

**Aplikasi Web CRUD Data Mahasiswa  
dengan NodeJs dan MongoDB**



**Kelompok :**

<b>Ara Fasaka</b>	<b>(21120117140014)</b>
<b>Nareswari Dyah P</b>	<b>(21120117120002)</b>
<b>Doni Hermawan</b>	<b>(21120117120009)</b>
<b>Irgantara Arda Pratama</b>	<b>(21120117140015)</b>

**PERANCANGAN BASIS DATA  
DEPARTEMEN TEKNIK KOMPUTER FAKULTAS TEKNIK  
UNIVERSITAS DIPONEGORO  
2020**

## DAFTAR ISI

DAFTAR ISI.....	2
1. PERSIAPAN.....	3
1.1    Instalasi NodeJs.....	5
1.2    MongoDB .....	9
2. Pengerjaan pembuatan web CRUD data mahasiswa dengan NodeJS dan MongoDB.....	14
2.1    Melakukan instalasi library dan konfigurasi project .....	14
3. Tampilan WEB.....	20
3.1    Tampilan awal / add mahasiswa.....	20
3.2    Tampilan daftar mahasiswa.....	21
3.3    Tampilan Edit mahasiswa.....	21

## Daftar Gambar

Gambar 1. 1 Laman unduh NodeJs .....	5
Gambar 1. 2 Tampilan awal instalasi .....	5
Gambar 1. 3 Tempat folder instalasi .....	6
Gambar 1. 4 Custon setup NodejS .....	6
Gambar 1. 5 Instalasi selesai .....	6
Gambar 1. 6 Properties This PC.....	7
Gambar 1. 7 pilih advance system settings\.....	7
Gambar 1. 8 Edit path pada system variable.....	8
Gambar 1. 9 Menambahkan direktori instalasi .....	8
Gambar 1. 10 Versi NodeJs .....	9
Gambar 1. 11mongodb instalasi1 .....	9
Gambar 1. 12 mongodb instalasi2.....	9
Gambar 1. 13 mongodb instalasi3.....	10
Gambar 1. 14 mongodb instalasi4.....	10
Gambar 1. 15 mongodb instalasi5.....	10
Gambar 1. 16 Folder bin MongoDB .....	11
Gambar 1. 17 System Environtment .....	12
Gambar 1. 18 Tab system properties .....	12
Gambar 1. 19 Edit pada path.....	13
Gambar 1. 20 New environtment variable .....	13
Gambar 1. 21 Masukan direktori mongodb .....	13
Gambar 1. 22 npm init .....	14
Gambar 1. 23 Kongfigurasi project.....	14
Gambar 1. 24 express npm.....	14
Gambar 1. 25 npm mongoose .....	15
Gambar 1. 26 Express Handlebars .....	15
Gambar 1. 27 Body-parser .....	15
Gambar 1. 28 npm di visual code.....	15
Gambar 1. 29 hasil npm install .....	15
Gambar 1. 30 Start MongoDB.bat .....	16

Gambar 1. 31 Database mahasiswaDB .....	16
Gambar 1. 32 Susunan arsitektur web .....	17
Gambar 1. 33 Bagian view .....	19
Gambar 1. 34 mengaktifkan port .....	20
Gambar 1. 35 Tampilan Awal/Add mahasiswa .....	20
Gambar 1. 36 Tampilan Awal/Add mahasiswa .....	21
Gambar 1. 37 Tampilan Edit Mahasiswa .....	21

## 1. PERSIAPAN

### 1.1 Instalasi NodeJs

1. Menuju website NodeJs untuk mengunduh instalasi NodeJs  
<https://nodejs.org/en/download/>

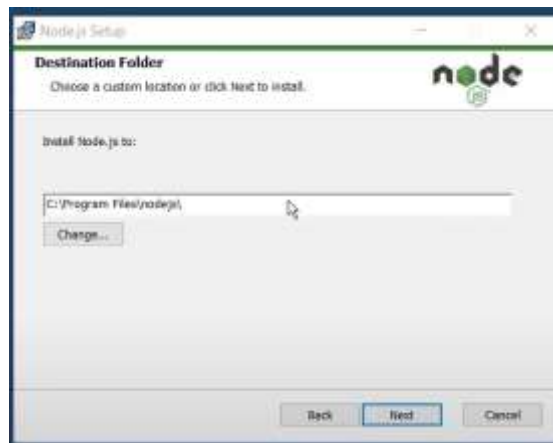


Gambar 1. 1 Laman unduh NodeJs

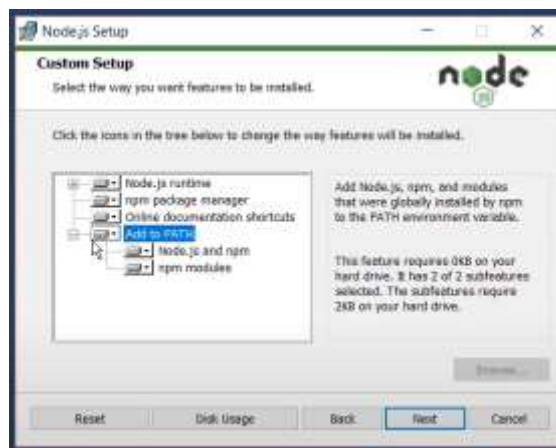
2. Kemudian install seperti biasa, next saja sampai selesai



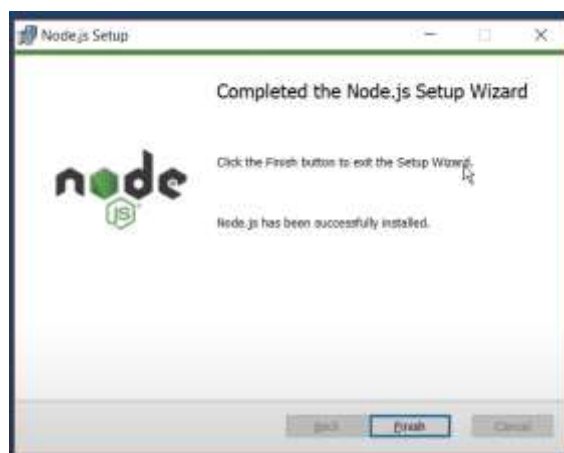
Gambar 1. 2 Tampilan awal instalasi



Gambar 1. 3 Tempat folder instalasi



Gambar 1. 4 Custon setup NodejS



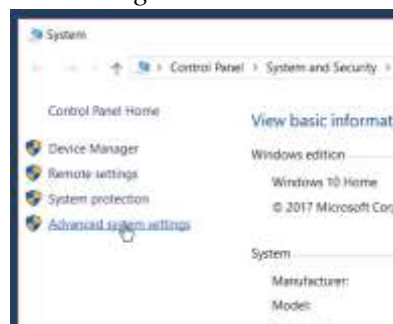
Gambar 1. 5 Instalasi selesai

3. Kemudian memberikan path environment pada windows agar NodeJs terdeteksi dengan IDE nantinya. Pertama buka *File Explorer* lalu properties di “*this PC*”



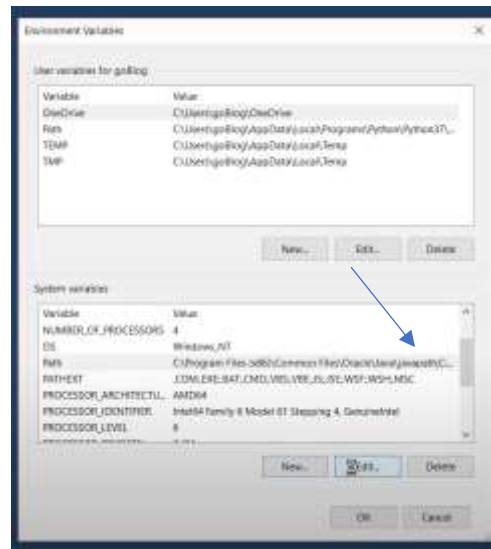
Gambar 1. 6 Properties This PC

4. Lalu ke *advance system setting*



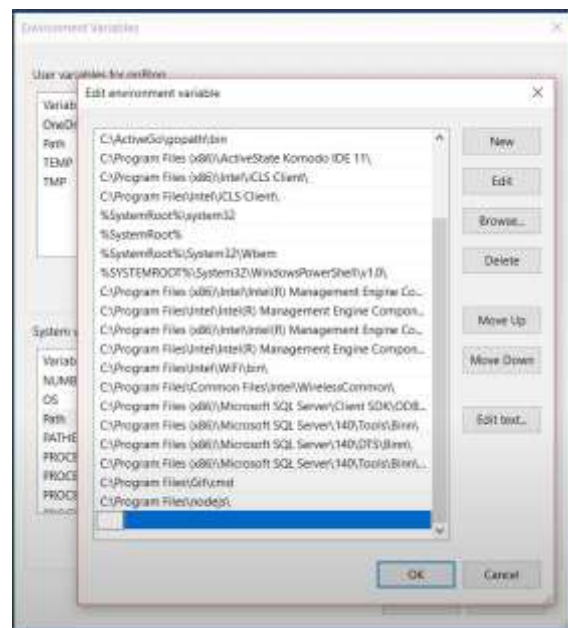
Gambar 1. 7 pilih advance system settings\

5. Klik edit path



Gambar 1. 8 Edit path pada system variable

6. Mengisi direktori instalasi dari NodeJs



Gambar 1. 9 Menambahkan direktori instalasi

7. Cek versi nodejs dengan CMD



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.18362.836]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Asus>node -version
node: bad option: -version

C:\Users\Asus>node --version
v12.16.3

C:\Users\Asus>
```

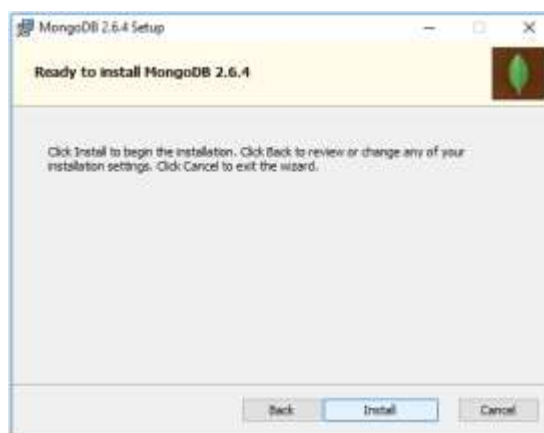
Gambar 1. 10 Versi NodeJs

## 1.2 MongoDB

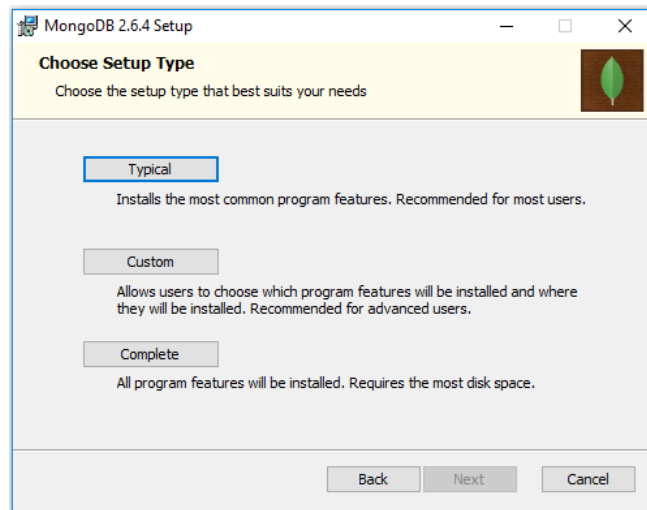
1. Mengunduh MongoDB mendapatkan file installer untuk sistem operasi windows pada website resmi [mongodb.com](https://www.mongodb.com). setelah selesai proses download silahkan double click pada file installer dan ikuti proses instalasi sebagai berikut:



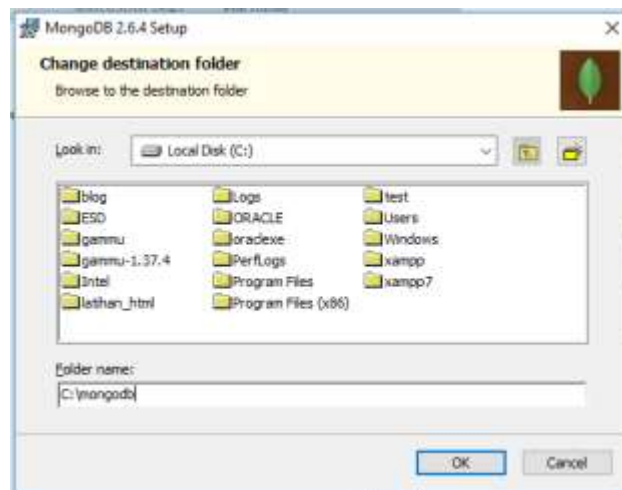
Gambar 1. 11mongodb instalasi1



Gambar 1. 12 mongodb instalasi2



Gambar 1. 13 mongodb instalasi3

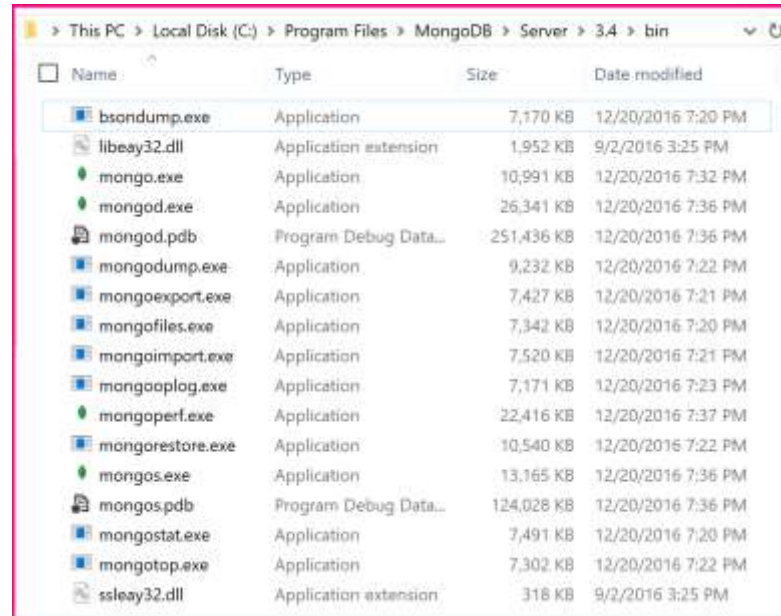


Gambar 1. 14 mongodb instalasi4



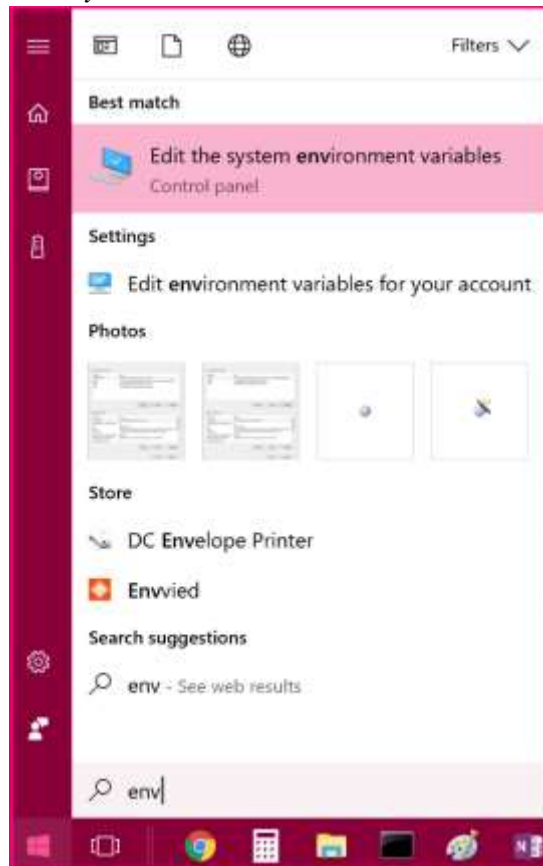
Gambar 1. 15 mongodb instalasi5

2. Menambahkan environment MongoDB pada windows dengan masuk ke bin dari folder instalasi MongoDB

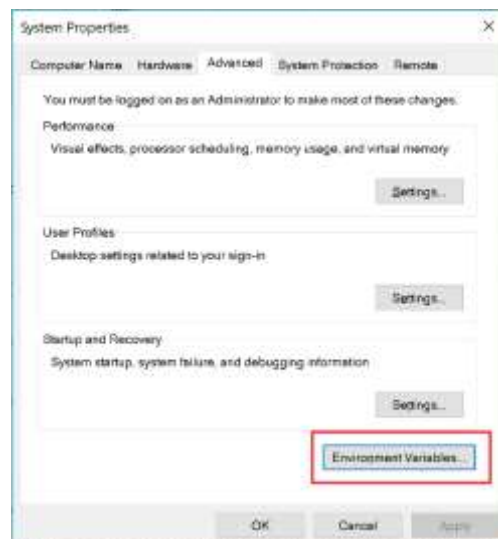


Gambar 1. 16 Folder bin MongoDB

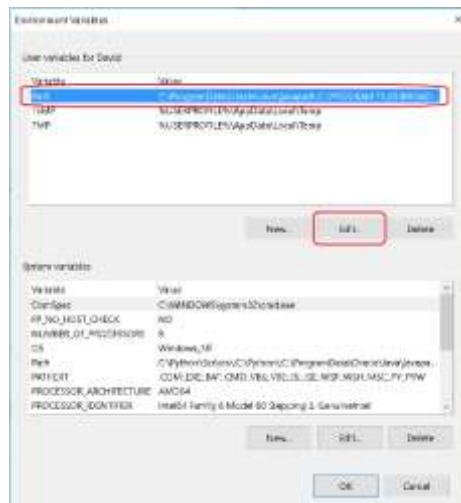
3. Masuk ke *Edit the System Environment Variables*



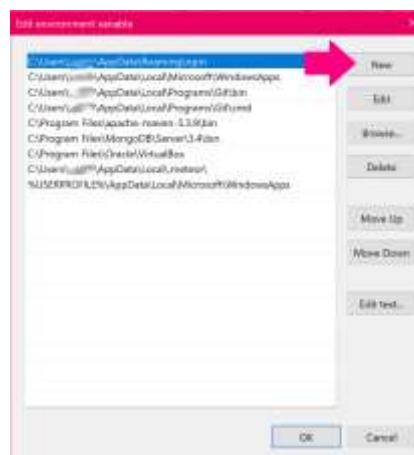
Gambar 1. 17 System Environment



Gambar 1. 18 Tab system properties



Gambar 1. 19 Edit pada path



Gambar 1. 20 New environment variable

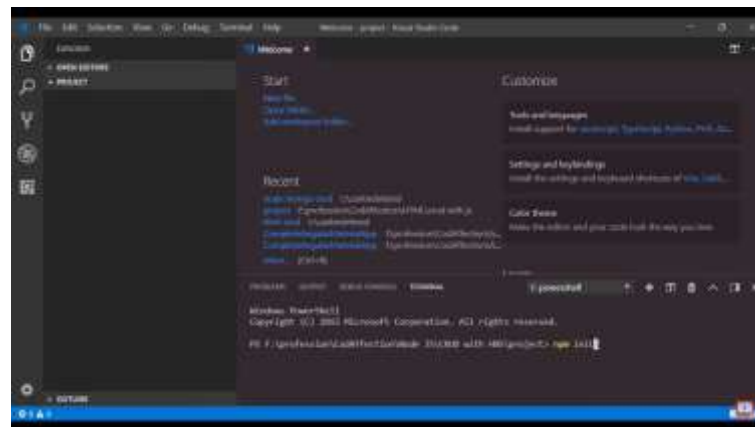


Gambar 1. 21 Masukan direktori mongodb

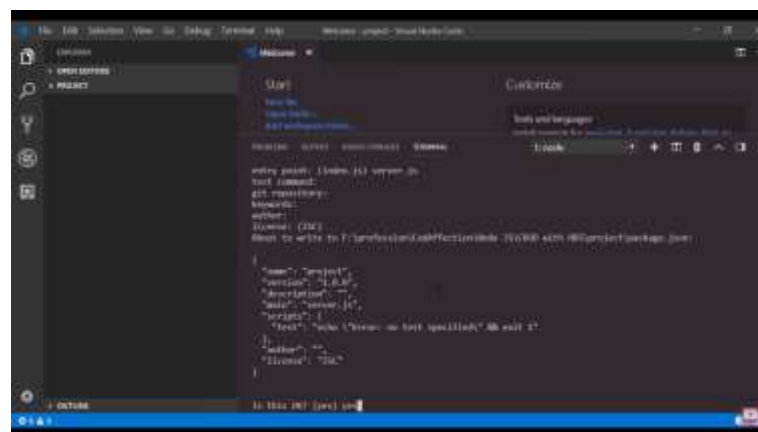
## 2. Pengerjaan pembuatan web CRUD data mahasiswa dengan NodeJS dan MongoDB

### 2.1 Melakukan instalasi library dan konfigurasi project

1. Pada Visual Code membuat project dan melakukan npm init



Gambar 1. 22 npm init



Gambar 1. 23 Konfigurasi project

2. Melakukan npm install



Gambar 1. 24 express npm



Gambar 1. 25 npm mongoose



Gambar 1. 26 Express Handlebars



Gambar 1. 27 Body-parser

```
PS F:\profession\CodAffection\Node JS\CRUD with HMS\project> npm i --s express@4.16.4 mongoose@5.3.4 express-handlebars@3.0.0 body-parser@1.18.3
```

Gambar 1. 28 npm di visual code

```
+ express@4.16.4
+ body-parser@1.18.3
+ express-handlebars@3.0.0
+ mongoose@5.3.4
added 98 packages from 97 contributors in 16.675s
```

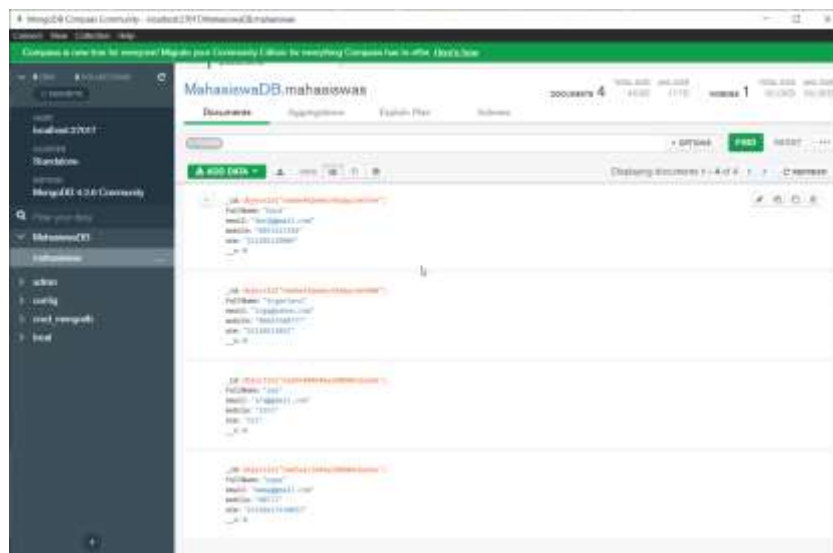
Gambar 1. 29 hasil npm install

3. Kemudian untuk mempermudah dalam melakukan start dari mongoDB, dibuat notepad khusus. Jadi kita hanya perlu klik notepad tersebut nantinya



Gambar 1. 30 Start MongoDB.bat

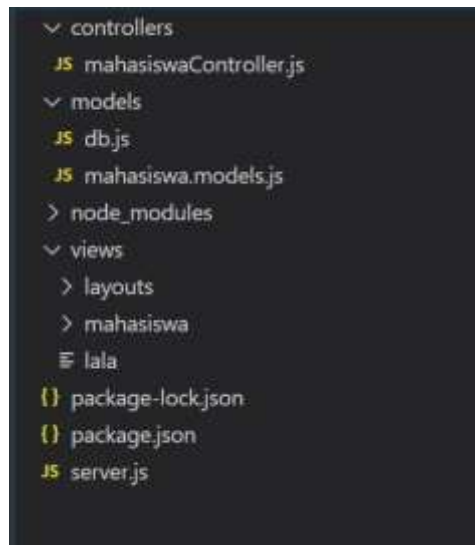
#### 4. Membuat Database MahasiswaDB dan collection mahasiswas



Gambar 1. 31 Database mahasiswaDB

#### 5. Kemudian membuat model MVC yakni Model, View, dan Controller.





Gambar 1. 32 Susunan arsitektur web

Pertama, kita membuat server.js untuk menghubungkan MVC, inialisasi dari npm yang telah diinstall sebelumnya, dan mengatur port untuk membuka web tersebut.

```
require('./models/db');

const express = require('express');
const path = require('path');
const exphbs = require('express-handlebars');
const bodyParser = require('body-parser');

const mahasiswaController = require('./controllers/mahasiswaController');

var app = express();
app.use(bodyParser.urlencoded({
  extended: true
}));

app.use(bodyParser.json());

app.set('views', path.join(__dirname, '/views/'));
app.engine('hbs', exphbs({ extname: 'hbs', defaultLayout:
'mainLayout', layoutsDir: __dirname + '/views/layouts/' }));
app.set('view engine', 'hbs');

app.listen(3000, () => {
  console.log('Express server started at port : 3000');
});

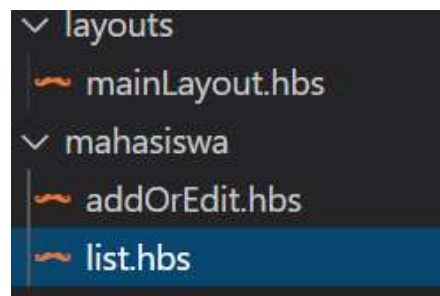
app.use('/mahasiswa', mahasiswaController);
```

Mahasiswa.model.js

Pada bagian Controller terdapat controller mahasiswaController.js yang berisi fungsi post dan get dari web. Di dalamnya kami buat function insertRecord, function updateRecord, function handleValidationError.

Pada bagian view, disini karena kami mengadopsi handle bars, pada view dibagi menjadi 2 bagian yakni layout dan v\_mahasiswa. Nantinya layout akan menjadi parent dari view, di dalamnya terdapat body yang akan mengambil view yang sesuai pada bagian v\_mahasiswa. Di dalam v\_mahasiswa terdapat

addOrEdit.hbs yang berguna dalam menampilkan tampilan tambah dari data dan juga edit dari data. Bedanya, jika edit maka akan terdapat data yang telah dipilih melalui get id nya. Jika tambah baru, maka kosongan sehingga masuk ke bagian insert record



Gambar 1. 33 Bagian view

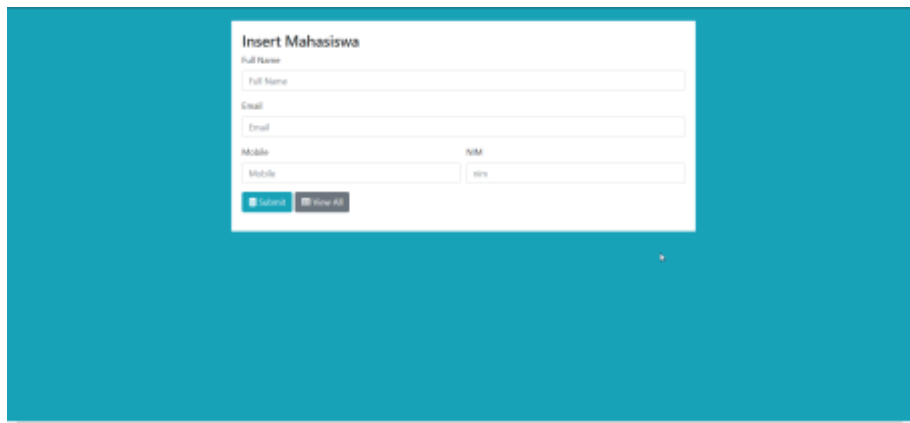
### 3. Tampilan WEB

Untuk membukanya, kita diharuskan melakukan “nodemon server.js” pada terminal. Kemudian bisa masuk ke browser dan mengetikan alamat “localhost:3000/mahasiswa”

```
PS D:\Matkul\Semester 6\PBD\NodeJs\project> nodemon server.js
[nodemon] 2.0.3
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node server.js`
(node:24980) DeprecationWarning: current Server Discovery and Monitoring engine, pass option { useUnifiedTopology }
Express server started at port : 3000
MongoDB Connection Succeeded
█
```

Gambar 1. 34 mengaktifkan port

#### 3.1 Tampilan awal / add mahasiswa



The screenshot shows a web application with a teal background. A white form titled "Insert Mahasiswa" is centered. The form contains the following fields: "Full Name" (text input), "Email" (text input), "Mobile" (text input), and "NIM" (text input). Below the "Mobile" field, there is a "Clear All" button. At the bottom of the form, there are two buttons: "Submit" (blue) and "Clear All" (grey).

Gambar 1. 35 Tampilan Awal/Add mahasiswa

Pada tampilan awal ini dapat terdapat bagian langsung menuju tambah data. Di dalamnya juga terdapat button menuju list dari mahasiswa yang telah didaftarkan

### 3.2 Tampilan daftar mahasiswa

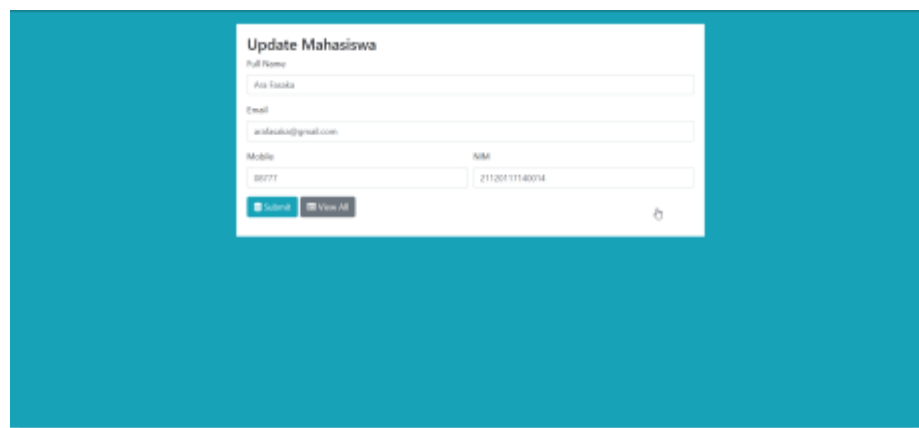


Full Name	Email	Mobile	NIM	
Doni	doni@gmail.com	0811111111	211101120001	 
ngamhar	ngamhar@gmail.com	086276077	211101120010	 
isa	isa@gmail.com	1233	123	 
Doni	doni@gmail.com	08777	2111011130003	 

Gambar 1. 36 Tampilan Awal/Add mahasiswa

Pada tampilan list mahasiswa terdapat list dari data yang telah dimasukkan. Pada tiap data juga kita bisa melakukan edit maupun delete.

### 3.3 Tampilan Edit mahasiswa



Update Mahasiswa	
Full Name	
<input type="text" value="Ara Faiska"/>	
Email	
<input type="text" value="arabauko@gmail.com"/>	
Mobile	NIM
<input type="text" value="08777"/>	<input type="text" value="21120111140074"/>
<input type="button" value="Submit"/> <input type="button" value="View All"/>	

Gambar 1. 37 Tampilan Edit Mahasiswa

Pada tampilan edit mahasiswa terdapat kolom nama, email, no hp, dan nim yang diambil dari database lalu untuk mengedit isi kolom sesuai yang diinginkan dan klik submit. Data berhasil dirubah.