

[Home](#)[Side Project / Books](#)[About](#)

Agung Setiawan

Be a Great Software Engineer

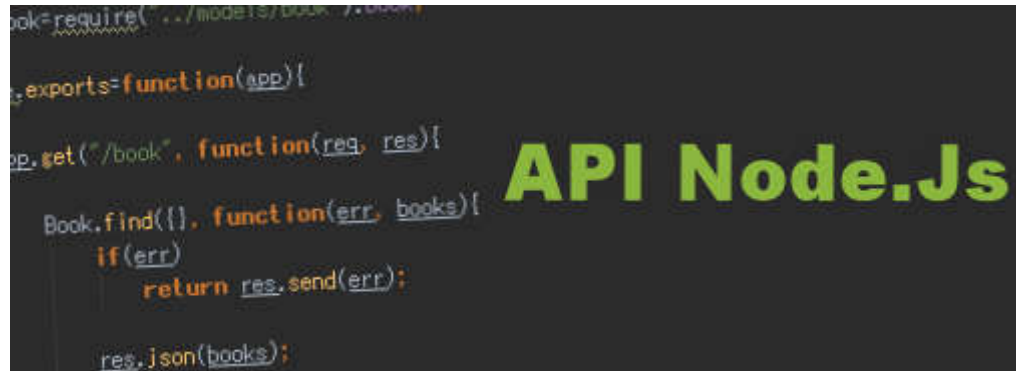


March 20, 2015 by Agung Setiawan | [Java Script](#), [Nodejs](#) in [api node.js](#), [crud node.js](#), [membuat api menggunakan node.js](#), [mongodb crud](#), [mongoose crud](#), [mongoose tutorial](#), [node.js api](#), [tutorial crud node.js](#), [tutorial node.js](#)

Belajar Membuat API Menggunakan Node.js, Express and MongoDB (Bagian 1)

Walaupun kelihatannya sepi-sepi saja penggunaan teknologi Node.js di Indonesia (atau saya yang kurang gaul), nyatanya teknologi ini sudah ramai digunakan di dunia barat sana. Nampaknya hal ini mirip dengan kisah Python dan Ruby (dan

juga Ruby on Rails) yang sejauh saya sudah cukup lama ramai di luar namun di sini masih relatif sepi dibandingkan dengan Java dan PHP.



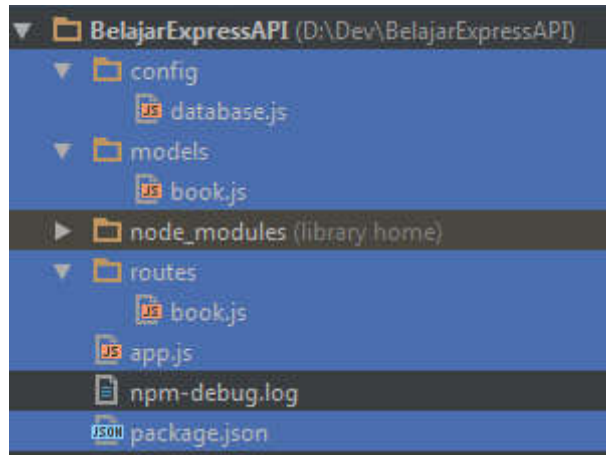
```
book=require('../models/book');  
exports=function(app){  
  app.get('/book',function(req,res){  
    Book.find({},function(err,books){  
      if(err)  
        return res.send(err);  
      res.json(books);  
    });  
  });  
}
```

API Node.js

Saya cukup terpukau ketika mengetahui salah satu perusahaan besar bidang software development di Indonesia mulai mencari software engineer yang menguasai Python dan Rails. Dalam beberapa waktu kedepan bukan tidak mungkin Node.js akan mengalami kisah yang sama.

Cukup pengantarnya sekarang mari kita belajar Node.js supaya siap saat teknologi ini mulai ramai di Indonesia :D.

Aplikasi prakarya yang akan kita buat berupa API untuk resource book. Struktur folder dan file akhir dari prakarya ini bisa dilihat pada gambar di bawah



Persiapan MongoDB

Pada terminal MongoDB, pada database apa saja yang pembaca inginkan buat collection baru dengan nama books dan masukkan beberapa sampel data yang memiliki field :

- title
- author
- page

contoh :

```
db.books.insert({"title" : "Bumi Manusia", "author" : "Pramoedya Ananta Toer", "page" :  
db.books.insert({"title" : "Bukan Pasar Malam", "author" : "Pramoedya Ananta Toer", "pag  
db.books.insert({"title" : "Mereka yang Dilumpuhkan", "author" : "Pramoedya Ananta Toer"
```

Membuat package.json

Sebelum memulai koding kita harus membuat file deklarasi project. Buat file ini di root folder dan isinya adalah

```
{  
  "name" : "Belajar-Express-API",  
  "dependencies" : {  
    "express" : "4.12.3",  
    "mongoose" : "3.8.25"  
  }  
}
```

Dependencies adalah library yang kita perlukan dalam aplikasi prakarya kita.

Setelah beres jangan lupa untuk menjalankan perintah install melalui terminal/cmd

```
npm install
```

Membuat Mongoose Model

Mongoose digunakan untuk mapping document MongoDB kedalam kode JavaScript. Pada folder project buat folder baru dengan nama models dan buat

file baru didalamnya dengan nama book.js.

Ketikkan (benar-benar diketik ya jangan dikopas) kode berikut didalam book.js

```
var mongoose=require("mongoose");
var Schema=mongoose.Schema;

var bookSchema=new Schema({
  title:String,
  author:String,
  page:Number
});

var Book=mongoose.model("Book", bookSchema);
module.exports.Book=Book;
```

Membuat Route

Nah, route ini digunakan untuk mengarahkan request oleh pengguna ke response yang ditulis oleh developer. Pada folder project buat folder baru dengan nama routes dan buat file baru didalamnya dengan nama book.js.

Ketikkan (Sekali lagi, benar-benar diketik ya jangan dikopas!) kode berikut didalam book.js

```
var Book=require("../models/book").Book;

module.exports=function(app){
```

```
app.get("/book", function(req, res){
  Book.find({}, function(err, books){
    if(err)
      return res.send(err);

    res.json(books);
  });
});

app.get("/book/:id", function (req, res) {
  Book.find({"_id":req.params.id}, function(err, book){
    if(err)
      return res.send(err);

    res.json(book);
  })
});
}
```

Dari kode di atas bisa dilihat ada 2 route yang bisa ditangani oleh aplikasi.

Pertama, saat pengguna mengarahkan ke domain/book maka response yang dikirim adalah berupa semua data book pada database dalam bentuk JSON.

Kedua, saat pengguna mengarahkan ke domain/book/someId maka response yang dikirim adalah berupa sebuah data book pada database yang memiliki field `_id` sama dengan parameter someId dalam bentuk JSON.

Kalau masih bingung kok `Book.find()` bisa mengambil data dari database, dan ada method apa saja selain `find()` silahkan buka [Koneksi Node.js MongoDB Menggunakan Mongoose](#).

Membuat Config Database

Sebentar lagi proses membuat aplikasi prakarya jilid 1 kita akan segera selesai, sebelum sampai ke tahap ada ada 1 tahap yang jangan terlewatkan.

Pada folder project buat folder baru dengan nama config dan buat file baru didalamnya dengan nama `database.js`. Berikut adalah isinya

```
module.exports = {  
  "url": "mongodb://localhost/mean"  
}
```

Secara gamblang terlihat bahwa file ini digunakan untuk menyimpan koneksi string ke MongoDB.

Membuat Main App

Akhirnya sampailah kita tahap akhir dari proses membuat aplikasi prakarya jilid 1 :D.

File ini digunakan sebagai semacam starter aplikasi. Pada file ini dipanggil file-file yang sudah sama-sama kita buat pada langkah-langkah di atas.

```
var express=require("express");
var mongoose=require("mongoose");
var dbConfig=require("./config/database");
var app=express();

mongoose.connect(dbConfig.url);

require("./routes/book")(app);

app.listen(1234);
console.log("listen on port 1234");
```

Testing App

Jalankan aplikasi dengan mengetikkan perintah berikut pada terminal/cmd

```
node app.js
```

Coba akses route yang telah dibuat tadi di browser atau pada POSTMAN, saya lebih merekomendasikan menggunakan POSTMAN.

Route `localhost:1234/book`

```
[
  {
    "_id": "550a3d6896feafd94d628b8f",
    "title": "Bumi Manusia",
    "author": "Pramoedya Ananta Toer",
    "page": 500
  },
  {
    "_id": "550a3d7996feafd94d628b90",
    "title": "Bukan Pasar Malam",
    "author": "Pramoedya Ananta Toer",
    "page": 150
  },
  {
    "_id": "550a3d9196feafd94d628b91",
    "title": "Mereka yang Dilumpuhkan",
    "author": "Pramoedya Ananta Toer",
    "page": 450
  },
  {
    "_id": "550a3da896feafd94d628b92",
    "title": "Harimau Harimau",
    "author": "Mochtar Lubis",
    "page": 250
  },
  {
    "_id": "550a3dc596feafd94d628b93",
    "title": "Para Priyayi",
    "author": "Umar Kayam",
    "page": 400
  }
]
```

Route `localhost:1234/book/550a3d6896feafd94d628b8f`

```
[
  {
    "_id": "550a3d6896feafd94d628b8f",
    "title": "Bumi Manusia",
    "author": "Pramoedya Ananta Toer",
    "page": 500
  }
]
```

Penutup

Aplikasi prakarya kita baru mampu digunakan untuk memperoleh semua data buku dan data buku dengan id tertentu. Pada jilid 2 nanti kita akan sama-sama menambahkan kemampuan untuk insert, update dan delete.

Semoga bermanfaat 😊

Facebook Comments

5 Comments

Sort by **Newest** ▼



Add a comment...



Ian Jr. · Jakarta, Indonesia

maaf pak saya mau bertanya,
pada bagian:

```
app.get("/book/:id", function (req, res) {
```

```
app.get('/book/:id', function(req, res) {
```

```
  Book.find({"_id":req.params.id}, function(err, books)
  })
```

```
});
```

jika saya mau menggunakan dua parameter dalam melakukan pemanggilan data apakah bisa dilakukan seperti ini?... [See More](#)

[Like](#) · [Reply](#) · Aug 11, 2016 1:54pm



Elizabeth Pardede · Jakarta, Indonesia

pas aq jalanin di terminal tidak error
listen on port 1234

tetapi ketika di browser, tidak error tp hanya array kosong, apakah penamaan collectionnya ??
di mongo "books" tp pas penggunaanya "book". mohon pencerahan

[Like](#) · [Reply](#) · Jul 5, 2016 2:46pm



Alif Benden Arnado · STMIK AKAKOM YOGJAKARTA

kok error gini ya gan

```
{ [Error: Cannot find module './build/Release/bson'] code: 'MODULE_NOT_FOUND' }
```

```
js-bson: Failed to load c++ bson extension, using pure JS version
```

listen on port 1234

[Like](#) · [Reply](#) · May 13, 2015 11:54pm



Adoel Mubarak · BO at Quadrant

itu penempatan lokasi file-nya gan, belum tepat, arti-nya si 'app.js' kita ga nemu di lokasi tersebut, coba penulisan require(../fileKamu.js); ----- lebih cermat lagi, antara letak file, dan penulisan ./ (titik & slash-nya)

[Like](#) · [Reply](#) · Mar 29, 2016 10:32am



Ibrahim Baim · UNY

kk tanya donk...pas jalanin..node app.js ko error ya..makasi

[Like](#) · [Reply](#) · Apr 18, 2015 12:29am



Adoel Mubarak · BO at Quadrant

Lebih rajin lagi gan oprek- oprek, more cretive lagi, pecahin masalah kamu sendiri. trial error coba samapai 10 kali

[Like](#) · [Reply](#) · Mar 29, 2016 10:32am



Si Baihaki · Backend at Pt 8villages

mantap bos.

lanjutin tutornya. node js masih jarang yang buat tutor dalam bahasa indonesia

[Like](#) · [Reply](#) · Mar 30, 2015 5:38pm



Facebook Comments Plugin



Agung Setiawan

Agung Setiawan adalah software engineer di BukaLapak.com, penulis sekaligus pecinta sastra, dan pembaca buku



Worth to Read

[Membuat Custom Domain Untuk Github Page →](#)

[← Belajar Membuat API Menggunakan Node.js, Express and MongoDB \(Bagian 2\)](#)

[Semua tulisan](#)

2 thoughts on “Belajar Membuat API Menggunakan Node.js, Express and MongoDB (Bagian 1)”



jokow says:

[May 25, 2015 at 4:15 am](#)

Terima kasih info nya. Sangat bermanfaat untuk belajar



jan sebastian says:

[August 9, 2016 at 10:19 am](#)

maaf pak saya mau bertanya,
pada bagian:

```
app.get("/book/:id", function (req, res) {
```

```
Book.find({"_id":req.params.id}, function(err, books)
{
  res.render('show', {books: books});
});
```

jika saya mau menggunakan dua parameter dalam melakukan pemanggilan data apakah bisa dilakukan seperti ini?

```
app.get("/book/:id&&:pass", function (req, res) {
```

```
  Book.find({$and:[{"_id":req.params.id},
    {"pass":req.params.pass}]},function(err,books)
  {
    res.render('show', {books: books});
  });
});
```

terima kasih



Leave a Reply

Your email address will not be published. Required fields are marked *

Comment

Name *

Email *

Website



About

Agung-Setiawan.com, adalah blog pribadi milik [Agung Setiawan](#) yang berisi opini, tutorial dan pengalaman seputar dunia software engineering.

Quick Links

[Nisfari](#)

[idRails](#)

Contact

com.agungsetiawan@gmail.com

@agungsetiawanmu

Salutation

[Casper WP](#) by Lacy Morrow

Hirtot.com	
Vis. today	145
Visits	102 157
Pag. today	252
Pages	163 830

