

Which is the Best NodeJS Framework to Use in Web Development?

 Published On: June 22, 2021  Last Updated: January 5, 2022



When it comes to frontend development, JavaScript is undoubtedly the most popular scripting language. Many frameworks and libraries for JavaScript are available to simplify the development process.

Node.js is one such JavaScript framework for server-side JavaScript programming. According to [W3Techs](#), 1.7% of websites on the web(approximately 20 million+ websites) are using **Node.js** in 2022.

Web developers prefer using Node.js owing to its scalability, speed, and performance. There are many frameworks and tools available to make working with NodeJS simple and efficient.

We list out **the best NodeJS frameworks** that you can use for seamless web development. We will also help you with criteria to consider when choosing the right NodeJS framework.

Table of Content [\[show\]](#)

Node.js – A Framework or An Environment?

This website uses cookies to ensure you get the best experience on our website [More info](#)

OK

components and many more packages for web development.

It is built on the fast and reliable **Google Chrome's V8 engine**, which can execute JavaScript code on a browser server.

Additionally, NodeJS does not allow a process to block CPU time for running input-output operations. This means multiple processes can execute at the same time. This **event-driven I/O management** keeps the framework lightweight and efficient.

Node.js is therefore perfect for working with data-intensive real-time applications that run across distributed devices.

What Are Node.js Frameworks?

A Node.js framework provides basic application architecture, template engines, and inbuilt libraries that help you develop scalable, high-performance websites and apps with Node.js. They simplify and streamline the development process of server-side web applications.

With an abstract design, a Node.js framework holds placeholder elements. You just have to write in your customizable code to make a complete web application. It takes care of the rest of the application structure. They also help developers define the control flow and data flow of the web app.

But why do you need Node.js frameworks?

Node.js comes with a lot of in-built functionalities that can be hard to focus on if you are not an expert. Node.js frameworks like **Sails** or **Express** help you streamline the development process while taking care of the background details for you. This encourages rapid development, increased productivity and most importantly, smooth testing and debugging.

They provide tools and plugins making apps easier to prototype and develop. Node.js frameworks also improve the performance and scalability of web applications.

Single-threaded or Multi-threaded?

Some sections of Node.js are written in JavaScript, while others have been coded in C++. The JS code sections of Node.js are single-threaded. But the low-level task that requires interacting with an operating system is multi-threaded.

Types of JS Frameworks

This website uses cookies to ensure you get the best experience on our website [More info](#)

OK



Types of JS Frameworks

- Full Stack
- Microframeworks
- Rest API Frameworks



1. Full Stack-

- Full-stack MVC frameworks are a **single-stop solution** for developers.
- They come bundled with a variety of features such as project scaffolding, in-built template engines, web socket support, and core UI libraries.
- You can use full-stack Node JS frameworks to build real-time scalable web apps. They provide backend and frontend management using only JavaScript.

2. Microframeworks –

- Micro frameworks are lightweight frameworks that are used for a particular development task.
- They are flexible and offer rich configuration and are less opinionated than full-stack frameworks.

3. Rest API Frameworks –

- Rest API frameworks are server-side frameworks specifically for the creation of robust APIs.
- You can develop a simple REST API for handling CRUD (Create-Read-Update-and Delete) requests on your server.

[Talk To Our Experts For Free →](#)

Popular Node.js Frameworks for Web App Development

1. Express

The logo for Express, featuring the word "Express" in a large, lowercase, sans-serif font.

Express.js is used for the development of backend services such as web sockets, web APIs, and other web services. It is free and open-source software, released under MIT license, and is closely inspired by the microframework Sinatra.

The key features of the Express.JS framework are-

- Express provides a layer on top of Node.js that provides web application development features such as basic routing, middleware, template engine and static files serving
- It is minimalist and does not overload the performative speed of NodeJS
- It is used for building efficient modern web applications and web API
- Express uses Model-View-Controller(MVC) structure for segmentation of code
- Flexible and customizable according to personal needs

This website uses cookies to ensure you get the best experience on our website [More info](#)

[OK](#)



Express.js is used by **Uber**, **PayPal**, **IBM** and other companies for server development.

Read our blog on [Express Vs Django](#), for a detailed comparison of the two backend frameworks.

2. Meteor



Meteor is a free and open-source framework for seamlessly building and deploying web, desktop and mobile applications in JavaScript. This framework is used for rapid prototyping using Node.js. Some other advantages of using Meteor are-

- You can develop a single cross-platform source code that works on the web, desktop, iOS, and android
- MongoDB relational database is integrable with Meteor.Js which helps you synchronize the data
- **LiveReload** is a built-in feature of Meteor which lets you visualize your code logic without rebuilding the project or refreshing the browser page.
- Meteor has a reactive UI development platform
- A single language consistently used all over the application thus improving consistency

This website uses cookies to ensure you get the best experience on our website [More info](#)

OK



3. Sails



Sails is an MVC architecture JavaScript framework developed on top of Node JS. It is the top choice for developing enterprise applications and web APIs in a simple and uncomplicated way.

Sails has an improved Rails version of MVC software design, incorporating more data requests, automatic generation of JSON requests and additional HTTP support.

- Sails bundles a powerful ORM, **Waterline**, which provides a simple data access layer that works with any popular database
- Sails incorporate **blueprints and scaffoldings** that you can use to immediately start your app's backend without any coding.
- Every route on your Sails app supports **web sockets** that will translate incoming socket messages.

4. Nest.js

This website uses cookies to ensure you get the best experience on our website [More info](#)

[OK](#)



One of the most progressive Node.js frameworks, Nest.js is used for building dynamic and scalable applications for enterprises and businesses. It offers complete flexibility to developers like you with an extensive selection of libraries. Nest uses progressive JavaScript and TypeScript as its core language.

- Typescript enabled, which ensures Nest.Js has strong typing and employs good coding practices
- It is a multi-paradigm framework, implementing **OOP**(Object Oriented Programming), **FP**(Functional Programming), and **FRP**(Functional Reactive Programming) concepts
- Comes with a backend productivity booster
- It has an extremely adaptable ecosystem that supports the development of fully-fledged server-side applications.
- The Nest CLI is powerful letting you generate many out-of-the-box features in your web application
- It gives a live preview feature which means you are able to see how your application looks in your personal browser.
- Very versatile, flexible and extensible with ready-to-use patterns
- Uses Express under the hood as an HTTP server framework



Koa is a highly extensible minimalist web framework for NodeJS developed by the Express.js team.

- A Koa web application looks like an object containing an array of middleware functions. Koa composes these functions in a stack and executes them when a user requests
- Improved interoperability and robustness with high-level JavaScript even in the lower middleware levels
- Offers tools for the support of content negotiation, cache freshness, proxy support, and pop-ups and redirection
- It provides an extendable platform for middlewares, without having any integrated to its core, thus maintaining a small footprint
- The most important feature is that it allows you to work without callbacks

6. Hapi



Hapi is a secure JS framework for developing JSON APIs and scalable web apps. With input validation, it is one of the most secure and minimal JS frameworks.

Some key features of the Hapi.js framework are –

- It uses a plugin-based architecture

- Has a rich ecosystem that has a lot of defaults for securing your website.
- Used by large-scale companies such as Disney, PayPal
- Hapi framework provides dynamic content for users by executing the JavaScript template engine quickly.
- It is extremely **scalable** and supports many third-party plugins. These plugins allow the building of real-time chatting apps that can be scaled to support thousands of users.
- It helps developers generate detailed documents.

7. Derby

This website uses cookies to ensure you get the best experience on our website [More info](#)

[OK](#)



Derby.js is a full-stack web framework for the development of Node.js web applications. It is well known for its ‘Racer Engine support’, which enables multiple users to work in real-time on the same data.

- Using the Operational Transformation services of **ShareDB**, this Node framework employs a sophisticated conflict resolution algorithm for offline access, real-time updates and instant interactions.
- Derby.js unifies the server and client web browsers, enabling the usage of single code for both backend and frontend.
- There is no need for implementing the costly DOM implementations on the server-side as Derby uses templates for data rendering.
- On the server-side, Derby.js renders the HTML output of templates for speedier page loading
- It offers full support for search engine optimizations.
- On the client-side web browsers, DerbyJS renders with fast, native DOM manipulation methods.

8. Loopback

This website uses cookies to ensure you get the best experience on our website [More info](#)

[OK](#)



LoopBack



A highly extensible API MVC framework for Node.js and TypeScript, LoopBack enables you to quickly create APIs and microservices composed of backend systems. Loopback establishes a seamless connection between your API and its backend data sources.

Built on top of Express, when you input a data model definition, LoopBack will generate an end-to-end REST API.

- Has an inbuilt client, API Explorer which lets you dynamically explore your application
- Uses MySQL, MongoDB, Oracle databases to create data models from schemas
- SOAP or REST services are employed for speed
- It has an easy to use CLI which enables simple and quick prototyping
- It supports native mobile and browser development for Android, iOS, and web browsers
- You can use add-on components for file management, 3rd-party login, and more.

9. Adonis.js

This website uses cookies to ensure you get the best experience on our website [More info](#)

OK



Adonis is an MVC JavaScript framework with internal support for Express, Koa and MEAN tech stack. The developers were hugely influenced by the PHP-based Laravel, and thus share many similarities with it.

- Adonis.js has a stable ecosystem with many baseline features built within the core of the framework.
- It follows the MVC architecture and focuses on getting things done with simplicity with concepts of dependency injection through IOC(Inversion of Control)
- AdonisJS has a feature-rich routing layer with support for route groups, subdomain based routing and resource resources
- Web sockets can be customized using this Node.js framework
- Also provides built-in modules for Redis, emails, data validation, ORM and authentication
- There is a diverse authentication system that provides support for sessions, API tokens and OAuth based user login
- Manage user access using the predefined authorization actions and policies
- An embedded CLI tool to help you create custom commands and automate your workflows
- It's easy to learn and has a smooth learning curve



Total is a modern and modular Node framework following the MVC design architecture.

- Total Platform includes a collection of libraries and packages
- Best development practices inspired from ASP.NET MVC and Django for web app development
- It is super fast and requires very little maintenance
- Incorporates Total Eshop, one of the best Node.js e-commerce systems.
- Extensive database support, integrable with PostgreSQL, MongoDB, CouchDB, Firebase, Redis, etc.
- The total.js platform is optimized to be independent. You can use Notepad or Nano code editors for developing web applications without requiring any additional tools or learning new programming languages.

Benefits of Using NodeJS

This website uses cookies to ensure you get the best experience on our website [More info](#)

[OK](#)



Benefits of Using NodeJS

- Ecosystem NPM
- Active Community Support
- Scalability
- Single place
- Constantly Updated
- Speed and Performance
- multiple HTTP requests



1. ==
2. ==
3. ==

1. A massive **ecosystem** of packages – **NPM**(The Node Package Manager) is a collaborative registry of software for JavaScript. It holds a collection of open source libraries and reusable modules that provide ready-made solutions saving development time.
2. **Active Community Support** – A vast community of experienced developers stands behind NodeJS. If you ever hit a snag, the community can help you get instant solutions. You don't need to reinvent the wheel, reusing the developed application components in your website.
3. **Scalability** – App-based startups are the right user base for NodeJS. Your business can use it to develop lightweight, scalable, and highly performative systems with real-time responses. You can add new modules, introducing new features, expanding your user base and data sets without compromising the ingenuity of your app.
4. Everything in a **single place** – One code for both frontend and backend. The app requires only one deployment.
5. **Constantly Updated** – Both the V8 engine and Node are frequently updated, with performance boosters, security patches, and support for modern JavaScript features.
6. **Speed and Performance** – The use of JavaScript also means that transforming JSON data is fast by default. Node.js executes the JS code outside the browser, which significantly improves the performance of the application. Plus using the



of single module caching allows the web application to load faster and be more responsive to user requests. Thus using Node.js ensures very high-speed websites.

7. Simultaneous handling of multiple HTTP requests – With non-blocking I/O systems you can process several requests concurrently. The system can handle concurrent request handling efficiently better than others including Ruby or Python. The incoming requests are arranged in a stack formation and executed quickly and efficiently.

These are just some of the advantages of using Node.js for server-side development. Wondering which IDE to use for Node.js programming? Check out our blog on the [best JavaScript IDE](#).

Confused Which Web Frameworks to Use?

Don't worry. We have Node.js experts who will help you find the right frameworks and tools for your project. For free.

[Book Your Free Consultation →](#)

How to Choose the Right Framework for Web Apps?

Node.js is a versatile tool for JavaScript that can be [used for developing](#) –

- Real-time web applications
- Streaming applications
- Messaging apps
- Chat apps
- Social media apps
- Virtual emulators
- Multiplayer games

This website uses cookies to ensure you get the best experience on our website [More info](#)

OK



We have curated a comparison table to help you decide which framework to choose according to your project requirements.

USE	WHEN
Meteor	<ul style="list-style-type: none"> ● Meteor.js is suitable for you if your development team is good with JavaScript. ● You can use it for quick prototyping as you can use the same code across various platforms, be it a web app or mobile app. ● Meteor allows you to deploy live updates while testing your web apps. You can even update installed apps without affecting an ongoing user session. ● It has a smooth learning curve and can be quite enjoyable to learn. ● With the built-in core of Meteor, you are able to develop real-time applications. ● You should choose Meteor if you have a huge team and are working under strict deadlines. It will allow team collaboration for real-time app projects and cross-platform products.
Express	<ul style="list-style-type: none"> ● It is a flexible framework suitable for all kinds of web and mobile applications. It supports small-scale websites and large-scale heavy traffic web apps too. ● You can use it for any enterprise application with cross-browser compatibility. ● Express is a great choice if you want to develop testable RESTful apps. You can develop web apps faster with this framework since it provides robust tools for API generation. ● It has powerful tools for testing and debugging. Develop websites with high-security features and immediate error handling. ● Express is the best choice for new developers and inexperienced teams



- | | |
|-------|--|
| Sails | <ul style="list-style-type: none">● Sails can build production-ready applications in just a few weeks. It carries the Ruby on Rails MVC architecture pattern and enables quick prototyping.● Sails is apt for developing chat rooms and messaging applications owing to its streamlined data and portable components and reusable-middleware functions.● It is extremely compatible with Socket.io and integrates seamlessly. You can use this combination for developing heavy-traffic social media platforms.● Game development with animations and graphics can also be implemented with Sails.js● It is robust and opinionated, so we would suggest avoiding this framework for small-scale websites and web app projects. |
| Nest | <ul style="list-style-type: none">● Sails.js is preferred for the development of custom enterprise applications.● Sails can build production-ready applications in just a few weeks. It carries the Ruby on Rails MVC architecture pattern and enables quick prototyping.● Sails is apt for developing chat rooms and messaging applications owing to its streamlined data and portable components and reusable-middleware functions.● It is extremely compatible with Socket.io and integrates seamlessly. You can use this combination for developing heavy-traffic social media platforms.● Game development with animations and graphics can also be implemented with Sails.js● It is robust and opinionated, so we would suggest avoiding this framework for small-scale websites and web app projects. |



	<ul style="list-style-type: none"> ● Conflict resolution and have highly performant when working with data-heavy applications. ● Works well when developing real-time chat apps and adding messaging modules in existing web apps. ● Can be used for offline coding and collaboration. ● Enhanced user experience, extremely user friendly
Koa	<ul style="list-style-type: none"> ● Koa is the perfect Node.js framework for you if you are developing high-performance applications. ● The framework is mainly used for developing complex, extensive apps with speed and efficiency. ● It is a good choice for large development teams with experienced developers and engineers as it has a steep learning curve. ● If you want to provide personalized content to users, this next-generation framework works for you. It can maintain different versions of content in a single URL. ● It does not have well-built community support and is still fairly new. ● We would recommend you use this framework on a trial basis for the rapid development of web services, i.e. APIs.
Loopback	<ul style="list-style-type: none"> ● Loopback is an ideal choice when developing powerful end-to-end APIs and handling API requests. ● Creating REST APIs and modules with minimal development time. ● It can protect your website against vulnerabilities, providing security features and authentication features. ● If you require fast and efficient deployment of web solutions, Loopback can be used as an external API service.
Hapi	<ul style="list-style-type: none"> ● Hapi is extremely secure. We suggest using it when security is of the



	<ul style="list-style-type: none"> ● Developers mostly use it to build proxy servers and API services. ● You can also use hapi.js for developing single-page applications. It is compatible with any JavaScript front-end framework like AngularJS, Vue or ReactJS.
Total	<ul style="list-style-type: none"> ● Total.js framework is used for developing customized web apps. ● It is a good option for developing messenger applications and eCommerce solutions. ● Total supports the development of Content Management Systems and programming interfaces such as Flow and Node.js CMS. ● If you need to develop apps within a short time frame, Total.js can help you speed up the development process.
Adonis	<ul style="list-style-type: none"> ● We suggest using Adonis.js for the development of super-fast web applications and web services. ● It can handle JSON responses efficiently and will keep your web app secure and interactive. ● If you require email support for your web app, Adonis is the right choice. ● You can also develop data-driven applications using Adonis. With its own ORM, Adonis can be used to create complex queries and handle migrations.

Frequently Asked Questions

What makes Node.js special?

Node.js is a JavaScript runtime environment for executing server code. It encompasses properties such as non-blocking asynchronous I/O, programmer productivity, and unified frontend and backend development. These make Node.js one of the most productive and efficient web development tools in terms of

This website uses cookies to ensure you get the best experience on our website [More info](#)

OK

DEVELOPMENT. Real-time applications, messenger apps, microservices, backend API and more are developed using NodeJS.

Which is the best Node JS framework?

There are many frameworks that allow optimal usage of Node.js features and advantages. Express is the oldest and most famous web framework for Node.js. Meteor, Loopback and Sails are a good choice for a monolithic, **batteries included** framework while for more **lightweight** options you can work with Derby, Adonis and Nest.

Conclusion

Node.js is an enterprise-grade framework for mobile app development that enables using JavaScript on servers along with the client-side. Even though Node.js frameworks aren't essential for backend development, they make a huge difference in development efficiency, product performance, and code quality.

Having access to templates, libraries, modules, and middleware, developers can significantly speed up the development process. They are able to handle thousands of requests and callbacks, and can easily integrate Node.js with client-side frameworks.

At Monocubed, we have experienced JavaScript developers who will assist you in all your development needs. Feel free to [contact us](#) anytime for a free consultancy about NodeJS development.

[Subscribe](#)

Author Bio

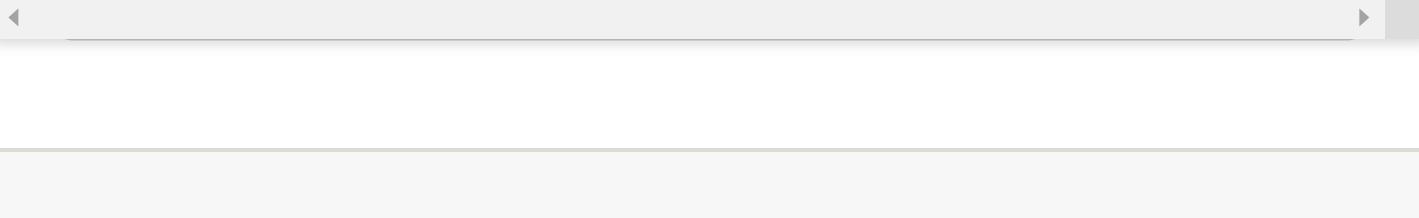


Jeel Patel

Founder of Monocubed

This website uses cookies to ensure you get the best experience on our website [More info](#)

[OK](#)



Let's Get To Know Each Other

We will validate your idea and provide a personalized quote. Without any commitments.
Without any fees.

 Name* Email*[Book Your Free Consultation](#)

All your data will remain strictly confidential. For more information check our [Privacy policies](#).

Our Office

**Canada**

2 County Court Blvd., Suite 400,
Brampton, Ontario L6W 3W8

HAVE QUESTIONS?

Get in touch with us

Phone: +1 (437) 488-7337

Email: sales@monocubed.com

FOLLOW US



This website uses cookies to ensure you get the best experience on our website [More info](#)

[OK](#)



Copyright © 2022 Monocubed. All Rights Reserved

DMCA PROTECTED

This website uses cookies to ensure you get the best experience on our website [More info](#)

OK