15
3. Petunjuk Instalasi Javascript 15
4. Petunjuk Instalasi MongoDB 16

**BAB 3 KEBUTUHAN DAN LOGIKA DALAM PANDUAN GOLANG,**

**JAVASCRIPT DAN MONGODB 18**

1. Kebutuhan yang harus Di Penuhi 18

**BAB 4 PANDUAN CODING FULL STACK STUDI KASUS PRESENSI 22**

1. Membuat Struktur Folder dan File untuk Package Back-Emd 22
2. Library Iteung 28
3. Membuat Struktur Folder dan File untuk Package Back-End 45

**GLOSARIUM**

**INDEKS 66**

**PROFIL PENULIS 70**

**BAB 1**

**PENGENALAN GOLANG, JAVASCRIPT DAN MONGODB**

1. **Pengenalan Golang**

Pada tahun 2009, Google berkolaborasi dengan Ken Thompson, Robert Griesemer, dan Rob Pike untuk mengembangkan bahasa pemrograman Go, atau disingkat Golang. Tujuan pengembangannya adalah untuk menciptakan bahasa yang mendapat manfaat dari kecepatan, ketergantungan, skalabilitas, dan kesederhanaan.

Seiring dengan menghasilkan kode biner pada sistem yang dikompilasi, Golang juga merupakan bahasa yang dapat diketik secara statis. Bahasa pemrograman abad 21 Golang juga dikompilasi dari bahasa pemrograman C. Aplikasi, situs web, dan perangkat lunak lain semuanya dapat dibuat menggunakan bahasa pemrograman Go.

A cartoon character next to a blue text

Description automatically generated

Gambar 1.1. Logo Bahasa Golang

1. **Fungsi Golang**

Berikut adalah beberapa fungsi dari bahasa pemrograman Golang:

* + - 1. Mendukung pengembangan tim pengembang yang dapat berkembang secara skalabel.
      2. Memfasilitasi pengembangan teknologi penyimpanan online dengan kapasitas penyimpanan yang besar.
      3. Merancang aplikasi web dengan tingkat keamanan yang tinggi.
      4. Membangun sistem yang memiliki kinerja tinggi dan kompleksitas yang lebih tinggi.
      5. Mengembangkan kode server untuk jaringan web server dan layanan mikro.

1. **Kelebihan Golang**

Golang (Go) merupakan bahasa pemrograman yang dirancang oleh Google dengan fokus pada kecepatan eksekusi, efisiensi, serta kesederhanaan. Dalam pengembangan website Golang mempunyai sebagian kelebihan serta fitur utama yang buatnya jadi opsi yang terkenal.

* + - 1. Kinerja Besar Golang dirancang buat membagikan kinerja yang besar dengan waktu eksekusi yang kilat Bahasa ini memakai mekanisme konkurensi yang efektif ialah go routine, yang membolehkan pemrosesan paralel yang ringan serta efisien.
      2. Skalabilitas yang Baik: Dalam pengembangan aplikasi website yang memerlukan skalabilitas besar Golang mempunyai keahlian buat menanggulangi beban yang berat dengan baik. Keahlian konkurensi serta manajemen memori yang efektif membolehkan aplikasi website Golang buat menanggulangi banyak permintaan secara bertepatan seiring dengan performa yang senantiasa normal.
      3. Kode yang Gampang Dikelola: Golang mempunyai sintaks yang simpel serta gampang dimengerti Desain bahasa yang jelas serta fokus pada kesederhanaan membuat kode Golang lebih gampang dibaca, ditulis, serta dikelola oleh regu pengembang. Ini membolehkan pengembangan website jadi lebih efektif serta mempermudah kerja sama dalam regu.
      4. Pustaka Standar yang Kokoh Golang dilengkapi dengan pustaka standar yang kaya, tercantum sokongan buat pengembangan website Pustaka standar ini sediakan bermacam fitur serta perlengkapan yang mempermudah pengembangan aplikasi website semacam HTTP server, routing, enkripsi, serta banyak lagi.
      5. Komunitas yang Aktif: Golang mempunyai komunitas pengembang yang aktif serta tumbuh pesat. Komunitas ini membagikan sokongan sumber energi serta berbagi pengetahuan yang bermanfaat dalam pengembangan website dengan Golang. Ada pula bermacam pustaka serta kerangka kerja (framework) yang terbuat oleh komunitas buat memperluas fungsionalitas serta memesatkan pengembangan aplikasi website.

Keunggulan-keunggulan di atas menjadikan Golang selaku opsi yang menarik dalam pengembangan website paling utama buat aplikasi yang memerlukan performa besar skalabilitas, serta kemudahan pengelolaan kode. Dengan memakai Golang, Kamu bisa membangun aplikasi website yang tangguh serta efektif dengan waktu pengembangan yang lebih pendek.

1. **Kekurangan Golang**

Golang juga selain memiliki kelebihan juga ada kekurangannya, yang antara lain sebagai berikut:

* + - 1. Terlalu mendasar

Beberapa pemrogram mungkin menganggap fitur utama bahasa Go sebagai kelemahan. Go memiliki beberapa fungsi paling sederhana dan merupakan bahasa pemrograman yang paling mudah diakses. Namun, ini mengurangi keserbagunaan bahasa ini. Saat pemrogram mengerjakan proyek yang lebih besar, kesederhanaannya terkadang bisa menjadi tantangan.

* + - 1. Belum menjadi populer

Google merancang bahasa Go untuk menangani tugas-tugas khusus yang diperlukan oleh para pemrogramnya dengan efisiensi yang tinggi. Ini tentu bisa menjadi keuntungan jika Anda bekerja untuk Google. Namun untuk pemrograman yang lebih luas, bahasa Go masih kalah populer dibanding bahasa pemrograman lainnya. Berbeda dengan Javascript yang sudah dipercaya sebagai bahasa pemrograman front end atau Python yang banyak digunakan dalam dunia visualisasi dan analisis data.

* + - 1. Kurangnya Framework

Sebagai bahasa pemrograman yang tergolong baru, Golang masih memiliki keterbatasan jumlah framework yang dapat dimanfaatkan. Tidak seperti Python, Javascript, atau PHP, yang sudah memiliki banyak kerangka kerja untuk membantu pemrogram dalam pekerjaannya.

1. **Manfaat Golang**

Pemakaian Golang (Go) dalam pembuatan layanan website membagikan sebagian khasiat yang signifikan. Berikut merupakan sebagian khasiat utama yang bisa Kamu peroleh dengan memakai Golang dalam konteks pembuatan layanan website.

1. Kecepatan Eksekusi yang Besar Golang dikenal mempunyai waktu eksekusi yang sangat kilat Bahasa ini dirancang dengan fokus pada kinerja yang besar sehingga bisa menanggulangi permintaan website dengan kilat serta efektif Perihal ini membuat Golang jadi opsi yang baik buat layanan website yang memerlukan responsifitas serta waktu reaksi yang rendah.
2. Efisiensi dalam Pemakaian Sumber Energi Golang memakai mekanisme konkurensi yang efektif lewat goroutine serta channel. Dengan pendekatan ini, Golang bisa mengelola pemrosesan paralel dengan overhead yang rendah serta menggunakan sumber energi secara efektif Selaku hasilnya, aplikasi website yang dibentuk dengan Golang mempunyai mengkonsumsi memori yang rendah serta sanggup menanggulangi beban besar dengan baik.
3. Kesederhanaan dalam Penyusunan serta Pemeliharaan Kode: Golang mempunyai sintaks yang simpel serta gampang dimengerti Desain bahasa yang fokus pada kesederhanaan membuat kode Golang lebih gampang dibaca, ditulis, serta dikelola. Ini membolehkan regu pengembang buat bekerja dengan efektif memesatkan proses pengembangan, serta meminimalkan kesalahan yang bisa jadi terjalin.
4. Pustaka Standar yang Kokoh Golang dilengkapi dengan pustaka standar yang lengkap serta kokoh Pustaka standar ini mencakup bermacam fitur serta perlengkapan yang menunjang pengembangan layanan website semacam HTTP server, routing, enkripsi, pengelolaan session, serta banyak lagi. Dengan pustaka standar yang kaya ini, Kamu bisa membangun layanan website yang lengkap tanpa butuh tergantung pada pustaka pihak ketiga.
5. Kemudahan dalam Pengelolaan Ketergantungan: Golang mempunyai sistem manajemen materi yang terintegrasi, yang mempermudah pengelolaan ketergantungan ataupun dependensi pada pustaka eksternal. Kamu bisa dengan gampang meningkatkan menghapus, ataupun memperbarui materi yang dibutuhkan dalam proyek Kamu Perihal ini mempermudah dalam mengelola tipe pustaka serta membenarkan stabilitas serta keamanan aplikasi website Kamu.

Kemudahan dalam Pengelolaan Ketergantungan: Golang mempunyai sistem manajemen materi yang terintegrasi, yang mempermudah pengelolaan ketergantungan ataupun dependensi pada pustaka eksternal. Kamu bisa dengan gampang meningkatkan menghapus, ataupun memperbarui materi yang dibutuhkan dalam proyek Kamu Perihal ini mempermudah dalam mengelola tipe pustaka serta membenarkan stabilitas serta keamanan aplikasi website Kamu.

1. **Pengenalan Javascript**

JavaScript merupakan bahasa pemrograman yang digunakan oleh para pengembang untuk menciptakan halaman web yang interaktif. Dari kemampuannya untuk memperbarui umpan media sosial hingga menampilkan animasi dan peta interaktif, JavaScript menjadi kunci dalam meningkatkan pengalaman pengguna saat mengakses situs web.

Sebagai bahasa skrip yang berjalan di sisi klien, JavaScript menjadi salah satu teknologi inti yang membangun World Wide Web. Contohnya, saat Anda menjelajahi internet, setiap kali Anda melihat carousel gambar, menu dropdown yang dapat diklik untuk tampil, atau elemen yang berubah warna secara dinamis di halaman web, Anda melihat dampak langsung dari JavaScript.

A computer with a screen with text

Description automatically generated

Gambar 1.2. Javascript

1. **Fungsi Javascript**

JavaScript memiliki beberapa fungsi utama dalam pengembangan web, yaitu:

* + - 1. Interaksi Pengguna

JavaScript memungkinkan pembuatan elemen interaktif seperti tombol, formulir, dan menu dropdown.

* + - 1. Manipulasi Konten

JavaScript dapat digunakan untuk mengubah konten HTML dan CSS secara dinamis di halaman web.

* + - 1. Validasi Data

JavaScript memungkinkan validasi data yang dimasukkan oleh pengguna sebelum pengiriman ke server.

* + - 1. Animasi dan Efek Visual

JavaScript dapat menciptakan animasi dan efek visual menarik pada halaman web.

1. **Kelebihan Javascript**

JavaScript merupakan bahasa pemrograman yang sangat terkenal dalam pengembangan website serta mempunyai beberapa kelebihan serta fitur utama yang buatnya jadi opsi yang kokoh Berikut merupakan sebagian keunggulan serta fitur utama JavaScript.

1. Keahlian Melaksanakan di Sisi Klien (Client-Side): JavaScript bisa dieksekusi di sisi klien (client-side), yang berarti kode JavaScript dieksekusi oleh browser pengguna. Perihal ini membolehkan interaktivitas serta pengalaman pengguna yang kaya dalam aplikasi website JavaScript bisa digunakan buat mengganti konten taman secara dinamis, menanggulangi interaksi pengguna, serta memvalidasi input informasi.
2. Integrasi dengan HTML serta CSS: JavaScript secara natural terintegrasi dengan HTML serta CSS, membolehkan manipulasi elemen-elemen HTML, mengganti tampilan serta style taman dan mengendalikan sikap serta interaksi pengguna. Dengan JavaScript, Kamu bisa mengakses, memanipulasi, serta menggunakan elemen-elemen DOM (Document Object Model) buat membuat aplikasi website yang interaktif serta dinamis.
3. Kaya hendak Framework serta Library: JavaScript mempunyai ekosistem yang kaya dengan bermacam framework serta library yang mempermudah pengembangan website Contohnya, Kamu bisa memakai framework semacam React, Angular, ataupun Vue.js buat membangun aplikasi website yang lingkungan serta skala besar dengan memakai konsep komponen serta pembuatan antarmuka yang dikelola secara efektif.
4. Keahlian Asynchronous: JavaScript mempunyai sokongan buat pemrograman asynchronous lewat fitur semacam Promise serta async/await. Ini membolehkan pengembangan aplikasi website yang responsif serta efektif di mana pembedahan yang memerlukan waktu semacam pengambilan informasi dari server bisa dicoba secara asynchronous tanpa membatasi antarmuka pengguna.
5. Pembaruan serta Kenaikan yang Terus-Menerus: JavaScript merupakan bahasa yang terus tumbuh dengan komunitas yang besar serta aktif. Ini berarti kalau bahasa ini terus memperoleh pembaruan serta kenaikan fitur, dan sokongan buat standar website yang baru. Perihal ini membolehkan para pengembang buat senantiasa memakai metode terkini serta menggunakan pertumbuhan terbaru dalam pengembangan website.

Keunggulan serta fitur utama JavaScript, semacam keahlian melaksanakan di sisi klien, integrasi dengan HTML serta CSS, ketersediaan framework serta library, keahlian asynchronous, dan pembaruan serta kenaikan yang terus-menerus, buatnya jadi bahasa yang sangat bermanfaat serta kokoh dalam pengembangan aplikasi website.

1. **Kekurangan Javascript**

Setiap bahasa pasti memiliki kekurangan nya, Javascript juga memiliki kekurangan sebagai berikut:

* + - 1. Tidak Cocok untuk Aplikasi Standalone.

Aplikasi yang berjalan sendiri dikenal sebagai aplikasi standalone, dan seringkali dibuat hanya menggunakan satu bahasa pemrograman. Bahasa pemrograman C++ dan Java yang rumit, kaya objek, dan kaya fungsi adalah dua contoh yang sesuai untuk aplikasi mandiri. Anda hanya dapat membuat program menggunakan dua bahasa tersebut: C++ dan Java. JavaScript harus digunakan bersamaan dengan bahasa pemrograman lain. HTML dan CSS adalah dua bahasa pemrograman yang paling sering digunakan dengan JS.

* + - 1. Objek Terbatas

Kekurangan selanjutnya dari JavaScript adalah objeknya yang terbatas. Karena itu, JavaScript cukup sederhana dan tidak memiliki banyak fitur di dalamnya. Sifat umum JS membuatnya tidak cocok untuk aplikasi kompleks yang berjalan tunggal.

* + - 1. Keamanan Rendah

Keamanan adalah area lain di mana JavaScript gagal. Ini karena banyak JavaScript dijalankan di sisi klien, sehingga sulit untuk melacak peretas. Bahasa pemrograman ini juga memiliki kemampuan enkripsi yang buruk. Ini membuat kode dan data di aplikasi Anda terbuka untuk umum dan dapat dieksploitasi.

1. **Manfaat Javascript**

Pemakaian JavaScript dalam pengembangan website mempunyai beberapa manfaat yang signifikan. Berikut merupakan sebagian manfaat utama dari pemakaian JavaScript

1. Interaktivitas serta Responsivitas: JavaScript membolehkan Kamu buat membuat aplikasi website yang interaktif serta responsif. Kamu bisa mengendalikan sikap serta interaksi pengguna, menanggulangi validasi input informasi dan merespons pergantian yang terjalin dalam taman website secara dinamis. Dengan JavaScript, Kamu bisa membagikan pengalaman pengguna yang lebih menarik serta responsif.
2. Validasi Informasi di Sisi Klien: JavaScript membolehkan Kamu buat melaksanakan validasi informasi di sisi klien saat sebelum informasi dikirim ke server. Perihal ini kurangi beban server dengan mengecek serta memvalidasi informasi secara langsung di browser pengguna. Dengan validasi informasi di sisi klien, Kamu bisa membagikan umpan balik kilat kepada pengguna tentang kesalahan input saat sebelum informasi dikirim ke server.
3. Manipulasi serta Pengendalian DOM: JavaScript membolehkan Kamu buat mengakses serta memanipulasi elemen-elemen DOM (Document Object Model) dalam taman website Kamu bisa mengganti konten, style serta sikap elemen-elemen HTML, dan menanggulangi peristiwa (event) yang terjalin di taman Dengan keahlian ini, Kamu bisa membuat tampilan serta antarmuka pengguna yang dinamis, dan tingkatkan interaktivitas taman website.
4. Integrasi dengan Layanan Website serta API: JavaScript membolehkan integrasi yang gampang dengan layanan website serta API. Dengan memakai metode semacam AJAX (Asynchronous JavaScript and XML) ataupun Fetch API, Kamu bisa mengambil informasi dari server secara asynchronous serta mengirim informasi ke server tanpa butuh muat ulang segala taman Perihal ini membolehkan pengembangan aplikasi website yang lebih kilat serta responsif.
5. Ekosistem Framework serta Library yang Kokoh JavaScript mempunyai ekosistem yang kaya dengan bermacam framework serta library yang bisa memesatkan pengembangan aplikasi website Contohnya, framework semacam React, Angular, ataupun Vue.js sediakan perlengkapan serta konsep yang kokoh buat membangun antarmuka pengguna yang lingkungan serta interaktif. Dengan memakai framework serta library tersebut, Kamu bisa mengirit waktu serta usaha dalam pengembangan aplikasi website.
6. Dukungan Cross-platform: JavaScript bisa dijalankan di bermacam platform serta fitur tercantum desktop, fitur seluler, serta browser website yang berbeda. Perihal ini membagikan fleksibilitas dalam meningkatkan aplikasi website yang bisa diakses oleh pengguna di bermacam platform.

Serta fitur Konsumsi JavaScript dalam pengembangan website membagikan khasiat signifikan dalam tingkatkan interaktivitas, responsivitas, validasi informasi di sisi klien, integrasi dengan layanan website serta kemudahan pemakaian framework serta library. JavaScript sudah jadi bahasa pemrograman yang berarti dalam membangun aplikasi website modern yang interaktif serta kokoh.

1. **Pengenalan MongoDB**

MongoDB adalah jenis database NoSQL yang menggunakan format JSON (JavaScript Object Notation) untuk menyimpan data dalam bentuk dokumen, berbeda dengan database SQL yang menggunakan tabel. MongoDB umumnya digunakan untuk mengelola volume data yang besar, seperti big data, dengan arsitektur yang berbeda dari database SQL. Karena alasan ini, sangatlah penting untuk dengan cermat mempertimbangkan teknologi database yang akan digunakan dalam proyek atau bisnis.

A green and black background with text

Description automatically generated

Gambar 1.3. MongoDB

1. **Fungsi MongoDB**

Berikut adalah beberapa fungsi MongoDB sebagai berikut:

* + - 1. Penyimpanan Fleksibel

MongoDB memungkinkan penggunaan struktur data yang berbeda tanpa batasan skema yang kaku karena data disimpan dalam format dokumen JSON yang fleksibel.

* + - 1. Skalabilitas Horizontal

MongoDB dirancang untuk dengan mudah diperluas dan dapat mengelola volume data yang besar dengan menggunakan replikasi dan partisi otomatis.

* + - 1. Querying yang Kuat

MongoDB menyediakan fitur querying yang kuat, termasuk indeks, operasi agregasi, dan kemampuan untuk melakukan query yang fleksibel pada data.

1. **Kelebihan MongoDB**

Penggunaan MongoDB sebagai basis data dalam pengembangan aplikasi web memiliki beberapa keunggulan dan fitur utama. Berikut adalah beberapa di antaranya:

1. Fleksibilitas Skema: MongoDB menggunakan pendekatan database NoSQL yang memungkinkan fleksibilitas dalam skema data. Tidak perlu mendefinisikan skema tabel sebelumnya seperti dalam basis data relasional. Ini memungkinkan penambahan, perubahan, dan penghapusan kolom dalam dokumen dengan mudah, tanpa mempengaruhi dokumen lainnya. Fleksibilitas ini berguna dalam menghadapi perubahan kebutuhan data dalam pengembangan aplikasi.
2. Skalabilitas Horisontal: MongoDB dirancang untuk mendukung skalabilitas horisontal dengan mudah. Anda dapat dengan cepat menambahkan server tambahan dan mendistribusikan beban kerja di antara mereka untuk meningkatkan kinerja dan kapasitas. Dengan pendekatan ini, Anda dapat mengatasi pertumbuhan data yang besar dan meningkatkan throughput aplikasi dengan mudah.
3. Kueri Kuat dan Ekspresif: MongoDB menyediakan kueri yang kuat dan ekspresif menggunakan bahasa kueri MongoDB sendiri yang disebut Query Language (MQL). MQL mendukung berbagai operator dan fungsi untuk melakukan pencarian, penyaringan, pengurutan, dan agregasi data dengan fleksibilitas tinggi. Anda dapat dengan mudah melakukan kueri kompleks dan mencari data dengan cara yang lebih intuitif.
4. Replikasi dan Ketahanan Terhadap Kegagalan: MongoDB mendukung replikasi, yang memungkinkan pembuatan salinan data di beberapa server yang berbeda. Ini meningkatkan ketersediaan data dan ketahanan terhadap kegagalan. Jika ada masalah pada salah satu server, replikasi memastikan data tetap dapat diakses melalui server lainnya.
5. Indeks yang Kuat: MongoDB menyediakan fitur indeks yang kuat untuk meningkatkan kinerja pencarian dan pengurutan data. Anda dapat membuat indeks pada satu atau beberapa kolom dalam koleksi dokumen, yang memungkinkan akses data lebih cepat. Indeks juga membantu mengoptimalkan kueri yang kompleks dan mempercepat operasi pengolahan data.
6. Skalabilitas Vertikal: Selain skalabilitas horisontal, MongoDB juga mendukung skalabilitas vertikal. Anda dapat meningkatkan kinerja server dengan menambahkan sumber daya (misalnya, RAM atau CPU) pada server tunggal. Ini memungkinkan Anda menangani beban kerja yang lebih besar pada server yang ada tanpa perlu menambahkan server tambahan.

Penggunaan MongoDB sebagai basis data dalam pengembangan aplikasi web memberikan keunggulan dalam fleksibilitas skema, skalabilitas horisontal dan vertikal, kueri yang kuat, replikasi data, indeks yang kuat, dan ketahanan terhadap kegagalan. Fitur-fitur ini membuat MongoDB menjadi pilihan populer untuk mengelola data dalam aplikasi web modern.

1. **Kekurangan MongoDB**

Berikut adalah kekurangan MongoDB sebagai berikut:

* + - 1. Kurangnya konsistensi data

Karena MongoDB adalah sistem database yang fleksibel, konsistensi data tidak dapat diantisipasi darinya. Basis data MongoDB bukanlah pilihan terbaik jika Anda ingin mengelola dan bekerja dengan data secara teratur.

* + - 1. Fungsionalitas SQL minimal

MongoDB, basis data berbasis NoSQL, tidak secara eksplisit mendukung SQL. Anda mungkin perlu terbiasa menggunakan MongoDB jika Anda terbiasa menggunakan kueri SQL dan bergantung pada kemampuan khusus SQL.

* + - 1. Penggunaan memori tinggi

Prosedur administrasi database untuk MongoDB menggunakan memori. Saat memproses volume informasi data yang signifikan, ini dapat mengakibatkan penggunaan memori yang cukup besar. Jika Anda ingin memanfaatkan MongoDB, disarankan agar Anda menggunakan server berkinerja tinggi, seperti VPS.

1. **Manfaat MongoDB**

Menggunakan MongoDB dalam pengembangan aplikasi web memberikan sejumlah manfaat yang signifikan. Berikut adalah beberapa keuntungan utama dari penggunaan MongoDB:

1. Fleksibilitas Skema: MongoDB mengadopsi pendekatan NoSQL yang memungkinkan penggunaan skema data yang dinamis. Anda tidak perlu menentukan skema tabel terlebih dahulu, sehingga memungkinkan penambahan, perubahan, dan penghapusan field dalam dokumen tanpa mempengaruhi dokumen lainnya. Ini memberikan fleksibilitas dalam menghadapi perubahan kebutuhan data selama proses pengembangan aplikasi.
2. Skalabilitas: MongoDB didesain untuk mengatasi pertumbuhan data yang besar dan mendukung skalabilitas secara horizontal. Anda dapat dengan mudah menambahkan server tambahan dan mendistribusikan beban kerja di antara mereka. Hal ini memungkinkan aplikasi untuk mengatasi peningkatan volume data dan lalu lintas dengan efisiensi yang tinggi.
3. Performa Tinggi: MongoDB memiliki performa yang tinggi dalam operasi penulisan dan pembacaan data. Dengan desain yang dioptimalkan dan fitur-fitur seperti indeks yang kuat, MongoDB mampu memberikan kinerja yang baik dalam mengakses dan memanipulasi data. Hal ini sangat penting untuk menjaga responsivitas aplikasi web dan memberikan pengalaman pengguna yang lancar.
4. Distribusi Data dan Skala Global: MongoDB memungkinkan distribusi data secara geografis melalui replikasi dan sharding. Ini memungkinkan data disimpan secara terdistribusi di berbagai lokasi, memungkinkan aplikasi beroperasi secara efisien dalam skala global dengan mengurangi latensi jaringan dan meningkatkan ketersediaan data.
5. Kueri yang Kuat dan Ekspresif: MongoDB menyediakan bahasa kueri yang kuat dan ekspresif yang memungkinkan pengguna untuk melakukan pencarian, penyaringan, pengurutan, dan agregasi data dengan fleksibilitas tinggi. Pengguna dapat dengan mudah membuat kueri yang kompleks dan melakukan manipulasi data sesuai dengan kebutuhan aplikasi.
6. Komunitas Aktif dan Ekosistem yang Kuat: MongoDB memiliki komunitas pengguna yang besar dan aktif, serta ekosistem alat dan kerangka kerja yang kuat. Hal ini memberikan akses ke berbagai sumber daya, dokumentasi, tutorial, dan dukungan yang membantu pengembang dalam menghadapi tantangan dan mempercepat proses pengembangan aplikasi.

Menggunakan MongoDB dalam pengembangan aplikasi web memberikan keuntungan seperti fleksibilitas skema, skalabilitas, performa tinggi, distribusi data, kemampuan kueri yang kuat, serta dukungan dari komunitas dan ekosistem yang kuat. Ini menjadikan MongoDB sebagai pilihan populer dalam mengelola data dalam konteks aplikasi web modern.

**BAB 2**

**PETUNJUK INSTALASI GOLANG, JAVASCRIPT, DAN MONGODB**

1. **Petunjuk Instalasi Golang**
   * + 1. Instalasi Golang

Kunjungi website resmi GoLang (https://go.dev/) dan ikuti langkah-langkah berikut ini:

A screenshot of a computer

Description automatically generated

Gambar 2.1. Website Golang

Setelah mengunjungi website resmi GoLang, di halaman resmi nya terdapat beberapa opsi untuk unduhan yang tersedia.

* + - 1. Pilih Sistem Operasi

Kemudian, pilihlah sistem operasi yang cocok dengan yang Anda gunakan. Misalnya anda menggunakan sistem operasi Windows 64-bit, pilih yang dengan ekstensi ".msi" untuk windows 64-bit.

1. **Instalasi Golang pada Sistem**

Setelah anda mengunduh GoLang dengan sesuai sistem operasi yang digunakan, selanjunya adalah menginstall GoLang pada sistem operasi tersebut. Berikut ini adalah langkah-langkah untuk menginstall GoLang pada sistem operasi:

* + - 1. Instalasi Golang di Windows

Buka file instalasi GoLang yang sudah anda unduh, dengan ekstensi ".msi".

A screenshot of a computer program

Description automatically generated

Gambar 2.1. Setup Golang

Ikuti saja instalasi dengan Klik Next dan anda akan memilih direktori instalasi GoLang secara default atau anda ingin mengubahnya sesuai keinginan anda. Setelah itu tunggu saja sampai instalasi berhasil.

1. **Petunjuk Instalasi Javascript**

JavaScript tidak perlu diinstal secara terpisah karena JavaScript adalah bahasa pemrograman yang sudah terintegrasi di dalam setiap browser web modern. JavaScript adalah sebuah bahasa pemrograman yang dieksekusi oleh sisi klien. (client-side) dan tidak memerlukan instalasi karena sudah terintegrasi dengan browser web. JavaScript dapat dijalankan pada browser web seperti Google Chrome, Mozilla Firefox, Microsoft Edge, dan lainnya. Namun, untuk menjalankan JavaScript pada browser web, pengguna harus mengaktifkan JavaScript pada pengaturan browser web yang digunakan. Jika Javascript telah dinonaktifkan dalam browser anda konten atau fungsi dari halaman web dapat dibatasi atau tidak tersedia.

A computer screen with writing tools and a pen

Description automatically generated

Gambar 2.2. Javascript tidak perlu di install.

1. **Petunjuk Instalasi MongoDB**
   * + 1. Instalasi MongoDB

Kunjungi website resmi MongoDB dan ikuti langkah-langkah berikut: (<https://www.mongodb.com/try/download/community>)

A screenshot of a computer

Description automatically generated

Gambar 2.3. Website MongoDB

Setelah mengunjungi website resmi MongoDB, di halaman resmi nya terdapat beberapa opsi untuk unduhan yang tersedia.

* + - 1. Pilih Sistem Operasi

Selanjutnya pilih sistem operasi yang sesuai dengan yang anda gunakan. Misalnya anda menggunakan sistem operasi Windows 64-bit, pilih yang dengan package ".msi" untuk windows 64-bit.

* + - 1. Lanjutkan Instalasi

Setelah unduh langsung ikuti saja seperti biasa anda menginstall dan pilih direktori yang sesuai atau defaultnya saja.

**BAB 3**

**KEBUTUHAN DAN LOGIKA DALAM PANDUAN GOLANG, JAVASCRIPT DAN MONGODB**

1. **Kebutuhan yang harus Di Penuhi**

Ketika memasuki bab ketiga dari buku ini, fokus utama kita adalah kebutuhan dan logika yang terlibat dalam panduan ini, yang melibatkan bahasa pemrograman Golang, JavaScript, dan MongoDB. Bagian ini akan membahas persyaratan yang harus dipenuhi agar dapat mengikuti pembuatan web-service menarik ini. Tujuan utama buku ini adalah untuk mengembangkan kemampuan kita dalam dua hal. Pertama, kita akan mempelajari cara mengambil data dari API yang kita buat dan menampilkan data tersebut secara dinamis di halaman index.html. Kedua, kita akan belajar bagaimana mengirim pesan chat ke database dan menyimpannya.

* + - 1. Akun Github

Membuat akun di GitHub memiliki peran penting dalam pembuatan proyek, karena memungkinkan kita untuk menyimpan proyek di sana dan memanfaatkan berbagai fitur yang disediakan oleh GitHub. GitHub juga menyediakan banyak platform untuk melakukan deploy baik Back-End maupun Front-End. Misalnya, GitHub Pages adalah fitur yang disediakan secara gratis oleh GitHub untuk melakukan deploy Front-End.

Selain itu, ada juga platform lain seperti 000Webhost, Netlify, dan Heroku yang terintegrasi dengan GitHub. Netlify digunakan untuk deploy atau mempublikasikan website statis, sementara Heroku digunakan untuk publikasi website dinamis. Untuk mendaftar, anda dapat mengunjungi website resmi GitHub. Pastikan kamu sudah menguasai git sebelumnya.



Gambar 3.1. Logo Github

* + - 1. Akun Heroku

Langkah selanjutnya adalah membuat akun di Heroku untuk mendeploy aplikasi Back-End Golang. Heroku adalah platform yang mempermudah publikasi aplikasi Back-End dengan mudah. Meskipun ada biaya terkait, kita dapat memanfaatkan akun akademik atau akun yang diberikan oleh kampus untuk mendapatkan manfaat tersebut. Beberapa kampus bahkan menyediakan alokasi dana atau saldo di Heroku yang dapat digunakan untuk mendeploy aplikasi Back-End.

Mendaftar di situs resmi Heroku atau melalui tautan yang disediakan dalam buku memungkinkan kita mengelola dan mendeploy aplikasi Back-End dengan mudah. Dengan akun Heroku, kita dapat mempublikasikan proyek dan membuatnya dapat diakses secara online oleh orang lain. Jadi, jangan ragu untuk mendaftar dan menjelajahi potensi yang ditawarkan oleh Heroku dalam pengembangan aplikasi web.

A purple and black logo

Description automatically generated

Gambar 3.2. Logo Heroku

* + - 1. Akun MongoDB

Langkah selanjutnya adalah membuat akun di MongoDB Atlas dan menginstal MongoDB sebagai tempat penyimpanan database. MongoDB Atlas menyediakan opsi gratis untuk pengguna, sehingga tidak diperlukan biaya tambahan. Kamu dapat membuat akun melalui tautan MongoDB Atlas atau situs resmi MongoDB.

Setelah memiliki akun, kamu dapat dengan mudah membuat dan mengelola database aplikasi. Selanjutnya, kamu perlu mengunduh dan menginstal MongoDB di komputer. Gunakan tautan MongoDB Compass Download atau situs resmi MongoDB untuk mendapatkan MongoDB Compass, antarmuka grafis yang memudahkan pengelolaan database MongoDB. Setelah memiliki akun MongoDB Atlas dan menginstal MongoDB, kamu siap melanjutkan ke tahap berikutnya dalam proyek ini.

A green and black background with text

Description automatically generated

Gambar 3.3. Logo MongoDB

* + - 1. Backend Golang

Pertama-tama, mari kita bahas tentang Golang, bahasa pemrograman yang akan kita gunakan sebagai Back-End dalam aplikasi kita. Golang, atau lebih dikenal sebagai Go, adalah bahasa pemrograman yang sangat kuat dan cepat untuk pengembangan Back-End. Keunggulan utama Go adalah kemampuannya untuk mengkonversi kode program menjadi biner, memberikan kecepatan eksekusi yang superior dibandingkan dengan bahasa lain.

Selanjutnya, kita akan membuat kode program dalam paket Back-End yang akan menjadi inti dari aplikasi kita. Kode program ini akan dipanggil dan digunakan dalam library bernama "Iteung". Jika kamu ingin mengecek lebih lanjut tentang library ini, kamu bisa mengunjungi tautan berikut: [aiteung/iteung](https://github.com/aiteung/iteung) di GitHub. Library ini merupakan boilerplate yang berguna untuk menggunakan paket Iteung.

Dengan menggunakan Golang dan library Iteung, kita akan memiliki fondasi yang kuat dalam membangun aplikasi kita. Golang memberikan performa yang optimal dan memungkinkan kita untuk mengoptimalkan waktu eksekusi. Jadi, mari kita lanjutkan perjalanan ini dan memanfaatkan kekuatan Golang dalam pengembangan Back-End yang tangguh dan efisien.

* + - 1. Frontend Javascript

Selanjutnya, mari kita bahas tentang JavaScript yang akan menjadi bahasa pemrograman yang digunakan untuk mengembangkan bagian Front-End dari aplikasi kita. JavaScript akan berperan penting dalam menghubungkan dan mengolah data dari API yang telah kita deploy di bagian Back-End. Tujuan utama kita adalah mengambil data dari API menggunakan fungsi-fungsi JavaScript dan juga melakukan pengiriman data melalui metode POST.

Dalam pengembangan Front-End ini, kita akan memanfaatkan library bernama JSCroot. Library ini merupakan sebuah VanillaJS Micro Framework yang tersedia di GitHub. Kamu dapat melihatnya dengan mengunjungi tautan berikut: [jscroot/jscroot.github.io.](https://github.com/jscroot/jscroot.github.io) Dengan menggunakan library ini, kita akan mendapatkan bantuan yang berharga dalam membangun fungsi-fungsi JavaScript yang diperlukan dalam aplikasi kita.

**BAB 4**

**PANDUAN CODING FULL STACK STUDI KASUS PRESENSI**

1. **Membuat Struktur Folder dan File untuk Package Back-End**

Selanjutnya, kita akan memasuki tahap inti dalam pembuatan aplikasi ini, yaitu tahap praktik pengkodean. Pada tahap ini, kita akan mengintegrasikan Golang dengan JavaScript. Tahap pertama yang akan kita lakukan adalah pembuatan Back-End menggunakan Golang untuk aplikasi kita.

* + - 1. Struktur Back-End Golang

Pada tahap ini, kita akan membuat dan memahami struktur kode yang akan kita buat, yaitu package Back-End. Tahap ini memiliki tingkat penting yang cukup tinggi, karena akan membentuk dasar dari aplikasi kita. Mari kita lanjutkan dengan melihat struktur file yang akan digunakan.

A screenshot of a computer

Description automatically generated

Gambar 4.1. Struktur Back-End

type.go

File yang akan kita gunakan di sini adalah struktur data atau struct yang akan mewakili tabel dan atribut dalam basis data aplikasi kita. Dalam kata lain, kita akan menggunakan struct ini sebagai representasi dari tabel dan atribut yang akan menjadi database aplikasi kita.

rizki.go

File "rizki.go" akan berisi fungsi-fungsi yang akan memungkinkan akses data dari database kita melalui API atau untuk dideploy ke server online. Kita akan mulai dengan membuat fungsi Get dan Post untuk berinteraksi dengan database kita. Dalam studi kasus ini, kita akan menggunakan MongoDB sebagai database kita.

rizki\_test.go

File "rizki\_test.go" akan digunakan sebagai tester untuk fungsi-fungsi yang telah kita buat dalam file "rizki\_test.go". Tester ini akan digunakan untuk memeriksa apakah fungsi-fungsi tersebut berjalan dengan benar atau tidak. Peran tester sangat penting karena tanpa adanya tester, kita tidak akan tahu apakah fungsi atau kode yang kita buat berfungsi dengan baik atau tidak.

* + - 1. Merancang Struct

Sekarang, kita akan memasuki tahap pembuatan struct untuk studi kasus kita mengenai presensi. Sebelumnya, kita perlu menjalankan perintah "go mod init" di terminal Visual Studio Code untuk menginisialisasi dependensi kita. Perintah ini akan mengatur dependensi dengan nama repositori GitHub yang telah kita buat untuk menyimpan file Back-End. Selanjutnya, kita akan melakukan kompilasi dependensi dengan menjalankan perintah "go mod tidy" di terminal. Setelah itu, kita dapat membuat struct seperti yang ditunjukkan dalam contoh gambar di bawah ini.

A screenshot of a computer program

Description automatically generated

Gambar 4.2. Struct Back-End

Dalam kode di atas, Anda dapat mengganti nama package sesuai keinginan Anda. Untuk struct, Anda dapat menggunakan contoh kode di atas atau mengubahnya sesuai kebutuhan Anda.

Paket "go.mongodb.org/mongo-driver/bson/primitive" diimpor untuk bekerja dengan BSON (Binary JSON), format dokumen yang digunakan dalam MongoDB. Dalam paket ini, kami mengimpor tipe data "primitive.ObjectID" yang digunakan sebagai tipe data unik untuk mengidentifikasi dokumen dalam MongoDB.

* + - 1. Merancang Function

Berikut ini adalah keseluruhan kode file "rizki.go" yang akan berisi semua fungsi-fungsi yang diperlukan. Dalam contoh ini, kami akan membuat dua fungsi, yaitu "Insert" dan "Post", untuk mengakses database kita. Anda dapat menyesuaikan kode di atas dengan kebutuhan Anda dan menambahkan fungsi-fungsi lain seperti "Get", "Update", atau "Delete" sesuai dengan logika dan kebutuhan aplikasi Anda

A screen shot of a computer program

Description automatically generated

Gambar 4.3. Function Back-End

* + - 1. Merancang Function Test

Setelah kita membuat fungsi-fungsi dalam file "rizki.go", langkah selanjutnya adalah membuat unit test untuk menguji kode tersebut. Berikut adalah contoh kode untuk file "rizki\_test.go":

A screen shot of a computer program

Description automatically generated

Gambar 4.4. Function Test Back-End.

* + - 1. Lakukan Push dan Membuat Tag di Github

Setelah kita telah membuat struktur, fungsi, dan kode pengujian, langkah selanjutnya adalah membuat tag untuk merujuk pada versi kode tersebut. Tag ini akan digunakan dalam library Iteung yang akan kita gunakan. Berikut adalah langkah-langkahnya:

Di terminal Visual Studio Code, jalankan perintah `git tag` untuk melihat daftar tag yang ada (jika ada).

Buat tag baru dengan perintah `git tag v0.0.1`, di mana `v0.0.1` adalah versi tag yang ingin Anda gunakan. Anda dapat mengganti angka dan huruf sesuai dengan keinginan Anda.

Push tag yang baru saja dibuat dengan perintah `git push origin --tags`. Hal ini akan mengirimkan tag ke repositori yang terhubung.

Jalankan perintah `go list -m github.com/username/namarepo@v0.0.1` untuk memverifikasi bahwa versi tag Anda telah terdaftar.

Selanjutnya, masuk ke repositori library Iteung dan jalankan perintah `go get github.com/usernamegithub/namarepo@v0.0.1` (ganti dengan versi tag yang Anda buat). Hal ini akan mengunduh dan menginstal library dengan versi tag yang ditentukan.

Setelah Anda melakukan langkah-langkah di atas, tag Anda akan tersedia untuk digunakan dalam library Iteung. Jangan lupa untuk melakukan push kode seperti biasa setelah membuat tag.

Dengan melakukan langkah-langkah ini, Anda dapat memastikan bahwa versi kode yang tepat digunakan dalam pengembangan aplikasi dengan menggunakan library Iteung.

* + - 1. Merancang Function dan Struct Gocroot Back-End.

Dalam pengembangan menggunakan Gocroot sebagai Back-End, langkah pertama yang perlu dilakukan adalah melakukan fork terhadap repositori Gocroot di GitHub masing-masing. Anda dapat mengakses repositori Gocroot melalui tautan berikut: https://github.com/gocroot. Setelah melakukan fork, langkah selanjutnya adalah menambahkan struct di dalam folder "model" dan function di dalam folder "module".

A screenshot of a computer program

Description automatically generated

Gambar 4.6. Function Module

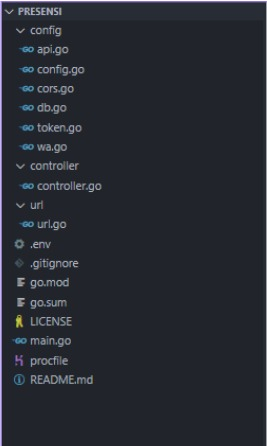
Langkah selanjutnya adalah memasukkan kode struct ke dalam folder "model" dengan memastikan bahwa nama package sesuai dengan nama folder tersebut.

A screenshot of a computer program

Description automatically generated

Gambar 4.7. Function Model

1. **Library Iteung**
   * + 1. Struktur Library Iteung



Gambar 4.8. Library Iteung

Dalam pengembangan aplikasi back-end, terdapat folder-folder penting seperti "Config", "Controller", dan "URL" yang memiliki peran masing-masing. Folder "Config" digunakan untuk mengatur konfigurasi aplikasi, seperti CORS, database, token, dan API. Folder "Controller" digunakan untuk membuat kontroler atau memanggil fungsi-fungsi dari modul "gocroot".

Folder "URL" berisi file-file yang digunakan untuk membuat dan memanggil kontroler sebagai API untuk aplikasi back-end. Anda dapat menggunakan repositori yang disediakan dan melakukan penyesuaian manual sesuai kebutuhan. Pastikan untuk mengatur dependensi dengan benar menggunakan perintah "go mod init" dan memperbarui dependensi dengan "go mod tidy".

Jika ada kesalahan, perbaiki dengan memastikan impor modul yang tepat. Dengan demikian, pengembangan aplikasi back-end dapat dilakukan dengan lebih mudah dan efisien.



Gambar 4.9. Import

Jika Anda mengalami kesalahan saat mengimpor package model dan module dari repositori kita, gunakan perintah "go get" untuk mengimpor package tersebut.

Misalnya, "go get github.com/gocroot/kampus/model" atau "go get github.com/gocroot/kampus/module".

Setelah itu, atur dependensi dengan "go mod init" dan alias model dan modul untuk penggunaan yang lebih mudah. Dengan langkah-langkah ini, Anda dapat mengatasi kesalahan impor dan mengimpor package yang diperlukan.

Anda dapat melihat library yang disebutkan di atas di repositori saya atau langsung di repositori iteung. Silakan cek link repositori berikut untuk informasi lebih lanjut:

Repo presensi:

https://github.com/FarhanRizkiM/presensi

Repo iteung:

<https://github.com/aiteung/iteung>

* + - 1. Konfigurasi Database

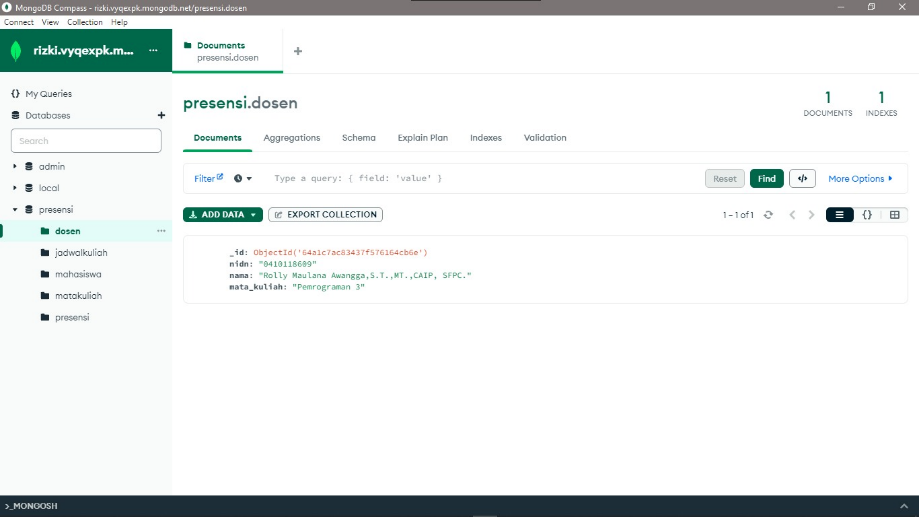
A screenshot of a computer program

Description automatically generated

Gambar 4.10. Konfigurasi Database

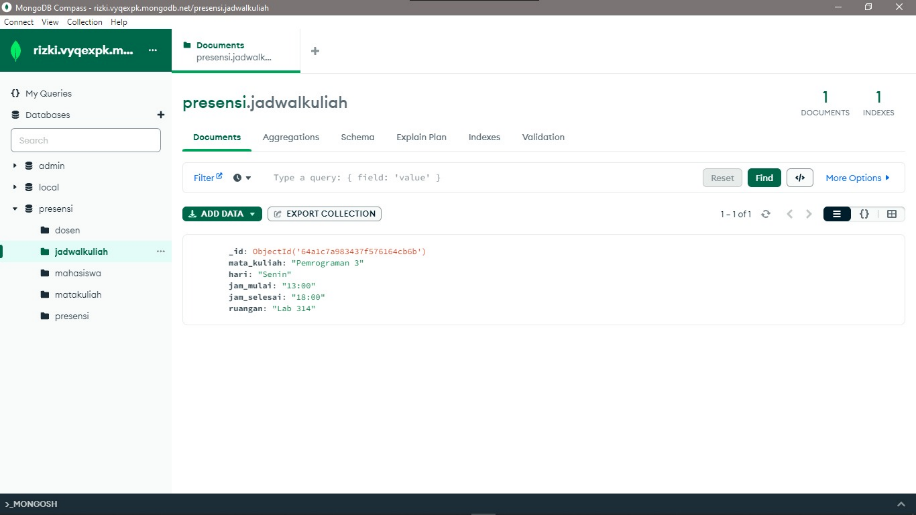
Tampilan Database di dalam MongoDB seperti gambar di bawah ini:

* Dosen



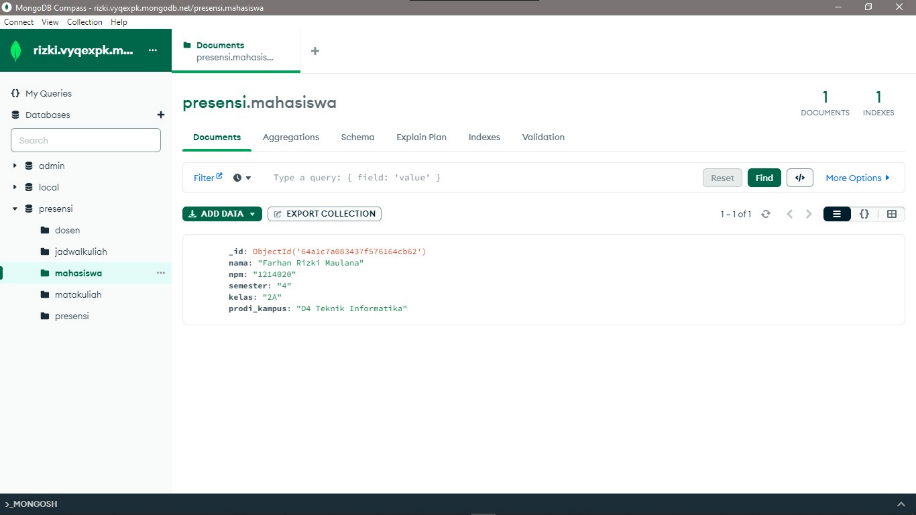
Gambar 4.11. MongoDB Dosen

* Jadwal Kuliah



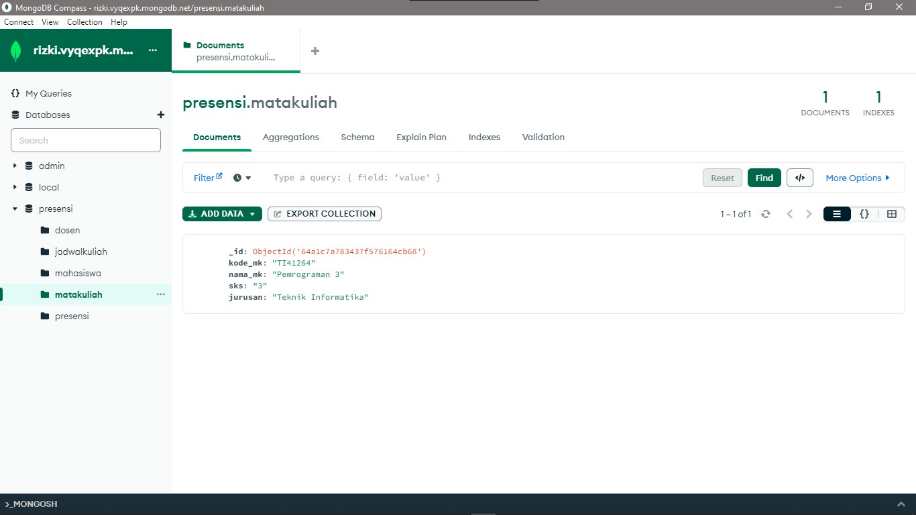
Gambar 4.12. MongoDB Jadwal Kuliah

* Mahasiswa



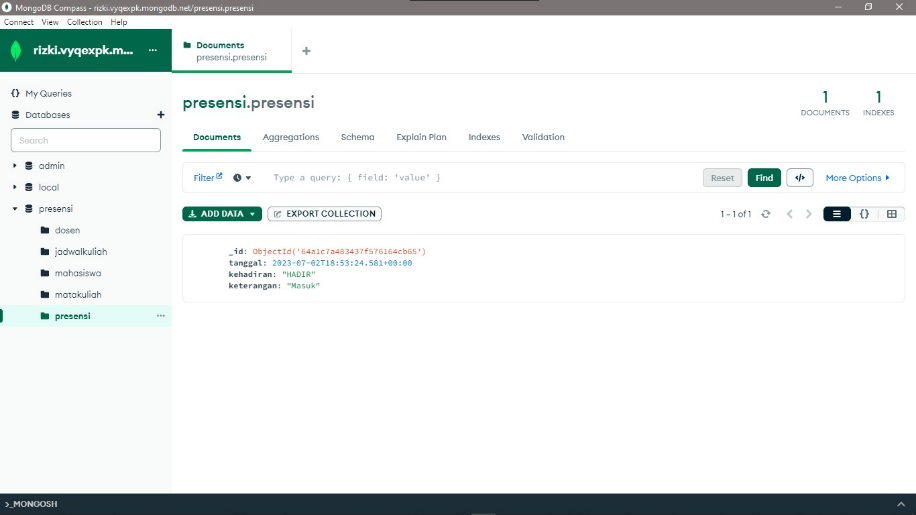
Gambar 4.13. MongoDB Mahasiswa

* Mata Kuliah



Gambar 4.14. MongoDB Mata Kuliah

* Presensi



Gambar 4.15. MongoDB Presensi

Pada 5 gambar di atas merupakan nama table dari database presensi, pada sidebar kiri terdapat nama database dan table dari isinya.

* + - 1. Merancang Controller

Sekarang kita akan memasuki tahap merancang sebuah controller. Pada tahap ini kita akan membuat mekanisme untuk mengambil suatu data melalui API dari database yang dibuat, berikut gambar code nya:

A screenshot of a computer program

Description automatically generated

Gambar 4.16. Controller Back-End

Source code di atas merupakan cara untuk memanggil function dan struct dari repositori github “username/kampus/model”.

* + - 1. Merancang URL

Setelah merancang sebuah controller, langkah selanjutnya adalah merancang sebuah URL yang akan memberikan akses pada Controller tersebut. Hal ini dilakukan untuk merancang URL yang mengarah ke Controller dalam file ini, sehingga URL tersebut dapat di akses. Berikut contoh code nya dibawah ini:

A screenshot of a computer program

Description automatically generated

Gambar 4.17. URL Back-End

* + - 1. Konfigurasi CORS

Setelah Merancang Controller dan URL, langkah selanjutnya adalah melakukan konfigurasi CORS atau Cross Origin Resource Sharing, CORS adalah kebijakan keamanan yang diterapkan oleh web browser untuk melindungi lintasan data.

A screen shot of a computer

Description automatically generated

Gambar 4.18. Konfigurasi CORS

Disini kita akan mengkonfigurasikan pengaturan CORS dalam folder “config” dengan file “cors.go”. di dalam file ini kita akan menentukan Header, Key, dan Web Aplikasi atau situs mana yang diizinkan untuk mengakses jalur data.

Pada kode CORS di atas, terdapat variabel AllowOrigins yang berfungsi untuk mengizinkan asal (origins) yang dapat mengakses jalur lintasan data. Untuk variabel Cors, teman-teman dapat mengikuti contoh kode di atas. Pertama, buat aplikasi front-end dan pastikan untuk mendeploynya menggunakan GitHub Pages.

Setelah membuat halaman GitHub Pages, cantumkan link GitHub Pages teman-teman, contohnya "https://farhanrizkim.github.io", dalam variabel AllowOrigins. Hal ini memberikan izin akses untuk lintasan data. Jika sudah siap, teman-teman dapat mendeploy aplikasi back-end di Heroku.

Jadi, langkah-langkahnya adalah: buat front-end, deploy front-end menggunakan GitHub Pages, cantumkan link GitHub Pages di variabel AllowOrigins untuk izin akses, dan terakhir, deploy aplikasi back-end di Heroku.

* + - 1. Push Github dan Deploy Heroku

Sebelum melakukan deploy, langkah pertama yang dilakukan adalah melakukan push ke repositori github terlebih dahulu. Setelah melakukan push github, kita dapat melakukan deploy ke Heroku dengan mengikuti langkah-langkah berikut ini:

Push Github

Disini saya akan melakukan push github menggunakan command dengan berikut:

* git status
* git add .
* git commit -m “Berikan Pesan Commit”
* git push

Membuat App di Heroku

Langkah pertama untuk membuat app di Heroku adalah pastinya anda sudah memiliki akun Heroku, setelah itu lanjutkan langkah-langkah berikut di bawah ini:

* Masuk ke akun Heroku

A screenshot of a computer

Description automatically generated

Gambar 4.19. Dashboard Heroku

* Pada dashboard Heroku, kalian akan melihat tombol “New” atau “Create New App”. Klik tombol tersebut untuk membuat aplikasi baru.

A screenshot of a computer

Description automatically generated

Gambar 4.20. Create New App

* Di halaman “Create New App”, kalian akan diminta untuk memberikan nama aplikasi. Disini saya akan memasukan nama aplikasi “presensi”

A screenshot of a computer

Description automatically generated

Gambar 4.21. Presensi Aplikasi

* Setelah membuat “Create New App” langsung melakukan add remote heroku sesuai nama aplikasi yang sudah kita buat.

Berikut command nya:

[ heroku git:remote -a “nama aplikasi” ]

* Setelah melakukan add remote heroku, selanjutnya kalian untuk mendeploy ke heroku dengan command:

[ git push heroku main ]

* Add on dan Setting ENV heroku Mariadb Jawsdb

Untuk menambahkan addons nya kalian diminta untuk mengunjungi link di bawah ini:

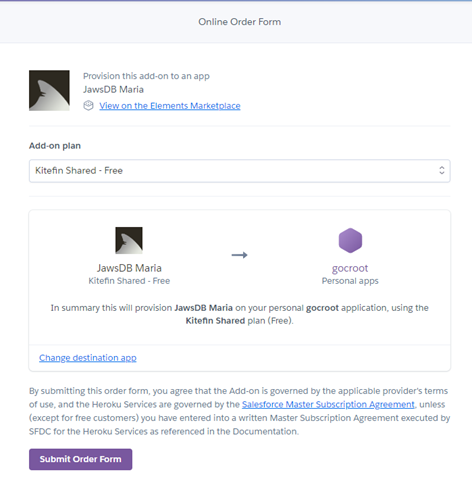
[ <https://elements.heroku.com/addons/jawsdb-maria> ]

A screenshot of a computer

Description automatically generated

Gambar 4.22. Jawsdb-maria

Selanjutnya kalian bisa langsung install seperti pada gambar di atas, lalu kalian bisa memilih yang gratis seperti gambar di bawah ini:



Gambar 4.23. Jawsdb-maria presensi

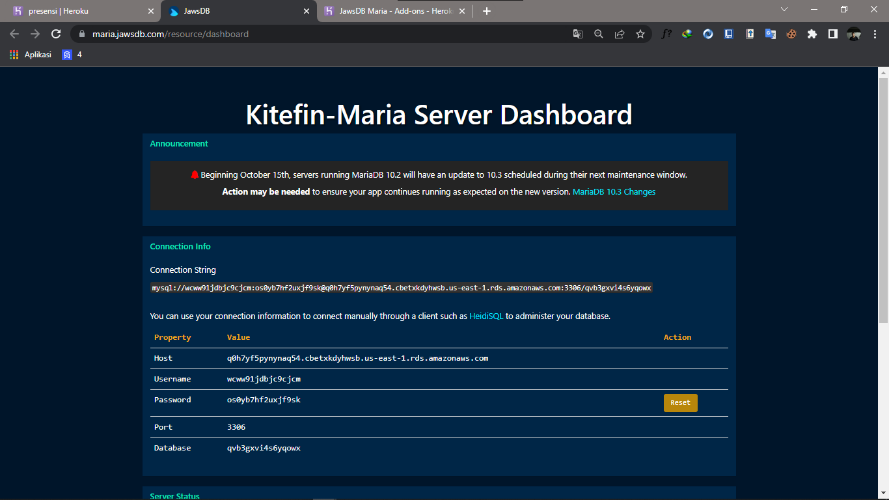
Untuk menggunakan JawsDB Maria (layanan basis data di Heroku), kita hanya perlu mengambil Connection String yang dapat dengan klik JawsDB Maria di bagian Add-ons.

A screenshot of a computer

Description automatically generated

Gambar 4.24. Jawsdb-maria add-ons

Setelah anda menklik akan muncul dan direct ke halaman seperti gambar di bawah ini:





Gambar 4.25. Kitefin-Maria Server Dashboard

Setelah mendapatkan key-key yang diperlukan seperti host, username, password, dan sebagainya, langkah selanjutnya adalah mengedit atau mengatur environment (env) di aplikasi Heroku kita.

A screenshot of a computer

Description automatically generated

Gambar 4.26. Settings Heroku

Di menu "Settings" di dashboard Heroku, kalian akan menemukan nama aplikasi nya. Di bawah nama aplikasi, kalian akan melihat bagian yang disebut "Config Vars". Di bagian ini, kalian dapat mengubah variabel-variabel konfigurasi yang digunakan oleh aplikasi.

Selanjutnya, kalian semua akan menambahkan variabel “MARIASTRINGAKADEMIK” dan “MONGOSTRING” ke konfigurasi. Berikut adalah contohnya:

* [“MARIASTRINGAKADEMIK=username:password”@tcp(q0h7yf5pynynaq54.cbetxkdyhwsb.us-east-1.rds.amazonaws.com:3306)/](mailto:)
* “MONGOSTRING=mongodb+srv://username:password@rizki.vyqexpk.mongodb.net/”

A screenshot of a computer

Description automatically generated

Gambar 4.27. Config Vars

Konfigurasikan Network Access MongoDB Atlas Website

Untuk memungkinkan aplikasi di Heroku mengakses MongoDB yang telah dibuat di MongoDB.com, Anda perlu memberikan alamat IP host Heroku yang diperoleh saat membuat controller homepage ke MongoDB.com.

Setelah itu, silakan masukan data dummy ke koleksi (collection) menggunakan MongoDB Compass yang diakses melalui paket yang telah dikembangkan sesuai keinginan.

A screenshot of a computer

Description automatically generated

Gambar 4.28. IP Access MongoDB

Jika masih belum, kamu klik Allow Acces From Anywhere dan isikan 0.0.0.0/0 lalu confirm.

Melihat Hasil Heroku “Open App”.

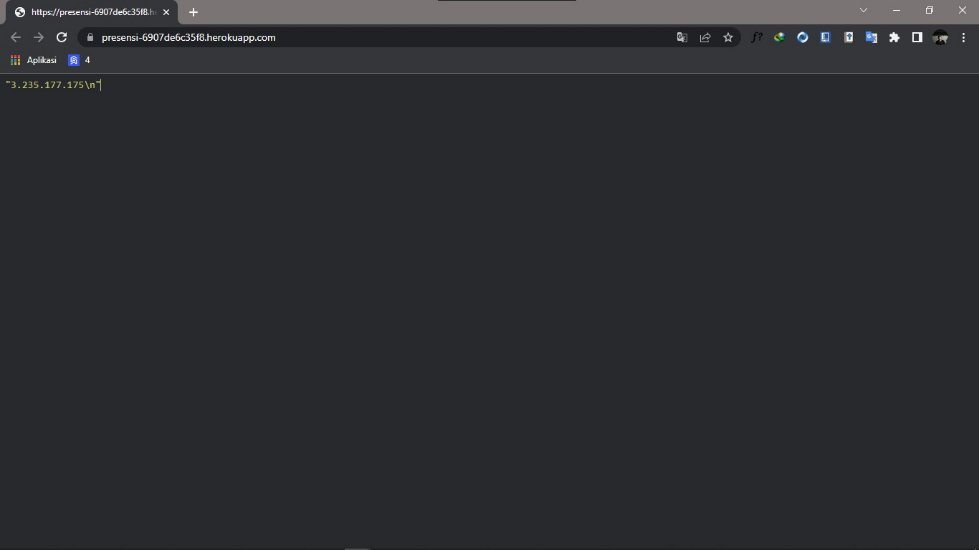
Setelah semua sudah dilakukan kalian bisa untuk melihat hasil yang telah kalian buat, cara melihatnya kalian bisa buka di heroku di dalam dashboard heroku dan klik “open app”.

A screenshot of a computer

Description automatically generated

Gambar 4.29. Open App Heroku

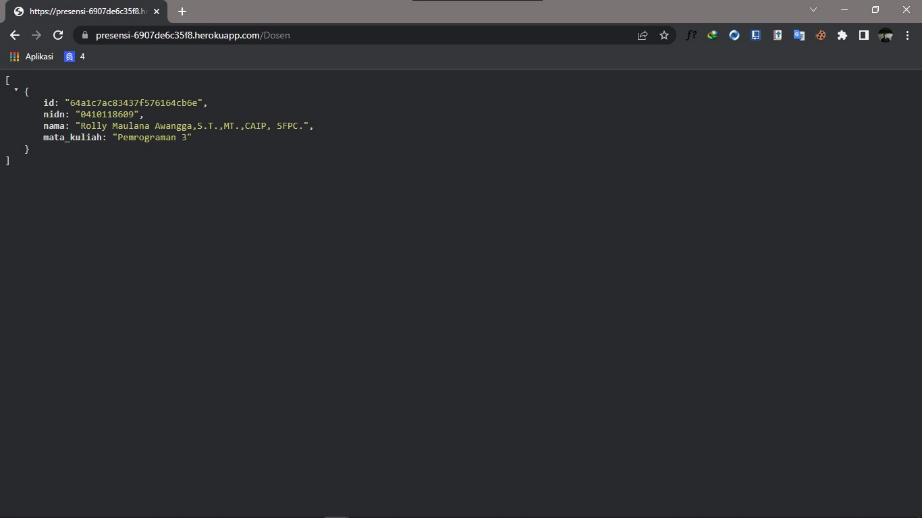
Kalian bisa langsung klik “Open App” untuk melihat hasilnya.



Gambar 4.30. link deploy aplikasi heroku

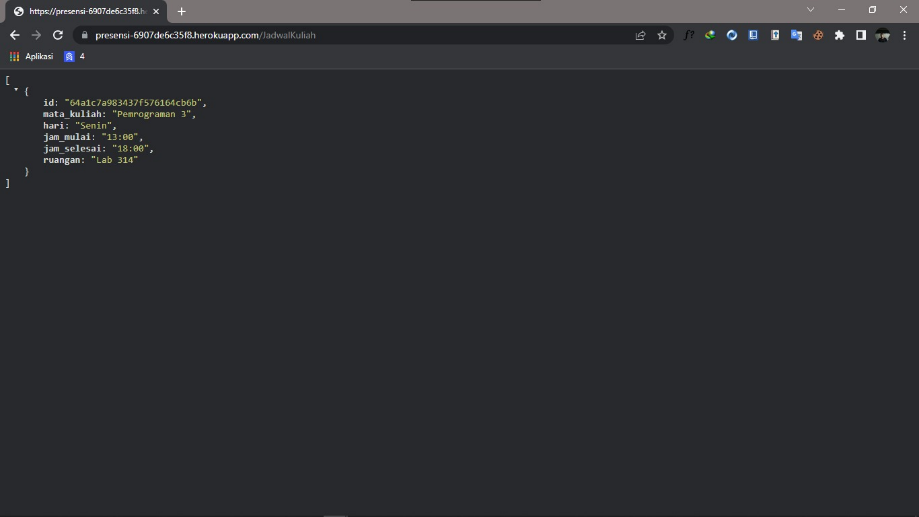
Hasil Deploy Heroku Studi Kasus Presensi

* Dosen



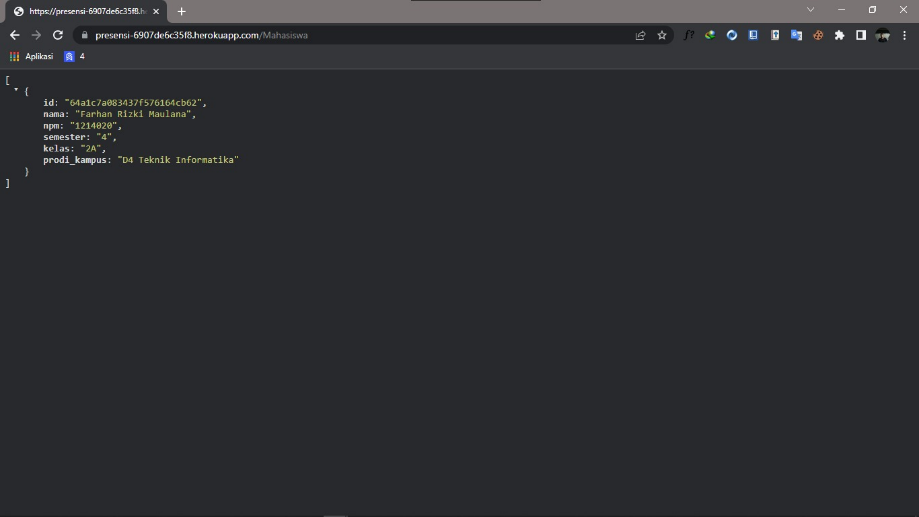
Gambar 4.31. hasil deploy heroku dosen

* Jadwal Kuliah



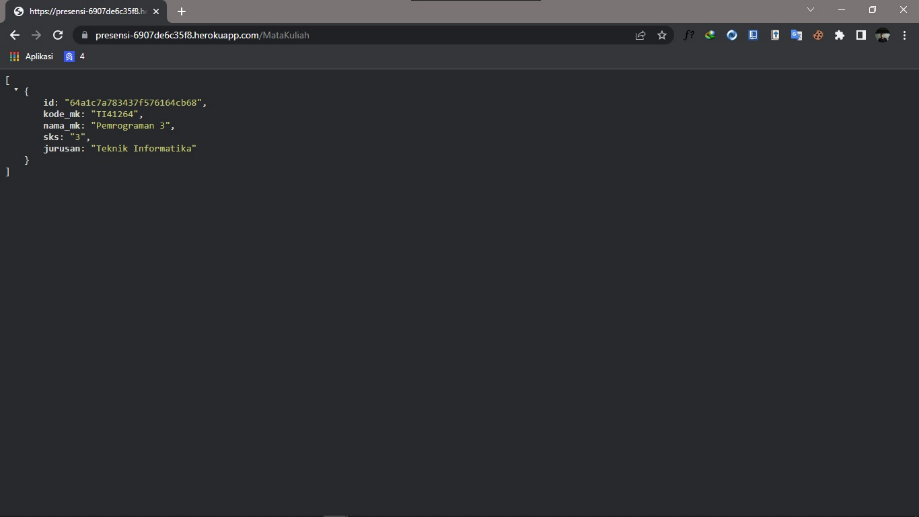
Gambar 4.32. hasil deploy heroku jadwal kuliah

* Mahasiswa



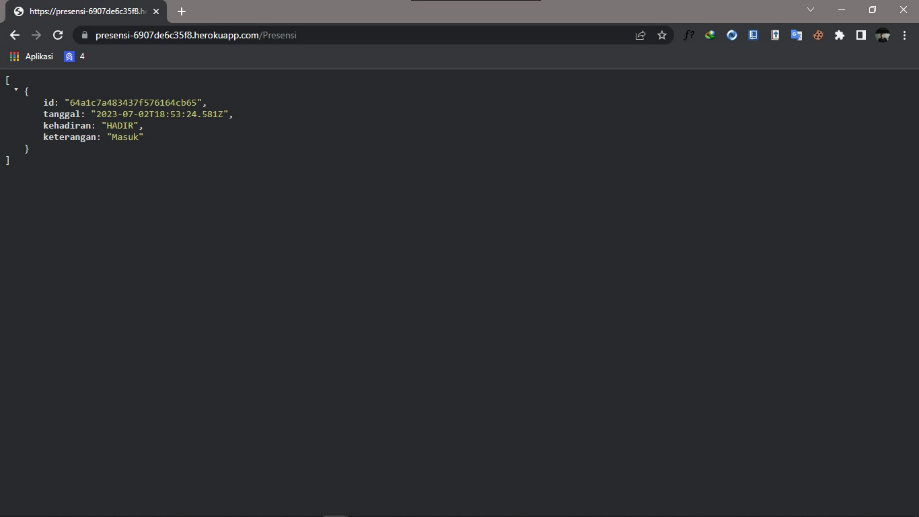
Gambar 4.33. hasil deploy heroku mahasiswa

* Mata Kuliah



Gambar 4.34. hasil deploy heroku mata kuliah

* Presensi



Gambar 4.35. hasil deploy heroku presensi

1. **Membuat Struktur Folder dan File untuk Package Back-End**

Selanjutnya kita memasuki tahap Frontend. Kita diharuskan mencari atau membuat tampilan Frontend yang menarik atau sederhana juga tidak masalah. Untuk memudahkan, kita dapat mencari template Frontend yang sesuai dengan studi kasus kita melalui mesin pencari Google.

Anda dapat mencari template Front-End gratis menggunakan Tailwind CSS di Google dengan kata kunci "Template Free Front-End Tailwind CSS Gratis". Setelah menemukan template yang sesuai, Anda bisa melakukan modifikasi sesuai kebutuhan. Selain itu, Anda juga dapat mengikuti kode Front-End pada repositori GitHub atau buku yang bisa Anda temukan melalui tautan ini:

[https://github.com/FarhanRizkiM/app/tree/main/akademik/presensi].

* + - 1. Struktur Front-End Javascript

Pada tahap ini, kita akan membuat dan memahami struktur kode untuk Front-End JavaScript. Tahap ini memiliki tingkat penting yang cukup tinggi, karena akan membentuk dasar dari aplikasi kita. Mari kita lanjutkan dengan melihat struktur file yang akan digunakan.

A screenshot of a computer program

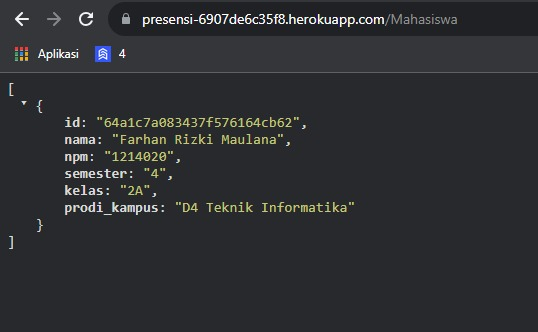
Description automatically generated

Gambar 4.35. struktur front-end

Pada gambar di atas, terdapat struktur file untuk Front-End JavaScript yang mencakup berkas HTML dan juga direktori JS yang akan digunakan untuk mengambil data dari API dan melakukan permintaan POST ke API.

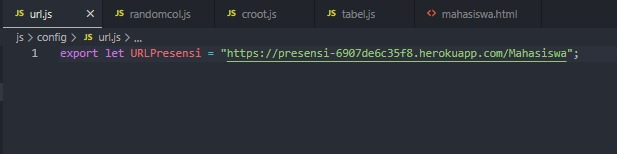
* Folder "Config" berisi URL-URL yang telah kita buat, seperti API untuk mengambil data dari database yang telah dibuat pada aplikasi Back-End. Sebagai contoh, misalnya urlapp.com/nama dan telah dibuat berbagai metode pada URL /surat/insert.
* Selanjutnya, terdapat "Controller" yang akan digunakan sebagai modul di HTML untuk mengirimkan data melalui aplikasi Front-End ke database melalui API.
* Folder "Template" berisi data API yang akan disimpan untuk ditampilkan dalam tabel HTML kita. Data ini akan diimpor dan digunakan dalam berkas "croot.js".
* Berkas "croot.js" akan dipanggil dalam berkas HTML kita dan berisi berbagai metode untuk mengambil data dari API yang telah kita buat pada aplikasi Back-End.
  + - 1. Merancang Get Data API dan HTML

Pertama, lakukan pengaturan pada berkas "Url.js" untuk menentukan URL tempat kita akan mengambil data melalui metode GET. Sebagai contoh, kita dapat menaruh link untuk mengambil data dengan cara sebagai berikut:



Gambar 4.36. get data

Selanjutnya, buka berkas "Url.js" dan masukkan URL <https://presensi-6907de6c35f8.herokuapp.com/Mahasiswa> sebagai tempat kita akan mengambil data. Pastikan URL tersebut terletak di berkas "Url.js" seperti yang terlihat pada gambar di bawah ini:



Gambar 4.37. tampilan url.js

Selanjutnya, mari kita persiapkan sebuah template untuk menyimpan atau menampilkan data dari API kita dalam bentuk tabel. Berikut adalah struktur template yang dapat digunakan:

**A screenshot of a computer program

Description automatically generated**

Gambar 4.38. tampilan url.js

Pada contoh kode di atas, terdapat lima placeholder yang ditandai dengan #nama#, #prodi\_kampus#, #npm#, #semester#, dan #kelas#. Fungsi dari placeholder ini adalah untuk menyimpan data nama, program studi kampus, nomor pokok mahasiswa (NPM), semester, dan kelas dari mahasiswa yang akan diambil dari sumber data, mungkin dari database atau API.

File ini nantinya akan diimpor (import) ke dalam file croot.js. Di dalam croot.js, nilai-nilai placeholder akan diisi dengan data yang relevan sebelum file tersebut dieksekusi. Dengan cara ini, data dari mahasiswa yang tersimpan dalam database atau API akan diolah dan ditampilkan dalam bentuk tabel atau informasi yang lebih terstruktur.

Dengan menggunakan struktur template dan placeholder ini, data dari sumber eksternal (misalnya database mahasiswa) dapat diintegrasikan ke dalam aplikasi yang lebih besar (seperti halaman web) melalui croot.js. Hal ini memungkinkan pengelolaan dan tampilan data yang lebih terorganisir dan dapat diakses dengan lebih mudah.

A screen shot of a computer program

Description automatically generated

Gambar 4.39. tampilan croot.js

Dengan menggunakan struktur template dan placeholder ini, data dari sumber eksternal (misalnya database mahasiswa) dapat diintegrasikan ke dalam aplikasi yang lebih besar (seperti halaman web) melalui croot.js. Hal ini memungkinkan pengelolaan dan tampilan data yang lebih terorganisir dan dapat diakses dengan lebih mudah.

Selanjutnya, panggil file JavaScript (croot.js) dalam HTML dengan menempatkannya di dalam bagian head atau setelah tag penutup </body>. Gunakan perintah <script type="module" src="js/controller/croot.js"></script> untuk menghubungkan dan memuat file JavaScript.

Dengan langkah ini, script JavaScript (croot.js) akan terpanggil dan siap digunakan dalam halaman HTML. Lalu, tambahkan atribut id="demo" pada elemen <tbody> dari tabel Anda. Contoh tampilan struktur tabel seperti pada gambar di bawah ini:

A screenshot of a computer program

Description automatically generated

Gambar 4.40. tampilan HTML

Dengan menambahkan id="demo" pada elemen <tbody>, Anda memberikan tempat bagi croot.js untuk mengisi data ke dalam elemen tersebut. Selanjutnya, JavaScript dari croot.js akan dieksekusi dan mengisi data dari sumber eksternal (API atau database) ke dalam tabel yang telah disiapkan.

Pastikan bahwa perintah <script type="module" src="js/controller/croot.js"></script> mengarahkan ke file croot.js yang berada dalam direktori "js/controller" (js/controller/croot.js). Pastikan juga alamat file tersebut sesuai dengan struktur direktori pada proyek Anda. Dengan cara ini, JavaScript akan dijalankan dan tabel akan diisi dengan data yang diambil dari sumber eksternal.

Jika Anda mengalami kesalahan atau masalah (Error) saat menggunakan kode tersebut, jangan khawatir. Anda dapat mencoba memecahkan masalah tersebut dengan mencari solusi melalui sumber-sumber seperti StackOverflow atau mengulas kembali buku atau materi yang sedang digunakan.

Jika Anda menghadapi masalah dalam kode JavaScript, Anda bisa menggunakan fitur Console dalam web browser untuk memeriksa dan menganalisis kesalahan yang terjadi. Console dapat memberikan informasi penting seperti pesan kesalahan, traceback, atau nilai variabel saat program dieksekusi. Dengan begitu, Anda dapat mengidentifikasi dan menemukan penyebab masalah tersebut.

Jangan ragu untuk mencari dukungan dari komunitas pengembang dan sumber-sumber online untuk menyelesaikan kesalahan atau masalah yang mungkin Anda temui. Dengan berbagai referensi dan alat debugging yang tersedia, Anda dapat memperbaiki kode dan mencapai hasil yang diharapkan.

* + - 1. Merancang Deploy Front-End Menggunakan Github Pages.

Langkah terakhir dalam pembuatan aplikasi presensi adalah melakukan deployment front-end. Untuk melakukannya, buka repositori GitHub Anda dan pastikan semua perubahan terakhir sudah di-update atau diperbarui dalam repositori tersebut. Pastikan juga Anda telah mem-push perubahan terakhir ke repositori agar semua pembaruan terakhir tersedia.

Setelah memastikan repositori telah diperbarui, lanjutkan dengan melakukan deployment pada front-end aplikasi presensi. Anda dapat mengakses halaman repositori di GitHub dan melakukan konfigurasi yang diperlukan, seperti memilih branch yang ingin dideploy, misalnya "gh-pages" atau branch yang sesuai dengan tujuan deployment.

Setelah mengatur dan menyimpan konfigurasi, proses deployment akan dimulai. Setelah selesai, aplikasi presensi akan tersedia secara publik dan dapat diakses melalui URL yang diberikan oleh GitHub Pages.

Pastikan untuk mengikuti langkah-langkah secara teliti agar proses deployment berjalan dengan lancar dan aplikasi presensi dapat diakses oleh pengguna secara online.

Mari segera kita akses ke repositori teman-teman. Gambar di bawah ini akan menunjukkan bagaimana tampilan repositori tersebut :

A screenshot of a computer

Description automatically generated

Gambar 4.41. tampilan Github Pages app

Untuk menuju ke repositori teman-teman, cukup akses repositori yang dimaksud. Setelah itu, pada bagian navbar terdapat opsi "Setting". Kemudian, pilih "Pages" pada sidebar kiri, dan tampilannya akan mirip dengan gambar di atas.

Selanjutnya, di bagian "Build and Deployment" untuk "Source", pilih opsi "Deploy from a branch". Anda perlu mengubah branch menjadi "main" atau "master" sesuai dengan branch utama repositori Anda. Setelah mengubahnya, klik tombol "Save".

Setelah menyimpan perubahan, Anda akan perlu menunggu beberapa saat untuk proses "Deployment Frontend" selesai. Ini mungkin memakan waktu 1-5 menit hingga halaman "Visit Site" muncul dan front-end aplikasi dapat diakses melalui tautan tersebut.

Jika proses deployment telah selesai, Anda akan melihat tampilan seperti ini:

A screenshot of a computer

Description automatically generated

Gambar 4.42. tampilan Front-end gocroot

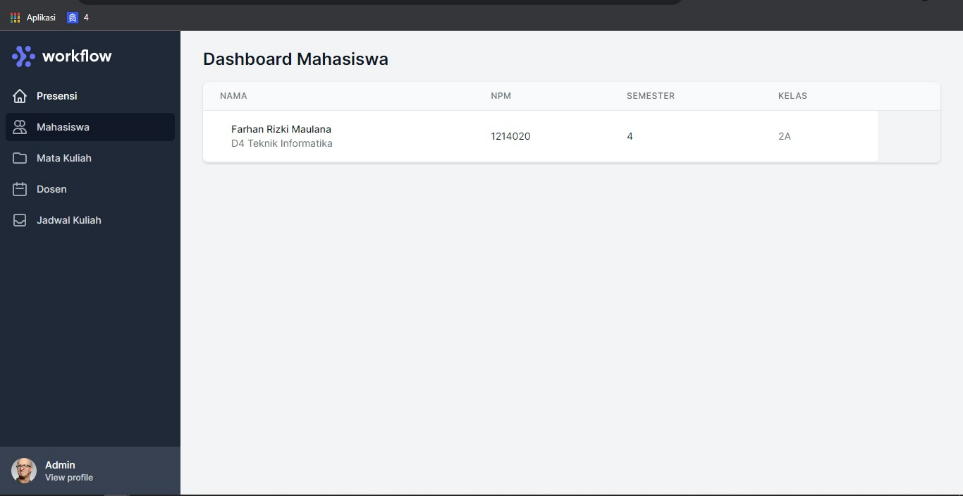
Selanjutnya, Anda hanya perlu mengarahkan ke file .html yang ingin Anda tampilkan. Misalnya, jika Anda ingin menampilkan halaman utama aplikasi, Anda dapat mengarahkan pengguna ke file "mahasiswa.html" dengan menggunakan URL seperti berikut:

**A screenshot of a computer

Description automatically generated**

Gambar 4.43. tampilan repo app

Untuk mengarahkan ke halaman "mahasiswa.html" dan menampilkan aplikasi akademik dan presensi, Anda hanya perlu mengganti link di kode HTML agar mengarah ke file "mahasiswa.html". Misalnya, jika Anda ingin mengarahkan pengguna ke halaman tersebut, Anda dapat melihat hasilnya sebagai berikut:

****

Gambar 4.43. tampilan hasil

Dengan menggunakan link di atas, pengguna akan diarahkan ke halaman "mahasiswa.html" yang berisi aplikasi akademik dan presensi yang Anda ingin tampilkan. Pastikan file "mahasiswa.html" sudah ada dan berisi kode yang sesuai untuk aplikasi tersebut.

**DAFTAR PUSTAKA**

Golang: Definisi, Fungsi, Framework dan Kelebihannya. (2021, February 16). Software House & System Integrator Di Malang, Indonesia. <https://www.sekawanmedia.co.id/blog/belajar-golang/>

Angela. (2022, August 19). Golang: Pengertian, Fungsi, dan Keunggulannya. Binaracademy.com. <https://www.binaracademy.com/blog/apa-itu-golang-dan-fungsinya>

Apa itu JavaScript? - Penjelasan tentang JavaScript (JS) - AWS. (2023). Amazon Web Services, Inc. <https://aws.amazon.com/id/what-is/javascript/>

MongoDB: Definisi, Kelebihan, Cara Kerja, dan Instalasinya. (2021, February 26). Software House & System Integrator Di Malang, Indonesia. <https://www.sekawanmedia.co.id/blog/belajar-mongodb/>

Dicoding Intern. (2021, January 12). Apa itu Web Service? Beserta Pengertian dan Contohnya - Dicoding Blog. Dicoding Blog. <https://www.dicoding.com/blog/apa-itu-web-service/>

All releases - The Go Programming Language. (2016). Go.dev. <https://go.dev/dl/>

Kelebihan dan Kekurangan Bahasa Pemrograman Golang. (2019, November 10). Artikel & Berita Terbaru Tentang Web Developer. <https://webdev-id.com/berita/kelebihan-dan-kekurangan-dari-penggunaan-bahasa-pemrograman-golang/#:~:text=Keunggulan%20utama%20bahasa%20Go%20terkadang,memiliki%20fleksibilitas%20yang%20kurang%20baik>.

Bagaimana mengaktifkan JavaScript di peramban web dan mengapa. (2023). Enable-Javascript.com. <https://enable-javascript.com/id/>

cnblogadmin. (2022, August 4). Belajar Golang : Apa Itu Golang , Pengertian Dan Kelebihannya. Course-Net Blog. <https://course-net.com/blog/belajar-golang/>

Content IDCloudHost. (2022, June 30). Apa Itu JavaScript? Fungsi, Kelebihan dan Kekurangannya. IDCloudHost. <https://idcloudhost.com/blog/apa-itu-javascript/>

PRIADI, S. (2022). Implementasi Rest Dalam Membangun Web Service Menggunakan Golang (Studi Kasus: Aplikasi Satudikti).

Sulistiani, W. (2020). Implementasi Web Service dengan Metode REST Berbasis Golang pada Layanan Google Cloud Platform di PT Sumber Alfaria Trijaya, Tbk (Doctoral dissertation).

Kroons, A. A., & Dewi, C. (2023). PENGEMBANGAN DASHBOARD TRIVY BERBASIS WEBSITE MENGGUNAKAN REACT JS DAN GOLANG. Jurnal Indonesia: Manajemen Informatika dan Komunikasi, 4(3), 1037-1049.

Trijtoho, D. S. (2022). PERANCANGAN APLIKASI KEHADIRAN PEGAWAI DENGAN FRAMEWORK ECHO GOLANG MENGGUNAKAN FACE-RECOGNITION DAN GEOLOCATION (STUDI KASUS PT KB FINANSIA MULTI FINANCE) (Doctoral dissertation, Program Studi Teknik Informatika Fakultas Teknik Universitas Widyatama).

Hermanto, N., & Riyanto, N. R. D. R. (2019). Aplikasi sistem presensi mahasiswa berbasis android. Simetris: Jurnal Teknik Mesin, Elektro dan Ilmu Komputer, 10(1), 107-116.

**GLOSARIUM**

A

***Aplikasi****:* Sebuah perangkat lunak yang dirancang untuk memberikan fungsi dan layanan tertentu kepada pengguna melalui perangkat elektronik, seperti komputer, smartphone, atau tablet.

***Algoritma****:* Sekumpulan instruksi atau langkah-langkah terdefinisi dengan jelas yang dirancang untuk menyelesaikan suatu masalah atau mencapai tujuan tertentu.

***Animasi****:* Proses menciptakan ilusi gerakan atau perubahan dalam gambar atau objek.

***Aksebilitas****:* Konsep yang mengacu pada kemudahan akses dan penggunaan yang setara bagi individu dengan berbagai kemampuan dan kondisi.

***API****:* Antarmuka perangkat lunak yang memungkinkan berbagai aplikasi untuk saling berinteraksi dan berkomunikasi.

B

***Browser***: Perangkat lunak atau aplikasi yang digunakan untuk mengakses dan menavigasi halaman web dan konten di internet.

**Bahasa:** Sistem komunikasi yang digunakan oleh manusia untuk berinteraksi, menyampaikan gagasan, dan memahami informasi.

***Back-end****:* Bagian dari suatu aplikasi atau sistem yang berfungsi di sisi server dan bertanggung jawab untuk pemrosesan dan manajemen data, logika bisnis, dan interaksi dengan pangkalan data.

C

***Carousel:*** Fitur atau elemen antarmuka pengguna yang memungkinkan pengguna untuk melihat konten secara bergantian atau meluncurkan gambar, teks, atau elemen media lainnya dalam urutan yang terdefinisi.

***CSS:*** Bahasa pemrograman yang digunakan untuk mengendalikan tampilan dan gaya elemen HTML atau XML dalam halaman web

***Coding:*** Proses menulis dan membuat program komputer menggunakan bahasa pemrograman.

***Code:*** Serangkaian instruksi atau pernyataan yang ditulis dalam bahasa pemrograman untuk menginstruksikan komputer melakukan tugas tertentu.

***Config:*** Singkatan dari "Configuration" yang merujuk pada pengaturan atau konfigurasi dalam suatu sistem atau program.

***CORS:*** sebuah mekanisme yang digunakan oleh browser web untuk memungkinkan permintaan sumber daya lintas asal (cross-origin resource request) dari satu domain ke domain lain.

***Commit:*** tindakan menyimpan perubahan pada repositori yang dilakukan dalam sistem kontrol versi.

***Command***: instruksi yang diberikan kepada sistem operasi atau program komputer untuk melakukan tugas tertentu.

***Create:*** tindakan atau proses membuat sesuatu yang baru, seperti menciptakan objek, dokumen, file, atau entitas lainnya dalam konteks pengembangan perangkat lunak atau desain.

***Connection:*** mengacu pada koneksi atau interaksi antara dua atau lebih entitas, baik itu antar orang, perangkat, sistem, atau entitas lainnya.

D

***Digital:*** merujuk pada representasi data atau informasi dalam bentuk angka atau bilangan diskret.

***Data:*** kumpulan fakta, angka, informasi, atau instruksi yang dikumpulkan, diatur, dan diolah untuk penggunaan atau analisis lebih lanjut.

***Dropdown:*** elemen antarmuka pengguna yang berisi daftar pilihan yang muncul ketika pengguna mengklik atau mengarahkan kursor ke suatu area.

***Database:*** kumpulan terstruktur dari data yang diatur dan disimpan dalam suatu sistem komputer.

***Deploy:*** tindakan atau proses menginstal, mengkonfigurasi, dan meluncurkan aplikasi atau sistem ke lingkungan produksi atau pengguna akhir.

***Dependensi:*** ketergantungan antara dua atau lebih komponen perangkat lunak di mana perubahan pada satu komponen dapat mempengaruhi komponen lainnya.

***Dummy:*** kata yang digunakan untuk merujuk pada objek atau entitas yang tidak memiliki fungsi atau nilai sebenarnya,

E

***Element:*** komponen dasar atau bagian utama dari sesuatu.

***Enviroment:***Seluruh kondisi fisik, kimia, biologis, sosial, dan budaya di sekitar individu, organisme, atau kelompok.

F

***Fungsi:*** peran atau tujuan sesuatu dalam konteks tertentu.

***Framework:*** Kerangka kerja perangkat lunak yang menyediakan struktur, komponen, dan alat bantu untuk membangun aplikasi back-end dengan lebih efisien.

***Front-End:*** pengembangan perangkat lunak yang berkaitan dengan tampilan dan interaksi yang langsung dilihat dan digunakan oleh pengguna,

***Fungsionalitas:*** Kemampuan suatu sistem, aplikasi, atau produk untuk menjalankan tugas atau fungsi tertentu yang diharapkan.

***Format:*** Pengaturan atau tata cara penyajian atau struktur data yang telah ditentukan.

***Fleksibel:*** Kemampuan untuk beradaptasi, berubah, atau menyesuaikan diri dengan situasi atau kebutuhan yang berbeda.

***Full Stack:*** kemampuan atau peran seseorang dalam mengembangkan aplikasi atau sistem yang mencakup pengembangan baik di sisi server (back-end) maupun di sisi klien (front-end).

***Function:*** Blok kode yang dapat menerima input, melakukan operasi atau tugas tertentu, dan mengembalikan output.

***Fork:*** operasi yang menghasilkan salinan identik dari suatu proses yang sudah ada.

G

***Golang*:** Bahasa pemrograman open-source yang dibuat Google.

***Google*:** Perusahaan teknologi yang menyediakan layanan dan produk.

***Grafis:*** Representasi visual dari data atau informasi menggunakan elemen-elemen seperti gambar, grafik, dan ilustrasi.

***Get:*** memperoleh atau menerima sesuatu.

***Github:*** *Platform web dan layanan berbasis cloud untuk mengelola dan menyimpan kode developer.*

***Gratis:*** *Tidak dikenakan biaya atau tidak dipungut bayaran.*

H

***HTML:*** *bahasa markup untuk halaman web.*

***Heroku:*** Platform cloud yang berbasis kontainer dan digunakan sebagai Platform as a Service (PaaS) untuk mengelola, mengelola, dan memperbesar aplikasi modern.

***Header:*** Bagian atas dari halaman web yang biasanya berisi logo, menu navigasi, dan informasi penting lainnya.

***Homepage:*** Halaman utama atau halaman pembuka dari sebuah situs web.

I

***Interaktif:*** Dapat berinteraksi atau berhubungan dengan pengguna atau pemirsa.

***Internet:*** Jaringan global yang menghubungkan jutaan komputer di seluruh dunia.

***Indeks:*** Daftar atau kumpulan data yang disusun secara teratur dan terorganisir untuk memudahkan pencarian atau pengambilan informasi.

***Instalasi:*** Proses pemasangan dan pengaturan perangkat lunak, perangkat keras, atau sistem pada komputer atau perangkat lainnya.

J

***Javascript:*** Bahasa pemrograman untuk halaman web menjadi interaktif dan responsif.

***Java:*** Bahasa pemrograman back-end.

***JSON:*** Model format data yang ringan, sederhana untuk dibaca dan ditulis oleh manusia, dan mudah diproses oleh komputer.

K

***Kompleksitas:*** Tingkat kerumitan atau keruwetan suatu sistem atau masalah.

***Kapasitas:*** Kemampuan suatu sistem atau objek untuk menampung atau memproses sesuatu.

***Kinerja:*** Hasil Kerja yang dicapai.

***Konfigurasi:*** Proses atau hasil dari mengatur atau mengonfigurasi suatu sistem,

 L

***Library:*** Kumpulan kode program untuk memudahkan pengembang aplikasi.

***Link:*** link adalah elemen dalam dokumen HTML yang digunakan untuk menghubungkan halaman web

M

***MongoDB:*** Database NoSQL yang digunakan untuk mengelola data secara fleksibel.

***Menu:*** halaman pada sebuah aplikasi atau website

***Minimal:*** Kuantitas atau ukuran yang terkecil atau paling sedikit

***Modern:*** Terkini, Baru.

N

***Network:*** Jaringan komunikasi.

O

***Object:*** instance dari sebuah kelas yang memiliki atribut dan method.

***Origins:*** Asal-usul atau sumber dari suatu hal atau konsep.

P

***Pemrograman:*** Sistem notasi untuk menulis program komputer

***Platform:*** media atau wadah yang digunakan untuk menjalankan perangkat lunak

***Python:*** bahasa pemrograman tingkat tinggi yang fokus pada keterbacaan kode dan produktivitas.

***Partisi:*** pembagian logis dari ruang penyimpanan pada media penyimpanan seperti hard disk.

***Package:***kumpulan modul atau file yang terorganisir dalam struktur direktori.

Q

***Query:*** permintaan untuk mengambil data dari basis data.

R

***Relatif:*** hubungan atau posisi suatu objek atau nilai terhadap objek atau nilai lainnya.

***Replikasi:*** proses membuat salinan data dari satu server ke server lain.

***Representasi:*** cara atau metode untuk menggambarkan atau menyajikan suatu objek atau konsep.

***Repositori:*** tempat penyimpanan untuk kode program atau dokumen

S

***Skalabilitas:*** kemampuan suatu sistem untuk menangani pertumbuhan data

***Sistem:*** Kumpulan komponen

***Struktur:*** susunan atau tata letak dari suatu objek atau sistem.

***Standalone:*** perangkat atau aplikasi yang dapat berjalan sendiri tanpa memerlukan koneksi ke jaringan

***SQL:*** bahasa pemrograman yang digunakan untuk mengakses dan memanipulasi data dalam basis data relasional.

***Statis:*** tidak berubah, tidak berinteraksi

***Struct:*** mengelompokkan beberapa variabel dengan tipe data yang berbeda menjadi satu unit.

***Server:***komputer atau perangkat keras yang digunakan untuk menyimpan dan mengelola data atau aplikasi

T

***Teknologi:*** kumpulan pengetahuan, keterampilan, dan alat yang digunakan untuk menciptakan, merancang, dan membangun produk atau layanan.

U

***Unduh:*** proses mengambil atau mengunduh file atau aplikasi dari internet ke komputer

***URL:*** Alamat yang unik dan terstruktur yang digunakan untuk mengidentifikasi dan mengakses sumber daya di internet.

V

***Visual:*** gambar, grafik, atau tampilan

***Visualisasi:*** proses membuat gambar atau grafik untuk mempresentasikan data atau informasi.

***Validasi:*** memastikan keakuratan dan keandalan data

W

***Web:*** mengakses dan berbagi informasi melalui halaman web dan aplikasi web

***Website:*** kumpulan halaman web.

***Windows:*** sistem operasi komputer yang dikembangkan oleh Microsoft.

X

Y

**KREDIT GAMBAR**

Halaman ini memuat daftar kepemilikan hak cipta/perizinan penggunaan gambar/foto berikut nomor halaman keberadaan gambar/foto tersebut.

Gambar 1.1. : Logo Bahasa Golang

Pencipta : Ryane Puspa

Sumber : <https://academy.alterra.id/blog/cara-terbaik-belajar-golang/>

Gambar 1.2. : Javascript

Pencipta : -

Sumber : <https://si.institutpendidikan.ac.id/product/belajar-javascript-pemula-sampai-mahir/>

Gambar 1.3. : MongoDB

Pencipta : Bauroziq

Sumber :<https://caraguna.com/pengertian-mongodb-kelebihan-dan-cara-kerjanya/>

Gambar 2.2. : Javascript tidak Perlu di Install

Pencipta : TheAlgorithms

Sumber : <https://github.com/TheAlgorithms/JavaScript>

**INDEKS**

A

agregasi 9

Agregasi 47

Aksebilitas 47, 60

Algoritma 5, 47

Animasi 7, 47

API 15, 19, 27, 31, 47, 60

Aplikasi 4, 7, 33, 36, 45, 46, 47

B

Back-end 47

Bahasa 4, 5, 8, 45, 47, 48, 50, 51

Browser 47

C

carousel 6

code 31, 32, 33

Command 48

commit 35

config 33

Connection 38, 48

controller 31, 33, 40

cors 33

Create 36, 37, 48

D

Data 7, 9, 49

Database 28, 29, 49, 52

Dependensi 49

Digital 49

Dropdown 49

Dummy 49

E

Element 49

Enviroment 49, 60

F

Fleksibel 9, 50

Fork 50

Format 50, 51

Framework 6, 45, 49

Full Stack 50

Function 21, 23, 24, 25, 26, 50

Fungsi iii, 4, 6, 8, 45, 49

Fungsionalitas 7, 9, 50

G

Get 19, 21, 50

Github 15, 16, 24, 35, 50

Golang i, iii, 4, 5, 6, 11, 12, 15, 16, 18, 45, 46, 50

Google 4, 5, 12, 46, 50

Grafis 50

Gratis 50

H

Header 33, 51

Heroku 15, 16, 34, 35, 36, 38, 39, 40, 41, 42, 51

Homepage 51

HTML 7, 8, 48, 51, 52

I

Indeks 51

Instalasi iii, 11, 12, 13, 14, 51

Interaktif 51

Internet 51

J

Java 8, 51

Javascript 0, ii, iii, 5, 6, 7, 12, 13, 45, 51

JSON 8, 9, 20, 51

K

Kapasitas 52

Kinerja 9, 52

Kompleksitas 52

Konfigurasi 28, 33, 34, 52

L

Library iii, 26, 52

Link 52

M

Menu 52

Minimal 52

Modern 52, 60

MongoDB i, iii, 8, 9, 10, 13, 14, 15, 17, 19, 20, 29, 30, 31, 40, 45, 52

N

Network 40, 52, 60

O

Object 8, 53, 60

Origins 53, 60

P

Package iii, 18, 53, 61

Partisi 53, 61

Pemrograman 45, 53, 60

PHP 6, 53, 60

Platform 46, 50, 51, 53, 60

Python ii, 6, 53, 60

Q

Query 53, 61

R

Relatif 53, 61

Replikasi 53, 61

Repositori 53, 61

Representasi 50, 53, 61

S

Server 39, 54, 61

Sistem iii, 11, 14, 47, 53, 54, 61

Skalabilitas 9, 54, 61

SQL 8, 9, 10, 54, 61

Standalone 7, 54, 61

Statis 54, 61

Struct 19, 20, 24, 54, 61

T

Teknologi 54, 61

U

Unduh 54, 61

URL 27, 33, 54, 61

V

Validasi 7, 55, 61

Visual 7, 19, 24, 55, 61

Visualisasi 55, 61

W

Web 0, i, 6, 33, 45, 46, 55, 61

Windows 11, 12, 14, 55, 61

**TENTANG PENULIS**

A person in a suit and tie

Description automatically generated

Perkenalkan nama saya Farhan Rizki Maulana, lahir di Jakarta 4 Mei 2003, tinggal dan besar di Bogor dari Sekolah Dasar sampai Sekolah Menengah Kejuruan. Sedang Menempuh Pendidikan Program Studi Diploma 4 Teknik Informatika di Universitas Logistik & Bisnis Internasional, Bandung. Aktivitas yang sering dijalani bermain futsal dan bulutangkis. Semoga adanya buku ini dapat menjadi manfaat bagi pembaca.

A person standing in front of a building

Description automatically generated

Fahad Abdul Aziz, lahir di Kota Garut pada tanggal 17 Juni 2003. Pendidikan tingkat dasar hingga menengah ditempuh di Garut. Mulai merantau sejak SMK, melatih kemandiriannya di SMKN 1 Garut dengan Jurusan Teknik Komputer jaringan dan mengikuti eskul Badminton. Melanjutkan pendidikan D4 Teknik Informatika di Universitas Logistik Dan Bisnis Internasional. Saya sangat menyukai teknologi dan ingin bekerja di bidang cyber security, backend dev, dan software engineering.

Nizar adalah seorang anak sulung yang tumbuh di Wanayasa. Minat utamanya adalah bermain game, membaca, petualangan, menonton film, dan ngopi. Dalam game, ia menemukan kegembiraan dan tantangan. Membaca membuka wawasannya, sementara petualangan mengembangkan pemahamannya tentang dunia. Menonton film memberinya inspirasi, sementara ngopi menjadi ritual ketenangan dan refleksi. Nizar berharap mengembangkan minatnya menjadi karier dan berbagi pengetahuan serta inspirasi dengan orang lain. Dia hidup dengan semangat dan siap menjalani perjalanan hidupnya dengan penuh gairah.

A child standing in front of a building

Description automatically generated

Ryaas Ishlah Ramadhan, lahir di Kota Bandung pada tanggal 13 November 2002. Pendidikan tingkat dasar hingga menengah ditempuh di Bandung. Mulai mengoding sejak kuliah, melatih pengodinganya di ULBI Bandung, Pengurus OSIS pada saat smp dan mengikuti eskul futsal. Melanjutkan pendidikan D4 di Universitas Logistik dan Bisnis Internasional dengan jurusan Teknik informatika. Saya Sangat menyukai teknogi canggih dan menyukai sepakbola dan ingin mempunyai usaha sendiri bersama keluarga dengan membagi tugas sesuai kemampuanya.

A person sitting on a rock

Description automatically generatedMochammad Iqbal Ramadhan, lahir di Kota Tasikmalaya pada tanggal 18 November 2002. Pendidikan tingkat dasar hingga menengah ditempuh di Bandung. Melanjutkan Sekolah Menengah Kejuruan Di SMK Bandung dan mengambil jurusan Rekayasa Perangkat Lunak . Mulai mengenal Dunia IT sejak SMK, melatih kemampuan pada saat di bangku SMK dan mengikuti eskul Basket , memiliki hobi bermain game dan bermain futsal . Melanjutkan pendidikan di Universitas Logistik Dan Bisnis Internasional dan mengambil Program Studi D4 Teknik Informatika karena meneruskan ilmu yang sudah didapat pada saat SMK . Saya mengambil jurusan Teknik Informatika karena saya menyukai beberapa hal yang dipelajari dari bidang IT, dan ingin mendapatkan pekerjaan sebagai Software Engineering, Web Developer.

A bar code on a white background

Description automatically generatedA black and white rectangular object with text

Description automatically generatedA picture containing icon

Description automatically generated