

PROGRAMAÇÃO \_

FRONT-END \_

DATA SCIENCE \_

INTELIGÊNCIA ARTIFICIAL \_

DEVOPS \_

UX &amp; DESIGN \_

MOBILE \_

INOVAÇÃO &amp; GESTÃO \_

Artigos > **Programação**

# Node.JS: definição, características, vantagens e usos possíveis

**Camila Pessoa**


08/03/2022

COMPARTILHE



Node.JS é uma ferramenta amplamente utilizada no universo de desenvolvimento e todos que atuam na área de tecnologia já ouviram falar, nem que seja nos milhares de anúncios de cursos ou mesmo de vagas. Mas você se pergunta: afinal, [o que é Node.js](#)? Para que serve? Como ele atua?

Calma que, antes de olhar para o ambiente Node.JS, precisamos entender um pouco sobre [o que é o JavaScript](#) e como ele funciona. Vamos lá!?



**Matricule-se na escola de PROGRAMAÇÃO**

Junte-se a uma comunidade de **+500 mil** estudantes

- Acesso a **TODOS** os cursos em uma única assinatura
- Novos lançamentos a cada semana
- Desafios práticos

**SAIBA MAIS**

## Node.JS e JavaScript

JavaScript é uma linguagem de programação que originalmente foi desenvolvida para trazer maior interatividade aos **websites** através da manipulação do *DOM* (Document Object Model). Vamos conferir a definição do livro que é uma referência na temática, o **Eloquent JavaScript**:

*O JavaScript foi introduzido em 1995 como uma forma de adicionar dinamicidade à páginas da web no navegador Netscape Navigator. Desde então, a linguagem foi adotada por todos os outros principais navegadores gráficos da web. Ela tornou possíveis o desenvolvimento das aplicações modernas da web - aplicações com os quais você pode interagir diretamente sem recarregar a página a cada ação. JavaScript também é usado em sites mais tradicionais para fornecer várias formas de interatividade de forma mais inteligente.*

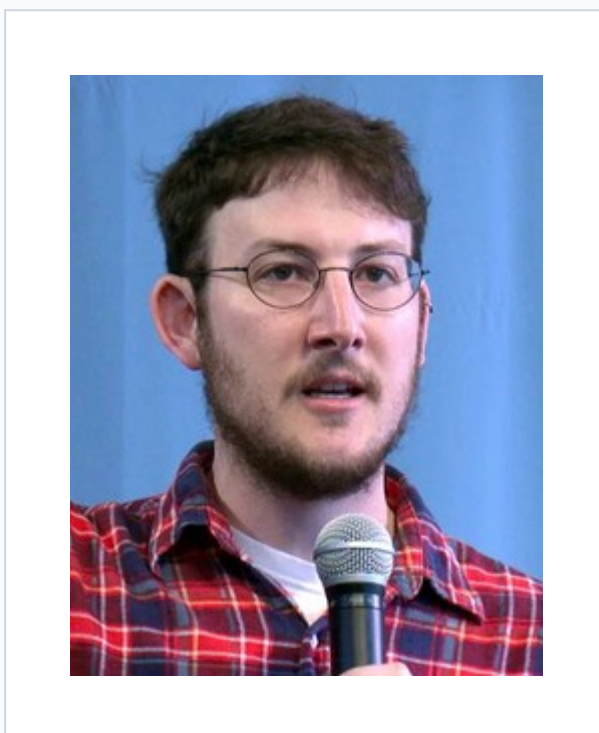
Fonte: [Tradução livre do livro Eloquent JavaScript](#)



MATRICULE-SE

Portanto, de forma resumida, podemos entender que o **JavaScript** foi pensado para ser rápido, dinâmico e acessível. A *linguagem interpretada* possibilita subir ou trabalhar em suas aplicações sem precisar configurar todo um ambiente complexo.

Essas características motivaram o engenheiro de software Ryan Dahl a desenvolver um ambiente, como um programa que você instala no seu computador, que trabalha com a linguagem JavaScript fora do navegador e pelo lado do servidor, via terminal, de uma maneira menos restritiva.



Fonte: [Ryan Dahl](#)

## Tudo bem, mas e o Node.JS?

O JavaScript nasceu para atender demandas voltadas ao Front e como as necessidades aumentam de acordo com o crescimento tecnológico, surgiu a ideia de utilizar uma mesma linguagem no lado do cliente e do servidor para otimizar processos e serviços. Dessa forma, o Node.JS aparece como uma alternativa viável para [programação Back-End](#) por se tratar de um ambiente para desenvolvimento utilizando a linguagem JavaScript.

De acordo com sua definição oficial, o Node é um **runtime**, que nada mais é do que um conjunto de códigos, API's, ou seja, são bibliotecas responsáveis pelo tempo de execução (é o que faz o seu programa rodar) que funciona como um interpretador de JavaScript fora do ambiente do **navegador web**.

[MATRICULE-SE](#)

É importante frisar que o Node.JS é um ambiente de execução assíncrono, isto é, trabalha de modo a não bloquear no momento da execução da aplicação, delegando os processos demorados a um segundo plano.

Ele dá muito certo com os servidores de arquitetura “**single threaded**”, isto significa que todos os pedidos para o servidor são executados no mesmo tópico - em vez de serem gerados em processos separados. Um dos grandes diferenciais da parceria Node.JS e Javascript é o bom desempenho no uso de APIs, já que o Javascript faz bastante uso de APIs assíncronas.

Você deve estar se perguntando: “Beleza, já entendi o que é o Node.JS e seu diferencial quando o assunto é o seu uso com o Javascript, mas o que isso significa?” Vamos entender um pouco mais... vem comigo, que te explico...

## Como o Node.JS funciona?

O **Node** é capaz de interpretar um código **JavaScript**, igual ao que o navegador faz. Sendo assim, quando o navegador recebe um comando em JavaScript, ele o interpreta e depois executa as instruções fornecidas.

O Node é fantástico (totalmente “sem parcialidade”, aqui pessoal, juro!) porque torna possível o envio de instruções (os nossos códigos) sem precisar de um navegador ativo, basta ter o Node.JS instalado e utilizar o terminal para executar um programa construído em JavaScript.

Além disso, você pode utilizar apenas uma linguagem de programação para tratar requisições entre cliente e servidor. Sensacional, não é?!

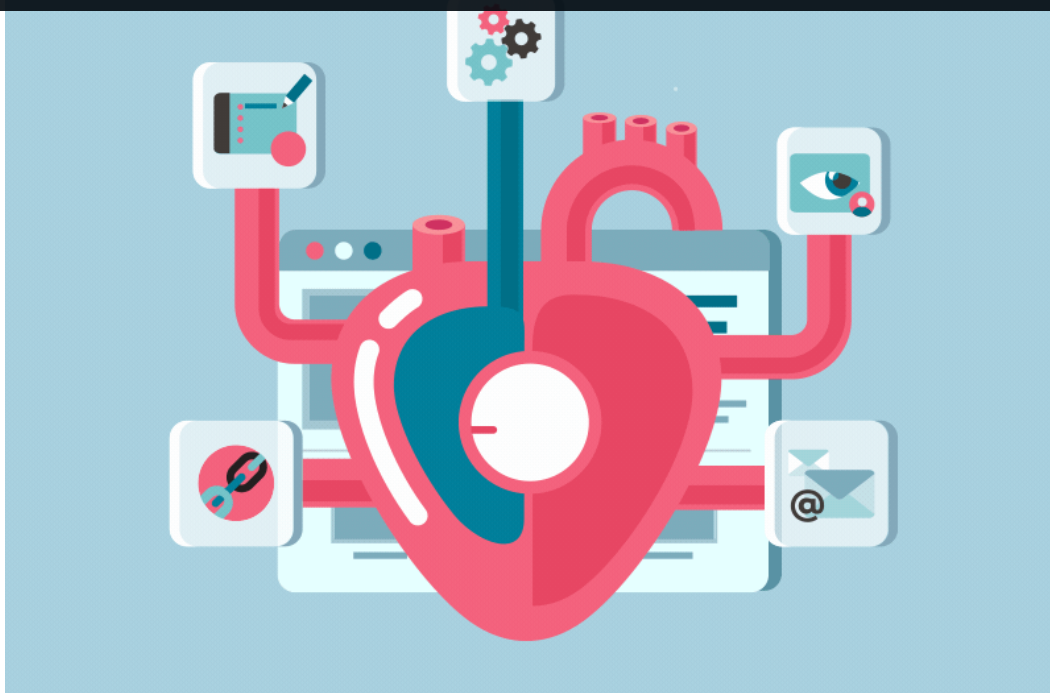


Para que todo esse processo seja possível de ocorrer fora do navegador, o Node utiliza uma outra ferramenta chamada de **Chrome's V8 JavaScript Engine**. É esse motor V8 do Chrome que compila e executa o código JavaScript no lugar de apenas interpretá-lo.

*Ainda parece muito confuso? Vem comigo que te explico melhor...*

O motor V8 da Google é o centro, o coração que processa todo o código JavaScript do navegador para que sua máquina compreenda e disponibilize os recursos e interações. Só é possível visualizar páginas em JavaScript ou trabalhar com um código pelo Node.JS devido ao processamento do motor V8 do Chrome.





Graças ao núcleo baseado em JavaScript e no motor V8 da Google, o Node.JS possui algumas características que, dependendo do cenário, podem ser consideradas como *benefícios* da plataforma, vamos conferir algumas?

## Características do Node.JS

Node.JS pode ser utilizado nas famosas APIS Rest, web scrapping, chatbots, IoT, web servers, aplicações Desktop, tudo devido a sua característica altamente versátil. Dentre as principais, podemos citar:

- **Multiplataforma:** permite criar desde aplicativos desktop, aplicativos móveis e até sites SaaS;
- **Multi-paradigma:** é possível programar em diferentes paradigmas, como: Orientado a Objetos, funcional, imperativo e dirigido à eventos;
- **Open Source:** é uma plataforma de código aberto, isso significa que você pode ter acesso ao código fonte do Node.JS e realizar suas próprias customizações ou mesmo contribuir para a comunidade de forma direta;
- **Escalável:** Node.JS foi criado para construir aplicações web escaláveis, como podemos ver na sua [documentação oficial](https://nodejs.org/en/docs/).

**alura****MATRICULE-SE**

O Node.JS ainda apresenta uma curva de aprendizado relativamente menor comparada às outras linguagens voltadas ao Back-End e possui uma comunidade gigante disposta a ajudar. E sem dúvida a união dessas forças é o **super poder do Node.JS!**



Fonte: <https://www.shutterstock.com/>

## Empresas que usam Node.JS

Tudo bem, falamos das inúmeras qualidades do Node.JS mas, na prática... onde eu posso trabalhar com a ferramenta?

Por conta dos *super-poderes* do Node.JS, uma gama de gigantes no mercado atua diretamente com a plataforma em suas regras de negócio, confere só a listinha de algumas:

- **NetFlix:** substituiu o renderizador baseado em Java pelo de JavaScript com Node.JS para diminuir o tempo de espera nas requisições dos usuários e conseguiu uma redução de cerca de 1 minuto;
- **IBM:** criou o The Canary in the Gold Mine (CITGM), um utilitário de testes amplamente utilizado pela companhia;
- **LinkedIn:** o aplicativo **mobile** da maior rede profissional é construído em Node e HTML5, você pode conferir a matéria na íntegra [aqui](#);

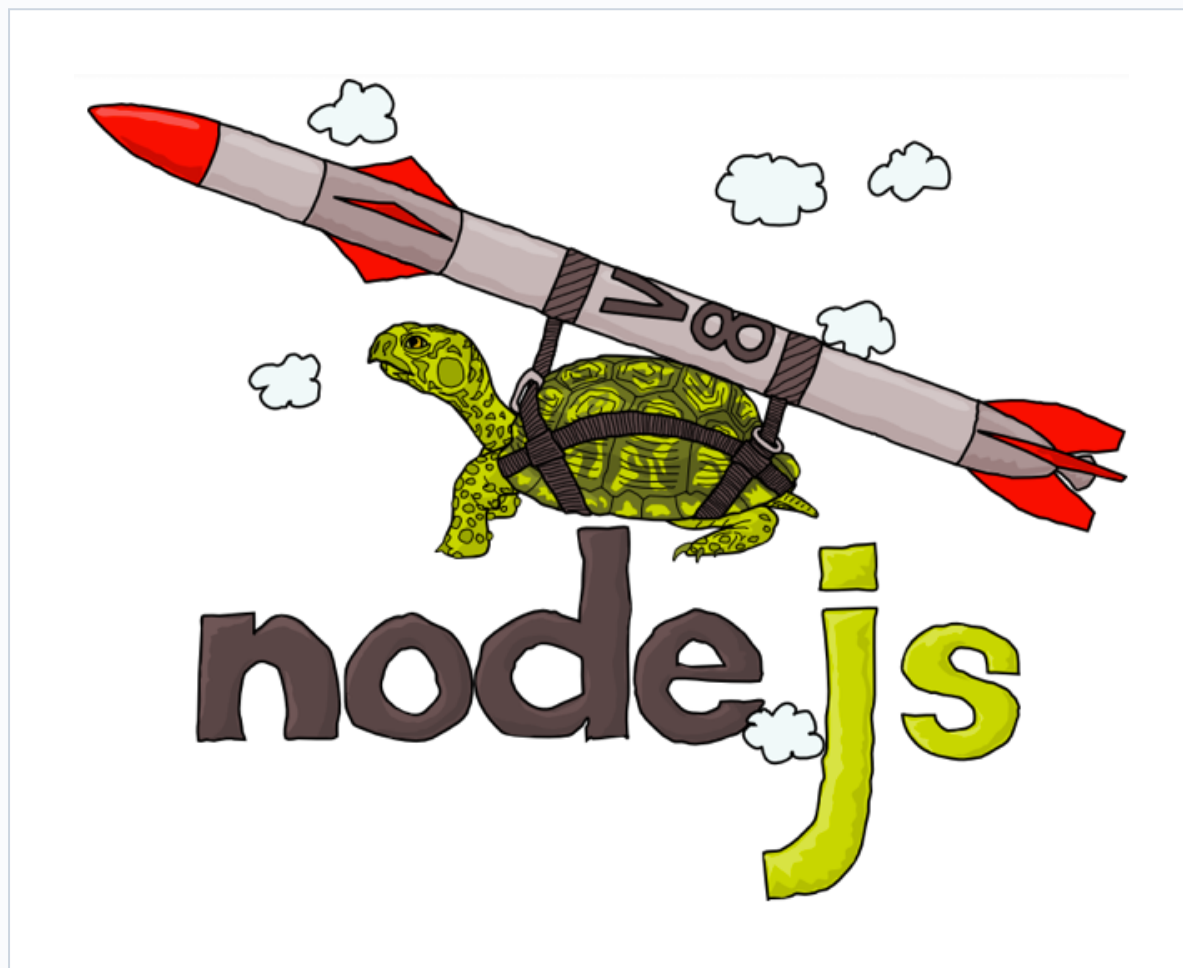
**alura**

- **Uber:** sistema de relacionamento entre usuários e condutores foi construído em Node.JS por conta de sua rápida resposta.

**MATRICULE-SE**

E... pasmem! Node.js foi inclusive utilizado pela **NASA** para criar uma API que integra três bases de dados em um só, justamente para otimizar o acesso aos dados. [Node.JS ajuda NASA a manter Astronautas seguros e arquivos acessíveis - em inglês.](#)

Node.JS está mesmo voando!!!



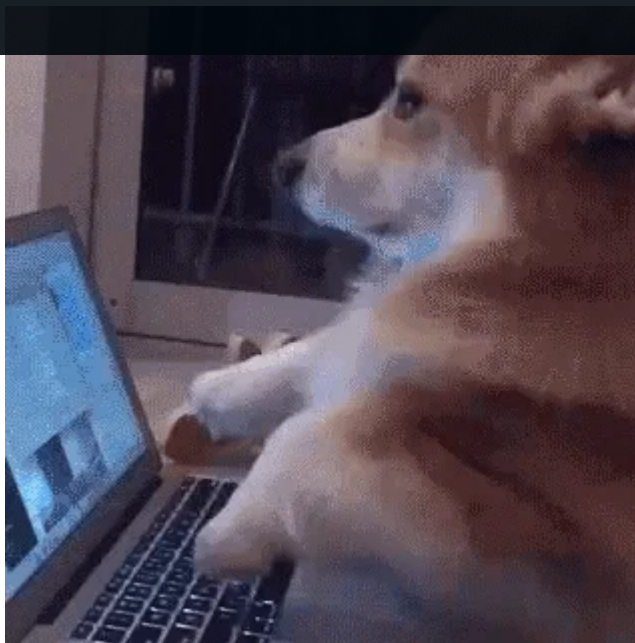
Fonte: <https://erickgledson.github.io/>

## Meu primeiro código em Node.JS

Para rodar a plataforma na sua máquina é necessário instalar o Node.JS. Caso você ainda não tenha, siga o tutorial [Como instalar o Node.js no Windows, Linux e macOS.](#)

E depois colocar a mão no código!





Fonte: <https://tenor.com/>

Aquele “Hello World!” para dar sorte:



```
console.log('Hello, World!')
```

E agora você pode subir seu primeiro servidor:



```
const http = require('http');  
const server = http.createServer(function (req, res) {  
  res.writeHead(200, {'Content-type': 'text/html'});  
  res.end('Hello Node!');  
});  
server.listen(3000);  
console.log('Servidor funcionando. Acesse: localhost:3000');
```

Ah, é importante lembrar que aqui o código está de acordo com a última versão estável, a versão LTS (Long-term support ou suporte de longo prazo,

Agora é só comemorar e aprofundar seus estudos em Node.JS!



## Conclusão

JavaScript é a linguagem mais usada no Brasil e no mundo. Sem dúvida o conhecimento nessa tecnologia é uma necessidade para qualquer desenvolvedor. O domínio de Node.JS pode ser um diferencial para quem deseja alçar voos mais altos em sua carreira, sobretudo o desenvolvedor web, e compreender mais sobre o ecossistema é fundamental para que a ferramenta sirva tenha alto nível de aproveitamento e qualidade de suas aplicações.

Compartilhe com a gente no servidor da Alura as suas impressões sobre Node.JS, se você é estudante ou se já trabalha na área: [Link para o Discord da Alura](#).

### E então, vamos aprender mais?

- [Formação JavaScript para Back-End](#)
- [Formação Iniciante em Programação](#)
- [Formação Node.js com Express](#)
- [Node.js v17.5.0](#)
- [O que é Node.JS](#)
- [Primeiros passos com Node.js - Casa do Código](#)

**alura**

O que é JavaScript

MATRICULE-SE

- [Node.js: O que é, quais as características e vantagens?](#)
- [O que é JavaScript? Hipster.tube](#)
- [O que é Node Documentação Microsoft](#)
- [Ecossistema Node.JS - Hipsters.tube](#)
- [JavaScript.com](#)

### Confira neste artigo:

- [Introdução](#)
- [Node.JS e JavaScript](#)
- [Tudo bem, mas e o Node.JS?](#)
- [Como o Node.JS funciona?](#)
- [Características do Node.JS](#)
- [Empresas que usam Node.JS](#)
- [Meu primeiro código em Node.JS](#)
- [Conclusão](#)




#### Camila Pessôa

Olá, sou a Camila ! Tenho 33 anos, sou mãe e ingressei na área de tecnologia por meio da robótica educacional. Participei do Bootcamp { Reprograma } com foco em Back-End /Node.js e curso Sistemas de Informação. Atualmente faço parte do Scuba-Team e tenho grande paixão por educação e tecnologia, pois acredito que essa combinação é transformadora!

[← Artigo Anterior](#)

[Próximo Artigo →](#)

 **alura**

[Como não aprender Java a Objetos: getters e setters](#)

[Artigo Anterior: Markdown com JS, hash e criptografia e mais Alura Stars](#)

[MATRICULE-SE](#)



Veja outros artigos sobre [Programação](#)

## Quer mergulhar em tecnologia e aprendizagem?

Receba a newsletter que o nosso CEO escreve pessoalmente, com insights do mercado de trabalho, ciência e desenvolvimento de software

**ME INSCREVA**

# alura

Nossas redes e apps



**Institucional**

**A Alura**



Sobre nós

**alura**

Fale conosco

Formações

Como Funciona

MATRICULE-SE

Para Empresas

Para Escolas

Política de Privacidade

Compromisso de Integridade

Termos de Uso

Status

Todos os cursos

Depoimentos

Instrutores(as)

Dev em <T>

Luri by ChatGPT

Conteúdos

Alura Cases

Imersões

Artigos

Podcasts

Artigos de educação corporativa

Fale Conosco

Email e telefone

Perguntas frequentes

Novidades e Lançamentos

Email\*ENVIAR

CURSOS

Cursos de Programação

Lógica | Python | PHP | Java | .NET | Node JS | C | Computação | Jogos | IoT



**Cursos de Front-end**

HTML | CSS | React | Angular | JavaScript | jQuery

**Cursos de Data Science**

Ciência de dados | BI | SQL e Banco de Dados | Excel | Machine Learning | NoSQL | Estatística

**Cursos de Inteligência Artificial**

IA para Programação | IA para Dados

**Cursos de DevOps**

AWS | Azure | Docker | Segurança | IaC | Linux

**Cursos de UX & Design**

Usabilidade e UX | Vídeo e Motion | 3D

**Cursos de Mobile**

React Native | Flutter | iOS e Swift | Android, Kotlin | Jogos

**Cursos de Inovação & Gestão**

Métodos Ágeis | Softskills | Liderança e Gestão | Startups | Vendas

**MATRICULE-SE**