**Top NodeJS packages to speed up application development -** [Benjamin Iduwe](https://bencoderus.medium.com/?source=post_page-----d4fa49acca5a--------------------------------) Sep 25, 2021

NodeJS is an open-source server-side runtime environment built on Chrome’s V8 JavaScript engine. It provides an event-driven, non-blocking (asynchronous) I/O and cross-platform runtime environment for building highly scalable server-side applications using JavaScript. One of the advantages of using NodeJS is the rich collection of packages it provides, I will be sharing some useful NodeJS packages and they would be categorized by their functionality.

**NodeJS Frameworks.**

1. **Express** is a minimal and flexible Node.js web application framework that provides a robust set of features. With a myriad of HTTP utility methods and middleware at your disposal, creating a robust API is quick and easy. Express also provides a thin layer of fundamental web application features, without obscuring Node.js features that you know and love <https://expressjs.com/>
2. **Fastify** is a web framework highly focused on providing the best developer experience with the least overhead and a powerful plugin architecture. It is inspired by Hapi and Express and as far as we know, it is one of the fastest web frameworks in town. Features: Highly performant, Logging, Routing, Middleware, Serverless support. <https://www.fastify.io/>
3. **Hapi.js** (derived from Http-API) is an open-source Node.js framework used to build powerful and scalable web applications. Hapi is commonly used to build Application Programming Interface servers, HTTP-proxy applications, and websites. Hapi.js was created by the mobile team at Walmart Labs to handle their traffic for events like Black Friday, which is by far one of the busiest days for online shopping on the U.S. calendar. Hapi was originally built using the express framework before facing challenges that drove Walmart to make hapi, its own stand-alone framework. <https://hapi.dev/>
4. **Koa** is a new web framework designed by the team behind Express, which aims to be a smaller, more expressive, and more robust foundation for web applications and APIs. By leveraging async functions, Koa allows you to ditch callbacks and greatly increase error handling. Koa does not bundle any middleware within its core, and it provides an elegant suite of methods that make writing servers fast and enjoyable. <https://koajs.com/>
5. **Sails** is the most popular MVC framework for Node.js, designed to emulate the familiar MVC pattern of frameworks like Ruby on Rails, but with support for the requirements of modern apps: data-driven APIs with scalable, and service-oriented architecture. Sails makes it easy to build custom, enterprise-grade Node.js apps. <https://sailsjs.com/>
6. **AdonisJS** includes everything you need to create a fully functional web app or an API server. Stop wasting hours in downloading and assembling hundreds of packages together — Use AdonisJS and be productive from day one. For PHP developers that love Laravel, you would also love AdonisJS. <https://adonisjs.com/>
7. **Nest (NestJS**) is a framework for building efficient, scalable Node.js server-side applications. It uses progressive JavaScript, is built with and fully supports TypeScript (yet still enables developers to code in pure JavaScript), and combines elements of OOP (Object Oriented Programming), FP (Functional Programming), and FRP (Functional Reactive Programming). <https://nestjs.com/>

**Testing frameworks.**

1. **Mocha** is a feature-rich JavaScript test framework running on Node.js and in the browser, making asynchronous testing simple and fun. Mocha tests run serially, allowing for flexible and accurate reporting while mapping uncaught exceptions to the correct test cases. <https://mochajs.org/>
2. **Jest** is a delightful JavaScript Testing Framework with a focus on simplicity. It works with projects using: [Babel](https://babeljs.io/), [TypeScript](https://www.typescriptlang.org/), [Node](https://nodejs.org/), [React](https://reactjs.org/), [Angular](https://angular.io/), [Vue](https://vuejs.org/), and more!. Jest comes with a lot of testing features out of the box and it also has full support for Typescript using TS Jest. <https://jestjs.io/>
3. **Chai** is a BDD / TDD assertion library for node and the browser that can be delightfully paired with any javascript testing framework. Chai works hand-in-hand with Mocha to provide full support for API testing. <https://www.chaijs.com/>
4. **Puppeteer** is a Node library that provides a high-level API to control Chrome or Chromium over the DevTools Protocol. Puppeteer runs headless by default but can be configured to run full (non-headless) Chrome or Chromium. Puppeteer lets you automate testing your frontend using the headless chromium, it also comes with tons of features worth exploring. <https://www.npmjs.com/package/puppeteer>
5. **Sinon JS** is a popular JavaScript library that lets you replace complicated parts of your code that are hard to test for “placeholders,” so you can keep your unit tests fast and deterministic, as they should be. <https://sinonjs.org/>

**Security.**

1. **Helmet** helps you secure your Express apps by setting various HTTP headers. It’s not a silver bullet, but it can help. The top-level helmet function is a wrapper around 15 smaller middlewares, 11 of which are enabled by default. <https://www.npmjs.com/package/helmet>
2. **Xss clean** provides a connect middleware to sanitize user input coming from POST body, GET queries, and URL params. Works with Express, Restify, or any other Connect app. <https://www.npmjs.com/package/xss-clean>
3. **Sanitize-html** is well suited for cleaning up HTML fragments such as those created by CKEditor and other rich text editors. It is especially handy for removing unwanted CSS when copying and pasting from Word. It also allows you to specify the tags you want to permit, and the permitted attributes for each of those tags. It helps you strip tags that could make your site vulnerable to XSS attacks. https://www.npmjs.com/package/sanitize-html

**Database.**

1. **Mongoose** ODM provides a straightforward, schema-based solution to model your application data. It includes built-in type casting, validation, query building, business logic hooks and more, out of the box. Everything you need to build an app using MongoDB is readily available for you on Mongoose. <https://mongoosejs.com/>
2. **Typegoose** is the Typescript version of mongoose, it provides a more robust approach to use MongoDB and Typescript. <https://typegoose.github.io/typegoose/>
3. **Sequelize** provides a straightforward and reliable ORM to interact with SQL databases like MySQL, PostgreSQL, SQLite, Maria DB, and MS SQL. <https://sequelize.org/>
4. **Typeorm** provides a seamless approach to interact with SQL databases using Typescript. <https://typeorm.io/>
5. **Prisma** helps app developers build faster and make fewer errors with an open-source database toolkit for PostgreSQL, MySQL, SQL Server, and SQLite. Prisma comes with tons of features to help you ship apps faster. Prisma currently supports JS and Typescript. <https://www.prisma.io/>

**Utility.**

1. [**Socket.io**](http://socket.io/) is a library that enables real-time, bidirectional, and event-based communication between the browser and the server. It scales to multiple servers and sends events to all connected clients with ease and provides a seamless approach to communicate with a web socket. <https://socket.io/>
2. **Dinero.js** makes it easy to create, calculate and format monetary values in JavaScript. You can perform arithmetic operations, extensively parse and format them, check for a number of things to make your own development process easier and safer.
3. **Joi** is an object schema description language and validator for JavaScript objects. It provides a cleaner approach for performing input validation in Nodejs. <https://joi.dev/>
4. **Morgan** is an HTTP request logger on Nodejs. It provides positions to log things like Request path, request method, response time, and response status code. <https://www.npmjs.com/package/morgan>
5. **Winston** is designed to be a simple and universal logging library with support for multiple transports. A transport is essentially a storage device for your logs. Each Winston logger can have multiple transports configured at different levels. For example, one may want error logs to be stored in a persistent remote location (like a database), but all logs output to the console or a local file. <https://www.npmjs.com/package/winston>
6. **Bcrypt** is a library that helps you hash and verify your password a plain password against a hash password. <https://www.npmjs.com/package/bcrypt>
7. **Dotenv** is a zero-dependency module that loads environment variables from a .env file into process.env. Storing configuration in the environment separate from code is based on The Twelve-Factor App methodology. <https://www.npmjs.com/package/dotenv>
8. **Jsonwebtoken**: JSON Web Token is a proposed Internet standard for creating data with optional signature and/or optional encryption whose payload holds JSON that asserts some number of claims. The tokens are signed either using a private secret or a public/private key. This package is an implementation of JWT functionalities for NodeJS. <https://www.npmjs.com/package/jsonwebtoken>
9. **i18n** is a lightweight simple translation module with dynamic JSON storage. Supports plain vanilla Node.js apps and should work with any Nodejs framework Stores language files in JSON files compatible to webtranslateit JSON format. Adds new strings on the fly when first used in your app. No extra parsing is needed. <https://www.npmjs.com/package/i18n>
10. **Bluebird is a fully-featured promise library with a focus on** innovative features and performance. It lets you convert your traditional callback functions into promises, with support for almost all the top packages. <https://www.npmjs.com/package/bluebird>
11. **Express Rate Limiter** provides a basic rate-limiting middleware for Express. Use to limit repeated requests to public APIs and/or endpoints such as password reset and login. <https://www.npmjs.com/package/express-rate-limit>
12. **Axios** provides a promise-based HTTP client for NodeJS. https[://www.npmjs.com/package/axios](https://www.npmjs.com/package/axios)
13. C**ross-env** makes it so you can have a single command without worrying about setting or using the environment variable properly for the platform. Just set it like you would if it’s running on a POSIX system, and cross-env will take care of setting it properly. <https://www.npmjs.com/package/cross-env>
14. **AWS-SDK** provides an interface to interact with AWS services using NodeJS. <https://www.npmjs.com/package/aws-sdk>

**Performance.**

1. **PM2** is a daemon process manager that will help you manage and keep your application online. Getting started with PM2 is straightforward, it is offered as a simple and intuitive CLI, installable via NPM. You would get a Load balancer, logs facility, startup script, micro-service management, cluster management at a glance. <https://pm2.keymetrics.io/>
2. **Bull** is a Node library that implements a fast and robust queue system based on Redis. Bull is very easy to use and it also comes with tons of features like named queues, FIFO or LIFO support, repetitive jobs, jobs rate-limiting, etc. <https://optimalbits.github.io/bull/>
3. **Redis** is an open-source (BSD licensed), in-memory data structure store, used as a database, cache, and message broker. Redis provides data structures such as strings, hashes, lists, sets, sorted sets with range queries, bitmaps, hyperloglogs, geospatial indexes, and streams. The node package provides robust support to interact with all the Redis features. <https://www.npmjs.com/package/redis>.
4. **Amqplib** is a NodeJS library for making AMQP 0–9–1 client for Node.JS, and an AMQP 0–9–1 client. It provides a smooth interface for interacting with RabbitMQ. <https://www.npmjs.com/package/amqplib>

**Package managers**

1. **NPM** (Node package manager) is the default package manager for NodeJS it helps you bootstrap your app and manage all your dependencies via the command line.
2. **Yarn** was built by Facebook to improve on some shortcomings of the default node package manager, Yarn also comes with offline support and It’s more network resilient than NPM.