CODE EVERYWHERE Follow
Jun 12 · 4 min read · Listen

NestJS and its Advantages for Backend Developers

NestJS is a framework for building efficient and scalable Node.js server-side applications built with and fully supported by TypeScript. It uses robust HTTP Server frameworks like Express or Fastify. Nest provides a level of abstraction above common Node.js frameworks and exposes its APIs to the developer. This gives a great amount of freedom to use third-party modules.

A good reason to choose NestJS over ExpressJS (one of the most popular Node.js frameworks) is the fact that when a new project in Node.js is started it is a clear architecture based on a few simple components (controllers, modules and providers). This gives great **ease to split applications into microservices**.

Are you wondering **why choose NestJS** as your backend framework? Keep reading.

What is NestJS?

As mentioned, NestJS is an **open-source, extensible, versatile, progressive Node.js framework** for creating compelling and demanding backend systems. It is currently the fastest-growing Node.js framework in TypeScript.

NestJS is used for writing **scalable, testable and loosely coupled applications**. It brings scalable Node.js servers to a whole new level. It supports databases like PostgreSQL, MongoDB, MySQL. **NestJS is heavily influenced by Angular, React and Vue** and offers dependency injection right out of the box.

As of January 2022, it has over **44k GitHub stars** and its weekly npm download rate is almost **180k**. It encourages developers to try, learn and use some well-known software development paradigms and its documentation has lots of examples, recipes and code sources.

NestJS is easily extendible, as it can be used with other libraries; versatile thanks to its adaptive fully-fledged ecosystem and progressive, bringing JavaScript features and design patterns.

Building blocks of NestJS

Modules: used to organize the code and split features into logical reusable units. Grouped TypeScript files are decorated with "@Module" decorator which provides metadata that Nest makes use of to organize the application structure.

Providers: also called services, which are designed to abstract any form of complexity and logic. Providers can be created and

Incredible features of NestJS

Recently, the NestJS framework has been gaining extreme popularity due to its incredible features. Some of them are:

- It leverages TypeScript — strongly typed language which is a superset of JavaScript
- Easy to use, learn and master
- Powerful Command Line Interface (CLI) to boost productivity and ease of development
- Detailed and well-maintained documentation
- Active codebase development and maintenance
- It is open-source (MIT license)
- Support for dozens of nest-specific modules that help you easily integrate with common technologies and concepts like TypeORM, Mongoose, GraphQL, Logging, Validation, Caching, WebSockets and much more
- Easy unit-testing applications
- Created for Monoliths and Micro-services (an entire section in the documentation regarding Microservice types of NestJS applications as well as techniques and recipes

Why choose NestJS

Java as a programming language is more than 20 years old. For many years, it was the main framework for creating business applications, but recently, the focus shifted towards Node.js. Because of its speed, Node.js took the developer world by storm allowing to enjoy all the benefits of JavaScript and writing both frontend and backend in the same programming language. Since NestJS is the fastest-growing framework for Node.js, let's talk about the reasons **why choose NestJS over your current framework**?

- A quick and efficient development process.
- Highly scalable and easy to maintain applications.
- Fastest growing Node.js framework for the past 3 years.
- Large community and support system.
- Nest has positioned itself at a unique crossroad of f… …velopment that many languages have struggled to find.
- The use of TypeScript helps ensure that Nest will remain relevant in the rapidly changing JavaScript landscape and gives developers less context switching.
- Great documentation.
- Easy unit testing.
- Built for large-scale enterprise applications.
- Nest provides an out of the box application architecture that allows developers and teams to create highly testable, scalable, loosely coupled and easily maintainable applications.
- With NestJS you can build Rest API's, MVC applications, microservices, GraphQL applications, Web Sockets or CLI's and CRON jobs.
- It uses the best of Node.js, TypeScript, and bullet-proof architectural patterns.
- The structure of the application in Nest is heavily based on Angular. The structure is very simple and allows more attention to be paid to the design of endpoints and their consumers, instead of the application structure.
- NestJS forces developers to use a specific architecture by introducing Angular like modules, services, and controllers, ensuring