

# **NPM, ExpressJs dan Serve Static**

**Sio Jurnal Pipin, S.Kom.**



# NPM (Node Package Manager)

- NPM menyediakan dua fungsionalitas utama: **repositori online untuk Paket / Modul Node.js** yang dapat dicari di **NodeJS.org**. dan, **Utilitas perintah** untuk Instal Paket Node.js, manajemen Versi dan Manajemen *dependency* module Node.js (lihat **package.json**)
- **NPM** digunakan untuk:
  1. **Membuat Project Baru;**
  2. **Menginstal modul atau library;**
  3. **Menjalankan skrip command line;**

Lihat: <https://docs.npmjs.com/about-npm>

# Membuat Proyek dengan NPM

```
SIO@DESKTOP-U2DMK91 MINGW64 ~/OneDrive - mikroskil.ac.id/Mikroskil/Mata Kuliah/6. Pemrograman Web Lanjutan/Webserver/M03-npm (main)
$ npm init
This utility will walk you through creating a package.json file.
It only covers the most common items, and tries to guess sensible defaults.

See `npm help init` for definitive documentation on these fields
and exactly what they do.

Use `npm install <pkg>` afterwards to install a package and
save it as a dependency in the package.json file.

Press ^C at any time to quit.
package name: (m03-npm)
version: (1.0.0)
description: Minggu3
entry point: (index.js)
test command:
git repository:
keywords:
author: Sio
license: (ISC)
About to write to C:\Users\SIO\OneDrive - mikroskil.ac.id\Mikroskil\Mata Kuliah\6. Pemrograman Web Lanjutan\Webserver\M03-npm\package.json:

{
  "name": "m03-npm",
  "version": "1.0.0",
  "description": "Minggu3",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "Sio",
  "license": "ISC"
}

Is this OK? (yes) █
```

Catatan:

- Pastikan telah menginstall Node.JS
- Buat folder baru: **M03-NPM**
- Pada folder tersebut, buka terminal dan ketikan perintah: **npm init**
- **Enter** untuk Ok.

# package.json pada NPM

- Untuk menggunakan package NPM, proyek harus memiliki file **package.json** yang terdapat metadata tertentu pada proyek.
- **Metadata** tersebut menunjukkan beberapa aspek dari proyek dengan urutan sebagai berikut:
  - Nama proyek
  - Versi
  - Deskripsi
  - Entry point
  - Test command
  - Repositori git
  - Keyword
  - Lisensi
  - Dependensi
  - devDependency
- Metadata dapat mempermudah identifikasi proyek dan berlaku juga sebagai dasar untuk user dalam mendapatkan informasi proyek.




# Package.json pada NPM

Informasi project berisi **nama**, **versi**, dan **deskripsi**. Lalu di bagian script, berisi script bash atau command line untuk dieksekusi dengan NPM.

Selain properti-properti di atas, masih ada lagi properti lain:

- **dependencies** berisi keterangan modul atau library yang dibutuhkan aplikasi;
- **devDependencies** berisi keterangan modul atau library yang dibutuhkan untuk pengembangan aplikasi.



```
1 {
2   "name": "m03-npm",
3   "version": "1.0.0",
4   "description": "Minggu3",
5   "main": "index.js",
6   "scripts": {
7     "test": "echo \\\"Error: no test specified\\\" && exit 1"
8   },
9   "author": "Sio",
10  "license": "ISC"
11 }
```

# Install Module dengan NPM

NPM memungkinkan user untuk menginstall modul atau library dengan perintah:

**npm install <nama modul>**

Cari nama package di: <https://www.npmjs.com/>

The screenshot shows the npmjs.com website for the 'express' package. At the top, there's a navigation bar with 'Products', 'Pricing', and 'Docun'. Below that, a search bar contains 'express' and a 'Search' button. A banner below the search bar says 'Wondering what's next for npm? [Check out our public roadmap!](#)'. The main content area features the 'express' package name with a 'DT' tag, version '4.17.1', and 'Public' status. It also shows 'Published 2 years ago'. Below this, there are links for 'Readme', 'Explore', '30 Dependencies', '51.785 Dependents', and '264 Versions'. The 'express' logo is prominently displayed. A description reads 'Fast, unopinionated, minimalist web framework for node.' Below the description, there are badges for 'npm v4.17.1', 'downloads 74M/month', 'linux passing', 'windows passing', and 'coverage 100%'. A code block shows a snippet of JavaScript code for setting up an Express app. On the right side, there's an 'Install' section with the command '> npm i express'. Below that, a 'Weekly Downloads' chart shows a value of '16.806.950'. A table lists package details: Version (4.17.1), License (MIT), Unpacked Size (208 kB), Total Files (16), Issues (107), and Pull Requests (59).

npmjs.com/package/express

No Punches Made

Products Pricing Docun

npm

express

Search

Wondering what's next for npm? [Check out our public roadmap!](#)

express DT

4.17.1 • Public • Published 2 years ago

Readme Explore BETA 30 Dependencies 51.785 Dependents 264 Versions

express

Fast, unopinionated, minimalist web framework for **node**.

npm v4.17.1 downloads 74M/month linux passing windows passing coverage 100%

```
const express = require('express')
const app = express()

app.get('/', function (req, res) {
  res.send('Hello World')
})

app.listen(3000)
```

Install

> npm i express

Weekly Downloads

16.806.950

|               |               |
|---------------|---------------|
| Version       | License       |
| 4.17.1        | MIT           |
| Unpacked Size | Total Files   |
| 208 kB        | 16            |
| Issues        | Pull Requests |
| 107           | 59            |

# Install Module dengan NPM

NPM memungkinkan user untuk menginstall modul atau library dengan perintah:

**npm install <nama modul>**

Cari nama package di: <https://www.npmjs.com/>

The screenshot shows the npmjs.com website for the 'express' package. At the top, there's a navigation bar with 'Products', 'Pricing', and 'Docun'. Below that, a search bar contains 'express' and a 'Search' button. A banner below the search bar says 'Wondering what's next for npm? [Check out our public roadmap!](#)'. The main content area features the 'express' package name with a 'DT' tag, version '4.17.1', and 'Public' status. It also shows 'Published 2 years ago'. Below this, there are links for 'Readme', 'Explore', '30 Dependencies', '51.785 Dependents', and '264 Versions'. The 'express' logo is prominently displayed. A description reads 'Fast, unopinionated, minimalist web framework for node.' Below the description, there are badges for 'npm v4.17.1', 'downloads 74M/month', 'linux passing', 'windows passing', and 'coverage 100%'. A code block shows a snippet of JavaScript code for setting up an Express app. On the right side, there's an 'Install' section with the command '> npm i express'. Below that, a 'Weekly Downloads' chart shows a value of '16.806.950'. A table lists package details: Version (4.17.1), License (MIT), Unpacked Size (208 kB), Total Files (16), Issues (107), and Pull Requests (59).

npmjs.com/package/express

No Punches Made

Products Pricing Docun

npm

express

Search

Wondering what's next for npm? [Check out our public roadmap!](#)

express DT

4.17.1 • Public • Published 2 years ago

Readme Explore BETA 30 Dependencies 51.785 Dependents 264 Versions

express

Fast, unopinionated, minimalist web framework for **node**.

npm v4.17.1 downloads 74M/month linux passing windows passing coverage 100%

```
const express = require('express')
const app = express()

app.get('/', function (req, res) {
  res.send('Hello World')
})

app.listen(3000)
```

Install

> npm i express

Weekly Downloads

16.806.950

|               |               |
|---------------|---------------|
| Version       | License       |
| 4.17.1        | MIT           |
| Unpacked Size | Total Files   |
| 208 kB        | 16            |
| Issues        | Pull Requests |
| 107           | 59            |

# Install ExpressJs dengan NPM | Cont.

Untuk menginstall package ExpressJs dapat dilakukan dengan melihat instruksi perintah pada situs [npmjs.com](https://expressjs.com/en/starter/installing.html)

## Install

```
> npm i express
```

Weekly Downloads

16.806.950

Pada folder M03-npm, install express dengan perintah:

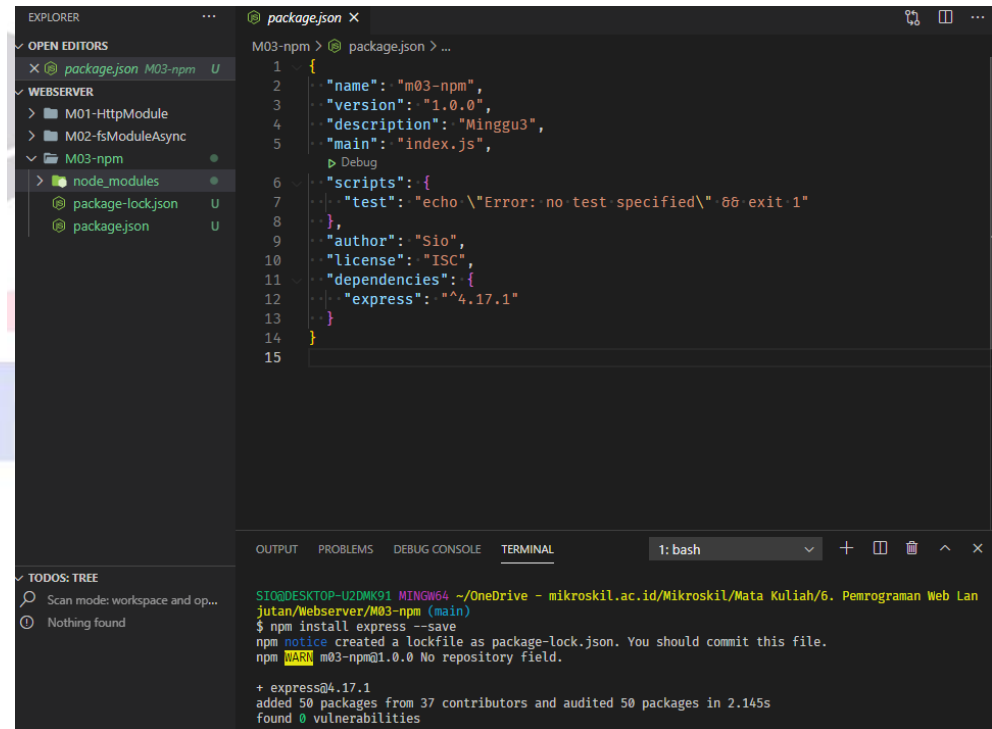
***npm install express --save***

Catatan: **--save**, berfungsi untuk menyimpan express dalam **dependencies list**.

Setelah install, pada package.json akan bertambah express beserta versinya. Juga folder **node\_module**.

Baca:

<https://expressjs.com/en/starter/installing.html>



```
package.json
1 {
2   "name": "m03-npm",
3   "version": "1.0.0",
4   "description": "Minggu3",
5   "main": "index.js",
6   "scripts": {
7     "test": "echo \"Error: no test specified\" && exit 1"
8   },
9   "author": "Sio",
10  "license": "ISC",
11  "dependencies": {
12    "express": "^4.17.1"
13  }
14 }
15
```

```
1: bash
$ npm install express --save
npm notice created a lockfile as package-lock.json. You should commit this file.
npm WARN m03-npm@1.0.0 No repository field.

+ express@4.17.1
added 50 packages from 37 contributors and audited 50 packages in 2.145s
found 0 vulnerabilities
```



# Cari tahu

1. **Bagaimana cara module / package untuk Dev dan Global**
2. **Bagaimana cara menghapus Package yang telah diinstall sebelumnya?**
3. **Bagaimana cara tambah scripts di package.json**
4. **Bagaimana cara run / mengeksekusi server http melalui NPM?**



# ExpressJs

- **ExpressJS** merupakan **framework** minimal yang sangat fleksibel. Dapat digunakan untuk **membuat web server** HTML, **server file statik**, aplikasi chat, search engine, sosial media, layanan web dengan akses melalui REST API atau aplikasi hybrid yaitu selain pengguna mempunyai akses melalui REST API juga mempunyai akses ke HTML page.
- Untuk memudahkan pembuatan aplikasi web anda bisa menggunakan framework ExpressJS daripada harus menggunakan module http bawaan Node.js.
- Framework ini menawarkan beberapa **fitur seperti routing, rendering view** dan **mendukung middleware** dengan kata lain anda akan banyak menghemat waktu dalam pengembangan aplikasi Node.Js.

Source: <https://expressjs.com/en/starter/installing.html>



# Server File - ExpressJs

Struktur file project ExpressJs adalah sebagai berikut:

- **node\_modules**, di dalamnya terdapat berbagai folder library - library Node.js yang dipasang melalui node package manager atau NPM
- **publik**, kita dapat menaruh berbagai file HTML, CSS, Javascript, atau gambar di dalam folder ini.
- **routes**, di dalamnya terdapat berbagai file yang berisi action yang diterima oleh routing yang kita definisikan. Dapat menerima request berupa GET, POST, PUT, DELETE, OPTION, dan HEAD
- **views**, di dalamnya terdapat berbagai file jade atau html yang digunakan oleh routes untuk menampilkan halaman
- **app.js**, file utama Express.js yang berisi penggunaan package utama dan konfigurasi utama
- **npm-debug.log**, file yang berisi hasil debug atau error yang dicatat oleh Express.js selama running
- **package.json**, berisi struktur JSON yang mendefinisikan profil proyek dan dependensi paket apa saja yang dibutuhkan aplikasi yang kita kembangkan. NPM akan melacak dependensi dengan melihat file ini

```
server-file-statis
├─ package.json
├─ app.js
├─ node_modules
└─ publik
    └─ index.html
```

# Server static dengan ExpressJs

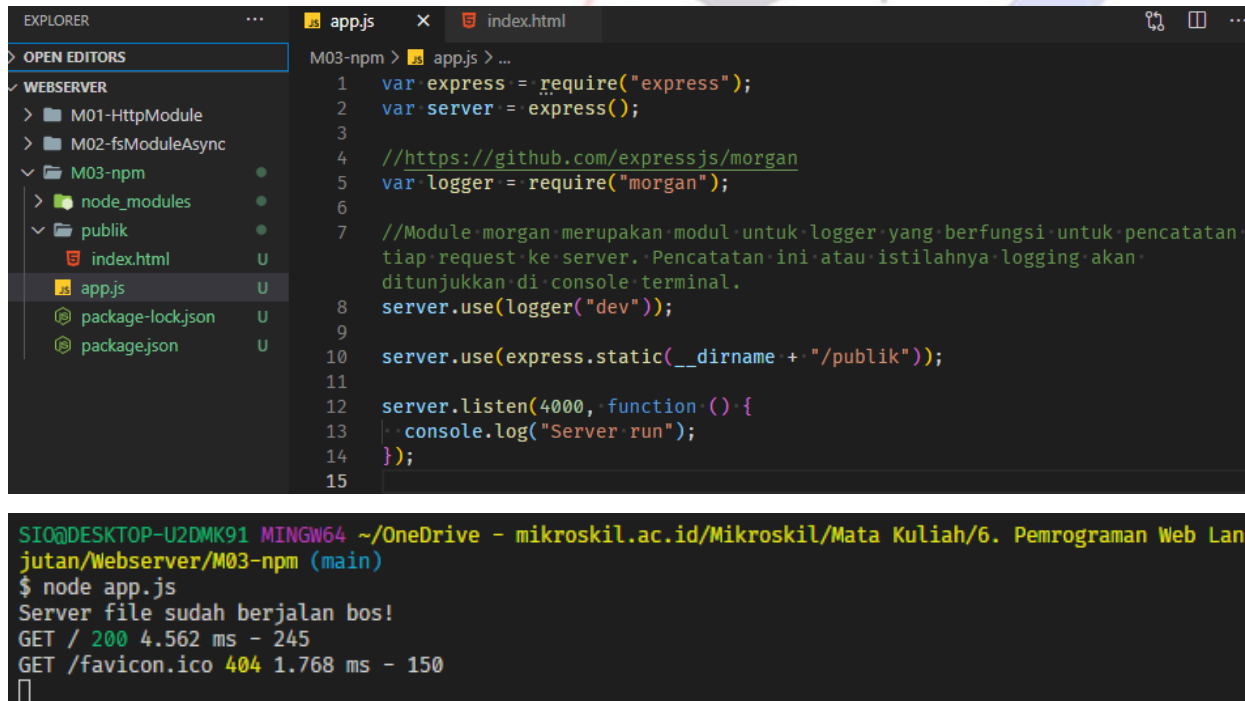
Server static ini akan menampilkan halaman index dari folder publik yaitu index.html

Pada server di bawah, digunakan **Module morgan** merupakan modul untuk logger yang berfungsi untuk **pencatatan tiap request ke server**.

Pencatatan ini atau istilahnya logging akan ditunjukkan di console terminal.

Install dengan: ***npm install morgan --save***

Run server: ***node app.js***



The screenshot shows a VS Code editor with a file explorer on the left and a code editor in the center. The file explorer shows a project structure with folders 'M01-HttpModule', 'M02-fsModuleAsync', and 'M03-npm'. The 'M03-npm' folder is expanded, showing 'node\_modules', 'publik', 'index.html', 'app.js', 'package-lock.json', and 'package.json'. The 'app.js' file is selected. The code editor shows the following code:

```
1 var express = require("express");
2 var server = express();
3
4 //https://github.com/expressjs/morgan
5 var logger = require("morgan");
6
7 //Module morgan merupakan modul untuk logger yang berfungsi untuk pencatatan
8 //tiap request ke server. Pencatatan ini atau istilahnya logging akan
9 //ditunjukkan di console terminal.
10 server.use(logger("dev"));
11
12 server.use(express.static(__dirname + "/publik"));
13
14 server.listen(4000, function () {
15   console.log("Server run");
16 });
```

Below the code editor, a terminal window shows the output of running the server:

```
SIO@DESKTOP-U2DMK91 MINGW64 ~/OneDrive - mikroskil.ac.id/Mikroskil/Mata Kuliah/6. Pemrograman Web Lan
jutan/Webserver/M03-npm (main)
$ node app.js
Server file sudah berjalan bos!
GET / 200 4.562 ms - 245
GET /favicon.ico 404 1.768 ms - 150
```

Dengan menggunakan **Morgan** setiap aksi oleh user akan dicatat terminal.

# Cari tahu

1. Bagaimana cara menjalankan server dengan npm? Contoh: `npm run dev`
2. Server REST dengan express js

