



# certificate of Completion

Hereby awarded to

**VIVI AISYAH**

For being a student in  
**Studi Independen MIKTI - MSIB 6**

Period  
**February - August 2022**

Pursuing Learning Path

Golang Backend Developer Dengan Pendekatan Computer Vision Dan Artificial Intelligence (AI)>

Your hard work, dedication, and pursuit of excellence have set a high standard and are a testament to your potential and bright future. We commend you on this significant achievement.

Jakarta, 25 Juni 2024

**Chairperson of MIKTI**

**Ibnu Sina Wardy**

# Certificate of Completion

No.	Kompetensi	Deskripsi	Jam	Nilai
1.	<b>Pengenalan pada Dunia Pemrograman</b>	<ol style="list-style-type: none"> <li>Mengidentifikasi dan menjelaskan tahapan-tahapan dalam siklus hidup pengembangan perangkat lunak (SDLC)</li> <li>Menjelaskan pentingnya dokumentasi yang baik dalam pengembangan perangkat lunak dan cara mengelola dan mempertahankan dokumen-dokumen proyek</li> <li>Mempelajari pendekatan sistematis dalam pemecahan masalah dengan mengidentifikasi masalah, menganalisis akar penyebabnya, dan merancang solusi yang efektif.</li> </ol>	10	90
2.	<b>Pengenalan Dasar Internet</b>	<ol style="list-style-type: none"> <li>Memahami konsep dasar infrastruktur Internet.</li> <li>Mengartikan konsep dan fungsi Domain Name System (DNS).</li> <li>Memahami konsep dasar Hypertext Transfer Protocol (HTTP)</li> <li>Memahami konsep hosting dan peran penyedia layanan hosting</li> <li>Menjelaskan tugas-tugas yang terkait dengan pengelolaan hosting, seperti pengaturan DNS, pengelolaan file, dan database</li> </ol>	10	90
3.	<b>Sistem Operasi dan Terminal</b>	<ol style="list-style-type: none"> <li>Memahami definisi dan peran sistem operasi</li> <li>Menggunakan perintah-perintah dasar terminal seperti navigasi direktori, membuat dan menghapus direktori/file, menyalin/memindahkan file.</li> <li>Memahami konsep pengelolaan memori</li> </ol>	10	90
4.	<b>Version Control System (VCS)</b>	<ol style="list-style-type: none"> <li>Memahami konsep dasar Version Control</li> <li>Menggunakan perintah dasar Git untuk mengelola repositori lokal</li> <li>Menggunakan perintah-perintah dasar Git seperti commit, push, pull, dan merge.</li> <li>Memahami konsep dasar dari platform GitHub.</li> <li>Mengelola repository Git melalui GitHub, termasuk membuat repository baru, mengkloning repository, dan mengirim perubahan melalui pull request.</li> </ol>	10	90

# Certificate of Completion

No.	Kompetensi	Deskripsi	Jam	Nilai
5.	<b>Pemrograman Dasar dengan Golang</b>	<ol style="list-style-type: none"> <li>Memahami proses instalasi dan konfigurasi lingkungan pengembangan Go.</li> <li>Mengenal tipe data dasar dalam Go dan dapat menggunakannya dalam kode.</li> <li>Memahami dan menggunakan konsep variabel dalam Go.</li> <li>Menggunakan kontrol alur (control flow) untuk mengatur jalannya program.</li> <li>Memahami dan menerapkan konsep fungsi dalam Go 6. Memahami dan menggunakan struktur data dasar dalam Go.</li> </ol>	10	90
6.	<b>Pemrograman Golang Lanjutan</b>	<ol style="list-style-type: none"> <li>Memahami konsep dan penggunaan defer, panic, dan recover dalam Go.</li> <li>Memahami konsep dan perbedaan antara struct dan interface dalam Go.</li> <li>Memahami konsep pointer dalam Go dan perbedaannya dengan nilai-nilai langsung.</li> <li>Memahami konsep dan struktur package dalam Go.</li> </ol>	10	90
7.	<b>Software Design Dasar pada Golang</b>	<ol style="list-style-type: none"> <li>Mengidentifikasi dan menggambarkan elemen-elemen utama dalam desain perangkat lunak.</li> <li>Memahami pentingnya clean code dalam pengembangan perangkat lunak.</li> <li>Memahami berbagai paradigma pemrograman yang relevan dalam pengembangan perangkat lunak.</li> <li>Memahami prinsip-prinsip SOLID (Single Responsibility, Open-Closed, Liskov Substitution, Interface Segregation, Dependency Inversion) dan pentingnya menerapkannya dalam desain perangkat lunak.</li> </ol>	10	90
8.	<b>Memulai Aplikasi Back-End Sederhana dengan Golang</b>	<ol style="list-style-type: none"> <li>Memahami konsep dasar API dan kegunaannya dalam pengembangan perangkat lunak.</li> <li>Mengidentifikasi dan menjelaskan jenis-jenis API yang umum digunakan, seperti RESTful API, SOAP API, dan GraphQL API.</li> <li>Menjelaskan konsep seperti endpoint, metode HTTP, dan format data yang umum digunakan dalam API.</li> <li>Menggunakan framework Gin untuk membuat dan mengelola RESTful API menggunakan Go.</li> <li>Menggunakan alat bantu untuk menghasilkan dokumentasi API, seperti Swagger atau Postman.</li> </ol>	10	90

# Certificate of Completion

No.	Kompetensi	Deskripsi	Jam	Nilai
9.	<b>Database pada Aplikasi Back-End Golang</b>	<ol style="list-style-type: none"> <li>Memahami konsep dasar database, termasuk struktur data, relasi antar data, dan kebutuhan penyimpanan data dalam aplikasi.</li> <li>Menggunakan GORM (Go Object Relational Mapping) sebagai penghubung antara Go dan database relasional untuk melakukan operasi CRUD.</li> <li>Menggunakan MongoDB sebagai database NoSQL untuk menyimpan dan mengelola data dalam aplikasi Go.</li> <li>Mengintegrasikan fungsi CRUD dengan operasi database relasional dalam API.</li> </ol>	10	90
10.	<b>Authentication and Authorization pada Aplikasi Back-End Golang</b>	<ol style="list-style-type: none"> <li>Memahami konsep dasar autentikasi dan otorisasi dalam pengembangan aplikasi web.</li> <li>Mampu mengimplementasikan autentikasi berbasis JWT dalam aplikasi web.</li> <li>Mampu mengimplementasikan aturan otorisasi berbasis peran dalam aplikasi web.</li> <li>Memahami konsep session dan cookie dalam konteks autentikasi dan otorisasi.</li> </ol>	10	90
11.	<b>Caching pada Aplikasi Back-End Golang</b>	<ol style="list-style-type: none"> <li>Mengidentifikasi keuntungan dan tujuan penggunaan caching.</li> <li>Mengenal beberapa teknik caching pada sisi klien seperti local storage, session storage, dan cookie caching.</li> <li>Menerapkan caching server-side dengan Redis menggunakan Go sebagai bahasa pemrograman.</li> <li>Mengoptimalkan penggunaan caching untuk mengurangi beban server dan meningkatkan performa aplikasi.</li> <li>Menjelaskan konsep CDN dan manfaatnya dalam pengiriman konten.</li> </ol>	20	90
12.	<b>Web Security pada Aplikasi Back-End</b>	<ol style="list-style-type: none"> <li>Memahami jenis-jenis risiko yang dapat mempengaruhi keamanan aplikasi web, seperti serangan injeksi, serangan cross-site scripting (XSS), serangan cross-site request forgery (CSRF), dan serangan brute force.</li> <li>Mampu mengkonfigurasi server web menggunakan HTTPS, termasuk menghasilkan sertifikat SSL/TLS dan mengaktifkan enkripsi data.</li> <li>Mampu mengimplementasikan fungsi hashing dalam Golang untuk mengamankan penyimpanan password dan data sensitif lainnya.</li> <li>Mampu mengidentifikasi kerentanan yang terkait dengan setiap risiko keamanan dan mengimplementasikan langkah-langkah perlindungan yang dianjurkan oleh OWASP.</li> <li>Mengkonfigurasi server web menggunakan SSL/TLS, termasuk menghasilkan sertifikat, mengaktifkan protokol yang aman, dan mengatur preferensi kriptografi.</li> </ol>	20	90

# Certificate of Completion

No.	Kompetensi	Deskripsi	Jam	Nilai
13.	<b>Testing pada Aplikasi Back-End</b>	<ol style="list-style-type: none"> <li>Memahami pentingnya pengujian dalam pengembangan perangkat lunak.</li> <li>Mampu melakukan testing terhadap fungsi-fungsi pada back-end menggunakan Go.</li> <li>Memahami perbedaan antara integration testing dan functional testing.</li> <li>Mampu menggunakan mocking framework seperti "gomock" atau "mockery" dalam pengujian Go.</li> <li>Mampu menjelaskan pentingnya test coverage dalam pengujian perangkat lunak.</li> <li>Memahami konsep benchmarking dan mampu mengimplementasikan benchmarking pada fungsi-fungsi kunci dalam back-end menggunakan Go.</li> </ol>	20	90
14.	<b>Architectural Patterns</b>	<ol style="list-style-type: none"> <li>Memahami konsep dasar tentang arsitektur perangkat lunak dan peran arsitektur dalam pengembangan backend.</li> <li>Memahami konsep dasar dari pola arsitektur Model-View-Controller (MVC) dan bagaimana menerapkannya dalam pengembangan aplikasi monolitik.</li> <li>Menjelaskan keuntungan dan tantangan dalam menggunakan arsitektur microservices.</li> <li>Memahami konsep serverless computing dan peranannya dalam pengembangan backend.</li> <li>Menerapkan Clean Architecture dalam pengembangan aplikasi backend menggunakan Golang.</li> </ol>	20	90
15.	<b>Continuous Integration and Deployment (CI/CD)</b>	<ol style="list-style-type: none"> <li>Memahami konsep dasar tentang CI/CD.</li> <li>Mampu mengkonfigurasi dan mengintegrasikan environment pendukung.</li> <li>Mampu mengimplementasikan konsep CI/CD.</li> <li>Mampu menerapkan konsep CI/CD pada Backend-API.</li> </ol>	20	90

# Certificate of Completion

No.	Kompetensi	Deskripsi	Jam	Nilai
16.	<b>Background Job</b>	<ol style="list-style-type: none"> <li>1. Mampu menjelaskan konsep Background Job dalam konteks pengembangan aplikasi web.</li> <li>2. Mampu menggambarkan manfaat dari penggunaan Background Job dalam meningkatkan performa dan skalabilitas aplikasi.</li> <li>3. Menjelaskan langkah-langkah yang diperlukan untuk mengkonfigurasi dan mengatur sistem antrian di dalam aplikasi berbasis Echo.</li> <li>4. Memahami konsep dasar Publisher/Subscriber (Pub/Sub) dalam arsitektur aplikasi.</li> <li>5. Mempraktikkan integrasi Background Job dengan kasus nyata di dalam aplikasi berbasis Golang.</li> </ol>	20	90
17.	<b>Deployment Scenario</b>	<ol style="list-style-type: none"> <li>1. Memahami konsep dasar lingkungan deployment dalam konteks backend development.</li> <li>2. Menggunakan alat dan teknik untuk mengkonfigurasi dan mengelola dependensi aplikasi.</li> <li>3. Memahami konsep Continuous Integration (CI) dan Continuous Deployment (CD).</li> <li>4. Mengimplementasikan mekanisme logging untuk melacak kejadian penting dan debugging aplikasi.</li> </ol>	20	90
18.	<b>Containerization</b>	<ol style="list-style-type: none"> <li>1. Menjelaskan penggunaan container dalam pengembangan perangkat lunak.</li> <li>2. Menginstal Docker pada sistem operasi yang relevan dan Menggunakan perintah dasar Docker untuk mengelola container.</li> <li>3. Menjalankan aplikasi Golang dalam container Docker.</li> <li>4. Menjelaskan konsep dan kegunaan Docker Compose.</li> <li>5. Memahami konsep dasar container orchestration.</li> <li>6. Menjelaskan praktik terbaik dalam containerization dengan Golang.</li> <li>7. Mempersiapkan implementasi containerization pada studi kasus backend development dengan Golang.</li> </ol>	20	90

# Certificate of Completion

No.	Kompetensi	Deskripsi	Jam	Nilai
19.	<b>Cloud Computing Service</b>	1. Menjelaskan keuntungan dan tantangan penggunaan cloud computing. 2. Menjelaskan konsep dan keuntungan menggunakan Backend as a Service (BaaS), Platform as a Service (PaaS), Infrastructure as a Service (IaaS). 3. Mengidentifikasi jenis-jenis penyedia layanan cloud (Cloud Service Providers). 4. Menggunakan alat dan layanan yang tersedia untuk deployment aplikasi Golang ke Cloud. 5. Memahami konsep dan prinsip dasar manajemen layanan cloud.	20	90
20.	<b>Use Case Image Recognition System</b>	1. Memahami peran TensorFlow dalam pengembangan aplikasi Computer Vision. 2. Menginstal dan mengkonfigurasi TensorFlow dan Golang. 3. Menerapkan mekanisme interaksi antara Front-End dan Back-End melalui API untuk penggunaan aplikasi Computer Vision. 4. Memahami strategi deployment untuk aplikasi Back-End dengan implementasi Computer Vision dan TensorFlow.	20	90
21.	<b>Capstone Project</b>		600	90
Total/Rerata			900	90

Jakarta, 25 Juni 2024

**Chairperson of MIKTI**




**Ibnu Sina Wardy**