

# **NodeJS Frameworks**

## **Hapi.Js**

It is introduced by Eran Hammer at Walmart while trying to handle traffic on black Friday. It is a powerful and robust open-source best node JS framework for developing JSON API. Application programming interface (API) servers, websites, and HTTP proxy applications are built with hapi.js. Various key features such as input validation, implement caching, configuration-based functionality, error handling, logging, and more and the well-developed plugin system and make the Hapi one of the most preferred frameworks. It is used in building useful applications and providing technology solutions by several large-scale websites such as PayPal, Disney.

### **Features:**

- Code reusability
- No external dependencies
- Security
- Integrated Architecture: comprehensive authorization and authentication API available in a node framework.

### **Advantages:**

- Hapi enables one to build a scalable web application or API.
- Hapi uses plugins which breaks applications in to smaller pieces thus helping with scaling.
- Hapi provides a deeper control over request handling through the use of context.
- Hapi provides input/output validation, routing and caching, hence making it a good choice for building REST APIs.
- Hapi provides good plugins such as Yar, Good, and Catbox, etc.

### **Disadvantages:**

- Refactoring in Hapi is manual.
- Endpoints must be created and tested manually.
- It restricts developers by using Hapi specific plugins and modules which are not compatible with other frameworks.

## **Express.js**

Built by TJ Holowaychuk, Express.js is a flexible and minimal Node.js application framework specifically designed for building single-page, multi-page, and hybrid applications that provide a robust set of features for web and mobile applications. Express has no out-of-the-box object-relational mapping engine. Express isn't built around specific components, having "no opinion" regarding what technologies you plug into it. This freedom, coupled with lightning-fast setup and the pure JavaScript environment of Node, makes Express a strong candidate for agile development and rapid prototyping. Express is most popular with startups that want to build a product as quickly as possible and don't have very much legacy code. The framework has the advantage of continuous updates and reforms of all the core features. It is a minimalist framework that is used to build several mobile applications and APIs.

### **Advantages:**

- Express has a large community.
- Express is simple, flexible, and scalable.
- Express is fully customizable.
- It is easy to integrate third party services and middleware.
- Express has a low learning curve.

### **Disadvantages:**

- Express requires a lot of manual labor since endpoints, middleware, etc. need to be created and tested manually.
- Refactoring larger code bases is very challenging.

## **Koa.js**

Developed and maintained by the creators of widely used Node.js framework — Express.js, Koa, a cross-platform server-side runtime environment application, is an object containing an array of middleware functions that are composed and executed in a stacked manner upon request making it easier for web developers to build fast and scalable network applications with JavaScript. It improves interoperability, robustness, and makes writing middleware much more enjoyable. Many web developers, at present, even use Node.js to write both the frontend and backend of a web application in JavaScript. Web developers can further accelerate the development of custom web applications and application programming interfaces (APIs) by using several Node.js frameworks.

### **Advantages:**

- Koa has a small footprint (about 600 lines of code) therefore making it very light weight.
- Koa uses try/catch which improves error handling.
- Koa has a cleaner and more readable asynchronous code.
- Koa does not use callbacks hence facilitating upstream and downstream flow of controls.
- Refactoring larger code bases in Koa is easier due to modularization.

### **Disadvantages:**

- Koa has a relatively small community.
- It is not compatible with Express middleware.
- Koa uses ES6 generators which are not compatible with middleware from other frameworks.

## Sails.js

It is a model–view–controller (MVC) framework for Node.js that follows the principle of “convention over configuration.” The Ruby on Rails web framework inspires it, thus emulates the familiar MVC pattern to build single-page apps, REST APIs, and real-time apps. Extensively uses code generators that allow building applications with less writing of code. The framework is built on top of Socket.io, a JavaScript library for adding real-time, bidirectional, event-based communication to applications and Express.js, one of the most popular Node.js libraries.

### **Advantages:**

- **Free Generation of JSON API**  
Sails.js offers free JSON API generation. You don't need extra routing and data becomes available anywhere.
- **Separates Business Logic from UI**  
Based on the MVC architecture, it splits business logic from UI, keeping them in different layers.
- **Well Supported Socket.io**  
It has got good support for the Socket.io that helps apps to get real time interaction like chat and games.
- **Convention preferred to Configuration**  
Sails.js is that framework which prefers convention over the configuration.
- **Allows you to store data anywhere**  
Sails.js uses Waterline ORM at its backend allowing you to store your data anywhere or in any database.

### **Disadvantages:**

- **Sails.js is a slow framework**  
Compared to others, Sails.js is slow with a huge memory.
- **Asset Management is Not Good**  
Asset management in Sails.js framework has been found to be a little messy and clumsy affair.
- **Project Development is Slow and Time Consuming**  
Because of its overall slowness, the project development also seems to be slow in the process and is somewhat like perpetual beta. It also consumes a lot of time to start up.

## Meteor.js

Meteor.js is used for building modern web and mobile applications and is defined as a full-stack JavaScript platform. The most important feature of Meteor.js is that it provides real-time updates so that all your changes on the web will get updated on the template instantly.

The framework has a simplified platform for the whole tier of the app and it's in the same language (JavaScript). This makes this framework work in a more efficient manner on the server side as well as the client side.

### **Features**

- It has the capability to manage larger projects
- Has rich and organized documentation community
- It leverages the Facebook GraphQL data stack
- It's easy to understand for most developers

**Advantages:**

- With Meteor.js, Your Applications are Real-Time By Default
- Saves Lot of Time With Packages
- It Has a Supportive Community
- Developed With a Single Language
- A Tailor-Made Framework for Web Developers
- Easier to Learn
- Easily Convert Web Apps into SmartPhone Apps

**Disadvantages:**

- It Lacks the Server-Side Support
- Absence of Native Widget Library
- A Website May Run Slowly
- It Lacks Inbuilt Support for PWA

**References:**

<https://hackr.io/blog/nodejs-frameworks>

<https://www.section.io/engineering-education/hapi-vs-koa-vs-express/>

<https://www.covetus.com/blog/7-meteorjs-benefits-that-make-it-an-excellent-framework-to-build-web-mobile-apps>