

LAPORAN AKHIR
MAGANG & STUDI INDEPENDEN BERSERTIFIKAT

NodeJS For Back End Web Developer
Studi Independen
Di PT Hactivate Teknologi Indonesia

Maulana Daffa Ardiansyah
123200130



INFORMATIKA
UNIVERSITAS PEMBANGUNAN NASIONAL "VETERAN"
YOGYAKARTA
2023

Kata Pengantar

Puji syukur kepada Tuhan Yang Maha Esa atas rahmat dan karunia-Nya yang dilimpahkan dalam kegiatan studi independen sehingga dapat menyelesaikan program NodeJS for Back End Web Developer.

Laporan akhir ini disusun sebagai bukti telah menyelesaikan kegiatan studi independen dari program MSIB yang diselenggarakan oleh Kampus Merdeka. Rasa terima kasih yang besar ditujukan kepada Bapak Nadiem Anwar Makarim, B. A, M. B. A. selaku Menteri Pendidikan, Kebudayaan, Riset, dan Teknologi, Ibu Vynska Amalia Permadi S. Kom., M. Kom. selaku koordinator KP program studi Informatika, Lutfi Dwimulya selaku PIC Kampus Merdeka PT Hactivate Teknologi Indonesia, Arif Susanto selaku mentor di kelas INJS-KS06, dan seluruh rekan-rekan kelas serta proyek yang telah membantu kelancaran dan pelaksanaan studi independen.

Laporan ini tentu masih memiliki banyak kekurangan dan jauh dari kata sempurna baik dari tata bahasa maupun penulisan. Oleh karena itu, kritik dan saran akan sangat membantu dalam mengembangkan laporan akhir ini.

Yogyakarta, 31 Desember 2023



Maulana Daffa Ardiansyah

Daftar Isi

Kata Pengantar	
Daftar Isi	
Daftar Lain-Lain	
Bab I Gambaran Umum	1
I.1 Profil Perusahaan	1
I.2 Deskripsi Kegiatan	1
Bab II Aktivitas Mingguan	3
Bab III Penutup	7
III.1 Kesimpulan	7
III.2 Saran	8
Referensi	
Lampiran	

Daftar lain-lain

Gambar 1

Gambar 2

Gambar 3

Gambar 4

Bab I Gambaran Umum

I.1 Profil Perusahaan

Hactiv8 mempersiapkan profesional untuk siap menghadapi karir di bidang IT.

I.2 Deskripsi Kegiatan

Posisi : NodeJs for Back End Web Development

Deskripsi : Pada *Course* ini, user akan dibekali dengan pengetahuan mengenai Nodejs serta Expressjs hingga Implementasi REST API menggunakan PostgreSQL hingga *Testing* dan *Deployment*, mengoptimalkan aplikasi dengan peningkatan performa, hingga ke *Deployment* pada Railway. Diharapkan setelah mengikuti Program ini *learner* akan menjadi seorang *Back End Web Developer* yang siap menghadapi dunia kerja di bidang *Web Development*. Untuk mengambil program ini, peserta diharapkan telah memahami kemampuan dasar komputer (web browsing, app installation, file navigating).

Kompetensi yang dikembangkan :

1. Fundamental back end web development
2. Mendesain dan membuat database
3. Memahami dan mengimplementasikan framework NodeJs
4. Mengamankan dan mengatur user authentication dan access control
5. Debugging dan optimalisasi kinerja web app

Program studi independen NodeJs for Back End Developer terdiri dari 2 jenis kegiatan, belajar mandiri (*self-paced learning*) dan dilanjutkan dengan menyelesaikan proyek akhir secara berkelompok.

Kegiatan belajar mandiri dilakukan melalui platform kode.id yang telah disediakan oleh Hactiv8. Kegiatan belajar mandiri tersebut difasilitasi oleh Hactiv8 dengan memberikan 3 *course* gratis yang berjudul NodeJs Tingkat Pemula, Menguasai PostgreSQL, dan Melakukan Deployment dengan NodeJs. Ketiga *course* tersebut diselesaikan secara mandiri sebelum melanjutkan pengerjaan proyek akhir.

Pada saat kegiatan belajar mandiri juga dijadwalkan untuk mengikuti *live session* bersama mentor. *Live session* membahas mengenai materi-materi yang ada pada *course* dan diajarkan bagaimana cara mengimplementasikan materi tersebut.

Setelah semua materi belajar mandiri terselesaikan, pembagian kelompok untuk proyek akhir dilakukan dan didapatkan dua anggota untuk tiap kelompok. Masing-masing kelompok akan mengerjakan 4 proyek akhir dengan ketentuan yang berbeda-beda tiap proyeknya.

Proyek pertama yaitu pembuatan Reflection API yang dibuat dengan ExpressJs dan PostgreSQL dimana proses CRUD harus dilakukan dengan mengeksekusi Query untuk setiap fiturnya. Proyek kedua yaitu membuat API media social berupa MyGram, dimana aplikasi tersebut dibuat untuk mengunggah foto dan memberikan komentar untuk setiap foto yang ada. Perbedaan proyek kedua ini dari yang pertama yaitu proses autentifikasi dan authorisasi yang lebih ketat serta mulai pada proyek kedua dan seterusnya, ORM harus diimplementasikan sehingga tidak perlu lagi menulis dan mengeksekusi Query.

Proyek ketiga berupa API untuk *e-commerce* dimana terdapat daftar kategori dan barang, serta jumlah stok yang tersedia dan nominal uang virtual yang dimiliki pengguna. Pada proyek ini pembagian *role* diberikan, sehingga ada beberapa fitur yang hanya bisa diakses oleh pengguna dengan *role* tertentu. Kemudian untuk proyek keempat yaitu melanjutkan aplikasi MyGram pada proyek kedua dan membuat *script testing* menggunakan *package* JEST dan Supertest yang tersedia pada JavaScript. Ketentuan proyek ini yaitu dengan membuat minimal 5 pengecekan pada tiap testing yang dijalankan.

Bab II Aktivitas Mingguan

Minggu	Kegiatan
1	Saya belajar banyak pada minggu ini di program msib. Mulai dari soft skill hingga hard skill. Pada studi independen di Hacktiv8 ini, diberikan materi mengenai Engineering Empathy, dimana saya diajarkan bagaimana menjadi orang yang adaptif dan memiliki growth mindset. Sedangkan untuk hard skill, materi yang diberikan mulai dari proses instalasi NodeJS hingga basic-basis dari NodeJs tersebut. Ada pula tugas yang diberikan yaitu untuk membuat summary dari beberapa video yang ada di course.
2	Minggu ini saya melanjutkan pembelajaran pada pre-requisite course yaitu NodeJs Tingkat Pemula. Pada sisa materi di course tersebut, saya mempelajari mengenai framework ExpressJs dan dikenalkan pada Template Engine berupa Embedded JavaScript atau Ejs. Setelah semua materi selesai pada course tersebut, diberikan contoh pembuatan website mengenai cuaca dimana kita mengambil data dari API dan ditampilkan pada website. Website tersebut dibuat dengan mengimplementasikan ExpressJs dan Ejs kemudian ditambahkan dengan penggunaan CSS agar website memiliki tampilan yang lebih menarik. Pada pembuatan website tersebut diberikan demo pembuatan yang cukup jelas dari awal hingga website tersebut selesai dibuat.
3	Pada minggu ini, live session baru dimulai karena terdapat kendala pada minggu sebelumnya. Live session dimulai mengikuti materi yang sudah ada di course pada website kode.id. Pada sesi pertama ini mentor memberikan pengenalan mengenai apa itu NodeJs, bagaimana cara menginstallnya, dan bagaimana cara menjalankan file JavaScript pada NodeJs. Kemudian pada sesi kedua, mentor mengajarkan mengenai dasar-dasar dari NodeJs berupa Event Handler, Buffer Streams, dan diberikan materi tambahan berupa CLI Exploration dimana hal ini akan sangat berguna sebagai Back-End Developer yang akan berurusan dengan server dan database, terutama pada database PostgreSQL yang akan dipelajari mulai pada pertemuan ke-7.
4	Minggu ini, saya mempelajari mengenai file system pada NodeJs dan bagaimana cara membuat server tanpa menggunakan framework dengan package bawaan pada NodeJs yaitu HTTP server. Setelah mempelajari HTTP server, kemudian mentor mengajarkan mengenai framework yang biasa digunakan pada

	back-end JavaScript berupa ExpressJs di pertemuan berikutnya. Mentor mengajarkan mulai dari bagaimana cara menginstall ExpressJs hingga pembuatan server dan routing pada NodeJs dengan memanfaatkan framework tersebut.
5	Pada pertemuan hari senin mentor mengajarkan mengenai file statis dan middleware yang kemudian pada pertemuan hari kamis dilanjutkan dengan Template Engine dan Querystring. Diberikan contoh oleh mentor dalam membuat website dengan memanfaatkan materi-materi pada pertemuan sebelumnya yaitu routing dan middleware yang kemudian dikombinasikan dengan Template Engine Ejs. Diajarkan juga cara menggunakan GET dan POST parameter pada Ejs dalam pembuatan website.
6	Materi pada minggu ini sudah tidak lagi membahas mengenai framework ataupun template engine. Mentor mulai melanjutkan materi mengenai Database menggunakan PostgreSQL. Dimulai dari membahas hal-hal dasar seperti Relational DataBase Management System atau RDBMS hingga menginstall sendiri PostgreSQL pada perangkat masing-masing.
7	Pada minggu ini, dipelajari tools yang dapat mempermudah dalam pengelolaan database sql pada PostgreSQL yaitu PgAdmin. Tools ini memiliki interface yang mudah dipahami sehingga tidak hanya berisi tulisan saja, dimana dalam interface tersebut juga dapat memperlihatkan isi dari database dan table yang ada. Setelah belajar mengenai PgAdmin, kemudian pembelajaran dilanjutkan dengan memahami Rest API, dimana hal tersebut tidak berkaitan langsung dengan database PostgreSQL.
8	Minggu ini mempelajari mengenai REST API dan bagaimana cara melakukan pembuatan, pembacaan, update, dan penghapusan pada endpoint sebuah API. API yang dipelajari yaitu API dengan bentuk JSON dimana data yang ada pada API didapatkan dari sebuah file .JSON yang ada pada folder. Kemudian bagaimana melakukan CRUD pada data di endpoint tersebut.
9	Pada minggu kesembilan ini, materi yang dipelajari yaitu mengenai Validasi dan Testing. Validasi yang dimaksud adalah pemberian sebuah token menggunakan package JWT pada saat login berhasil, serta validasi data menggunakan Sequelize. Pada materi testing, digunakan package JEST pada JavaScript yang berguna untuk mengetes sebuah fungsi/unit. Kedua materi ini menurut saya sangat

	menarik untuk dipejari. Namun, pada minggu ini juga terdapat tugas untuk masing-masing materi. Pada weekend saya baru dapat menyelesaikan tugas pertama yaitu pembuatan API dan authorization menggunakan JWT dan memanfaatkan tools berupa Postman. Untuk tugas kedua baru saya kerjakan pada minggu berikutnya.
10	Pada minggu ini waktu kebanyakan dihabiskan untuk mengerjakan tugas berupa unit testing, dimana pada tugas tersebut diperintahkan untuk menulis kode yang berguna untuk melakukan testing API dari sebuah aplikasi website yang telah dibuat pada sesi sebelumnya. Selain itu, diperkenalkan juga dua tools baru yaitu Travis-Ci yang digunakan untuk melakukan testing dari aplikasi yang sudah terintegrasi dan Railway yang digunakan untuk melakukan deployment secara gratis.
11	Pada minggu ini, pengerjaan Final Project 1 berupa Reflection API dimulai. Endpoint-endpoint dibagi menjadi dua bagian untuk dikerjakan bersama dengan rekan kelompok. Setelah mencoba untuk mengerjakan proyek tersebut, terdapat beberapa kendala. Untuk itu, dilanjutkan dengan melakukan mentoring dengan mentor pada hari Jum'at untuk mendapatkan solusi dari kendala yang dialami.
12	Menyelesaikan dan menyempurnakan aplikasi pada Final Project 1. Kemudian dilanjutkan untuk mengikuti meet yang membahas progress masing-masing kelompok dan briefing mengenai Final Project selanjutnya. Setelah itu menginisiasi pengerjaan Final Project 2 dengan membuat repository di GitHub dan menambahkan register dan login API.
13	Minggu ini saya manfaatkan untuk menyelesaikan Final Project 2 sesuai dengan tugas masing-masing anggota kelompok. Tugas saya yaitu untuk membuat fungsi dari fitur/endpoint sesuai dengan ketentuan Final Project 2. Pada project ini mulai dimanfaatkan ORM yang mana pada Final Project 1 belum digunakan. Penggunaan ORM sangat membantu agar dalam pengiriman data ke dalam database dapat meminimalisir terjadinya kesalahan syntax karena tidak perlu menuliskan query.
14	Melakukan finishing terhadap Final Project 2 dan melakukan deployment ke platform Railway. Setelah di deploy, dilakukan testing terlebih dahulu untuk mengecek apakah semua endpoint

	dan fungsi sudah bekerja dengan baik. Kemudian mengsubmit pekerjaan ke dalam Google Classroom. Dilanjutkan dengan mengikuti meet online bersama dengan mentor untuk melakukan briefing terhadap Final Project 3.
15	Minggu ini dimulai dengan mengerjakan Final Project 3. Mulai dari membuat database, ORM Sequelize, dan juga Sequelize Validationnya. Kemudian melanjutkan mengerjakan fungsi setiap endpoint yang ada.
16	Melakukan finishing terhadap Final Project 3 dan melakukan deployment ke platform Railway. Setelah di deploy, dilakukan testing terlebih dahulu untuk mengecek apakah semua endpoint dan fungsi sudah bekerja dengan baik. Kemudian mengsubmit pekerjaan ke dalam Google Classroom. Dilanjutkan dengan mengikuti meet online bersama dengan mentor untuk melakukan briefing terhadap Final Project 4.
17	Memulai pengerjaan pada Final Project 4 dengan menambahkan testing kepada Final Project 2 yang telah dibuat sebelumnya. Karena memiliki beberapa kesibukan dalam mempersiapkan kegiatan KKN, pengerjaan Final Project 4 ini sering saya tunda.
18	Minggu ini saya lalui dengan melanjutkan pengerjaan Final project 4 yang sekaligus menjadi Final Project terakhir.
19	Minggu ini melanjutkan dan melakukan finishing terhadap Final Project 4 yang merupakan final project terakhir pada program studi independent. Pada penyelesaian dilakukan mentoring bersama dengan mentor untuk menyelesaikan kendala yang dialami. Kemudian dilanjutkan dengan mengikuti meet terakhir bersama mentor sebagai bentuk perpisahan dan mengucapkan terimakasih karena telah membimbing selama program studi independent.
20	Minggu ini mulai mengerjakan laporan akhir Kampus Merdeka dan harapannya sebelum minggu depan laporan telah selesai.

Bab III Penutup

III.1 Kesimpulan

Proses pelaksanaan sebagai murid pada studi independen di PT Hactivate Teknologi Indonesia pada program NodeJs for Back End Web Development telah memberikan pengetahuan yang sangat berharga. Pembelajaran ini mencakup dua aspek utama, yakni pengetahuan teoretis dan praktis yang masing-masing membuka wawasan baru dalam dunia pengembangan web.

Pertama, pengetahuan teoretis yang diperoleh selama melakukan pembelajaran mandiri memberikan pemahaman yang mendalam tentang berbagai komponen penyusun back-end dari suatu *website*. Materi teoretis ini mencakup konsep dasar NodeJs, manajemen basis data, keamanan, dan arsitektur server. Dengan pemahaman ini, dapat dilihat gambaran menyeluruh tentang bagaimana server beroperasi dan berinteraksi dengan komponen-komponen lain dalam ekosistem web.

Kedua, pengalaman praktis yang diperoleh dari pengerjaan proyek akhir sangat berharga dalam mengembangkan keterampilan logika dan alur program. Proyek akhir mengharuskan untuk mengaplikasikan konsep-konsep teoretis yang telah dipelajari ke dalam proyek nyata. Ini tidak hanya memperkuat pemahaman terhadap konsep-konsep tersebut, tetapi juga membantu mengasah keterampilan praktis dalam menangani tantangan pengembangan web sehari-hari.

Melalui kombinasi pengetahuan teoretis dan pengalaman praktis ini, dapat membantu meningkatkan kesiapan untuk terjun ke dunia industri pengembangan web back-end. PT Hactivate Teknologi Indonesia dengan program NodeJs for Back End Web Development memberikan fondasi yang kokoh, mencakup aspek-aspek penting yang dibutuhkan untuk berhasil dalam peran sebagai pengembang back-end yang kompeten. Seluruh proses ini telah memberikan landasan yang kuat dan keyakinan diri dalam menghadapi proyek-proyek pengembangan web di masa depan.

III.2 Saran

Secara garis besar, pengalaman yang dialami dalam program NodeJs for Back End Web Development di PT Hactivate Teknologi Indonesia telah sangat baik. Materi teoretis dan praktis yang diberikan memberikan pemahaman mendalam tentang pengembangan web back-end. Namun, seiring perjalanan, masih terdapat ruang untuk pengembangan.

Salah satunya dalam hal pembagian masing-masing peran di antara berbagai pihak yang terlibat, seperti DPP, mentor, PIC, dan lainnya. Terasa sedikit membingungkan mengenai peran dari pihak tersebut, yang kedepannya agar bisa lebih diperjelas lagi.

Satu hal lagi yang menjadi perhatian saya yaitu pada platform pembelajaran yang disediakan oleh PT Hactivate Teknologi Indonesia, kode.id. Platform tersebut tentu sangat membantu dan mempermudah proses pembelajaran secara mandiri, tetapi pada platform tersebut tidak tersedia *progress bar* yang berguna untuk melakukan *fast-forward* atau *rewind*. Sehingga saat ada materi yang terlewat atau ingin diulang, harus memulai *video* dari awal.

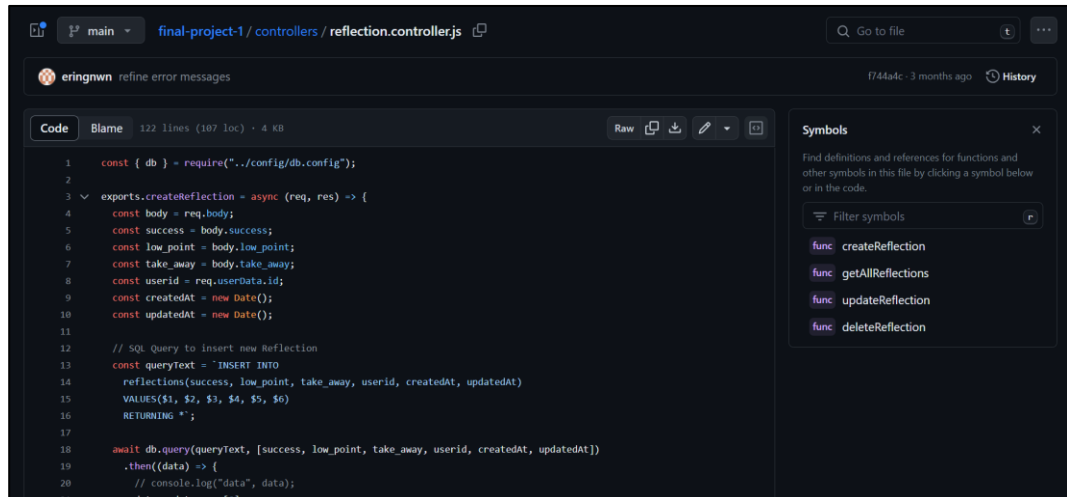
Referensi

<https://kampusmerdeka.kemdikbud.go.id/>

<https://www.kode.id/>

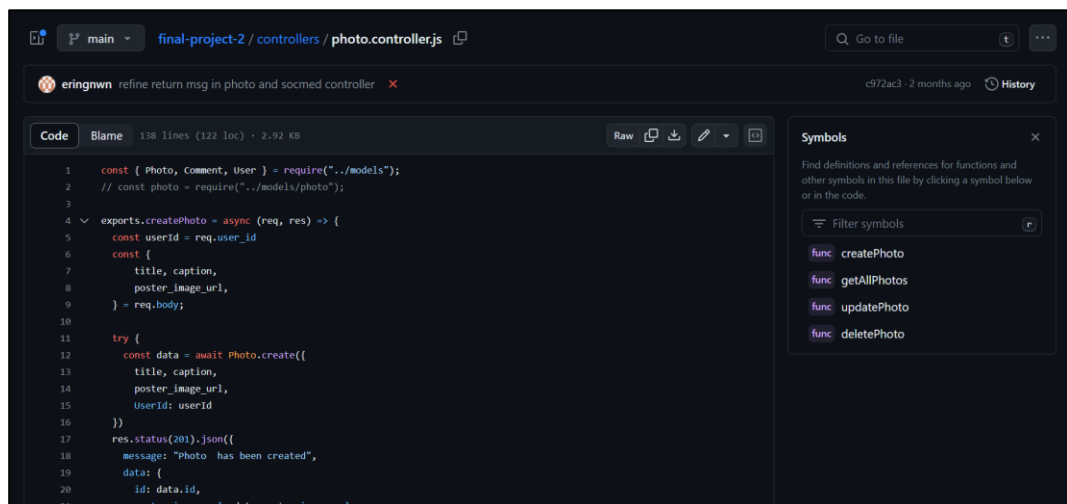
<https://www.hacktiv8.com/>

Lampiran



```
1  const { db } = require("../config/db.config");
2
3  exports.createReflection = async (req, res) => {
4    const body = req.body;
5    const success = body.success;
6    const low_point = body.low_point;
7    const take_away = body.take_away;
8    const userId = req.userData.id;
9    const createdAt = new Date();
10   const updatedAt = new Date();
11
12   // SQL Query to insert new Reflection
13   const queryText = `INSERT INTO
14     reflections(success, low_point, take_away, userId, createdAt, updatedAt)
15     VALUES($1, $2, $3, $4, $5, $6)
16     RETURNING *`;
17
18   await db.query(queryText, [success, low_point, take_away, userId, createdAt, updatedAt])
19     .then((data) => {
20       // console.log("data", data);
21       data = data.rows[0];
22     })
23     .catch((err) => {
24       console.log("err", err);
25     })
26     .finally(() => {
27       res.status(201).json({
28         message: "Reflection has been created",
29         data: data
30       });
31     })
32   }
33 }
```

Gambar 1 Kode Proyek Akhir 1



```
1  const { Photo, Comment, User } = require("../models");
2  // const photo = require("../models/photo");
3
4  exports.createPhoto = async (req, res) => {
5    const userId = req.user_id;
6    const {
7      title, caption,
8      poster_image_url,
9    } = req.body;
10
11    try {
12      const data = await Photo.create({
13        title, caption,
14        poster_image_url,
15        userId: userId
16      })
17      res.status(201).json({
18        message: "Photo has been created",
19        data: {
20          id: data.id,
21          poster_image_url: data.poster_image_url
22        }
23      })
24    } catch (err) {
25      console.log("err", err);
26    }
27  }
28 }
```

Gambar 2 Kode Proyek Akhir 2

```
1  const { moneyFormat } = require("../helpers/utills");
2  const { User, Product, Category, TransactionHistory, sequelize } = require("../models");
3
4  exports.postTransaction = async (req, res) => {
5    const productId = req.body.productId;
6    const quantity = Number(req.body.quantity);
7    const userId = req.user_id;
8
9    if (quantity <= 0){
10     return res.status(500).json({ message: "Amount bought must be more than 1" });
11    }
12    try{
13     const product = await Product.findById(productId);
14     if (!product){
15       return res.status(500).json({ message: "Product not found" });
16     }
17
18     const productdata = product.dataValues;
19     const stock = productdata.stock;
20     const totalprice = productdata.price * quantity;
```

Gambar 3 Kode Proyek Akhir 3

```
1  const request = require("supertest")
2  const app = require("../index")
3  const { User, Photo } = require("../models")
4
5  const dataUser1 = {
6    id: 1,
7    full_name: "user1",
8    email: "user1@mail.com",
9    username: "user1",
10    password: "rahasia",
11    profile_image_url: "www.freepik.com",
12    age: 20,
13    phone_number: "081234567890"
14  }
15
16  const dataUser2 = {
17    id: 2,
18    full_name: "user2",
19    email: "user2@mail.com",
20    username: "user2",
21    password: "rahasia",
```

Gambar 4 Kode Proyek Akhir 4