



Introdução ao Node.js

Node.js é uma plataforma de tempo de execução de código aberto e multiplataforma que permite aos desenvolvedores construir aplicações web rápidas e escaláveis.

E by Eduardo Affonso Kasprovicz

O que é o Node.js?

Construído sobre o motor JavaScript V8 do Chrome, o Node.js usa um modelo de I/O de evento assíncrono não bloqueante. Essa abordagem é ideal para construir aplicações de rede intensivas em I/O, como servidores web e APIs.

Flexibilidade

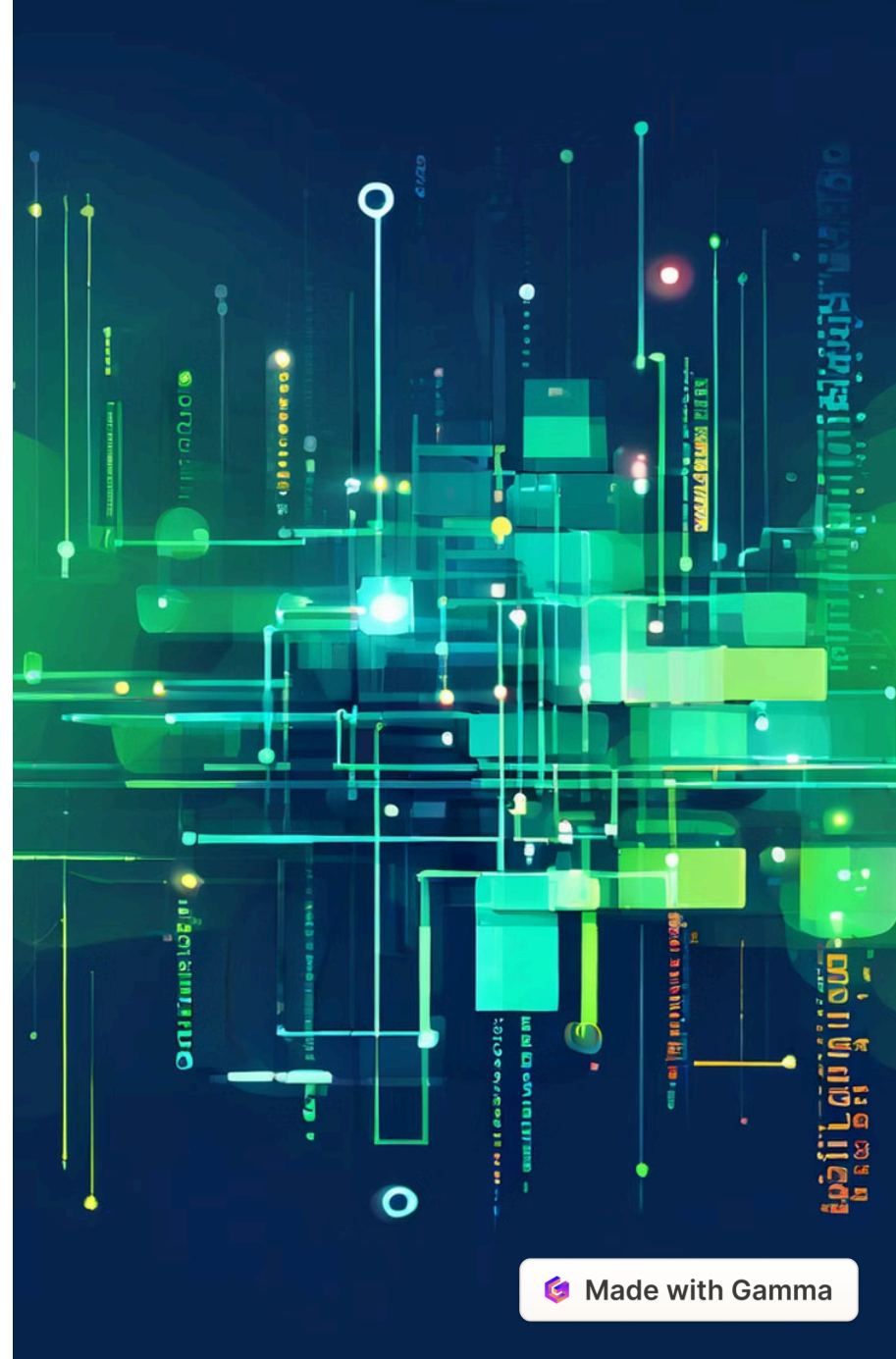
Pode ser usado para construir uma variedade de aplicações, desde servidores web a ferramentas de linha de comando.

Escalabilidade

Sua natureza assíncrona e não bloqueante o torna ideal para lidar com um grande número de solicitações simultâneas.

Comunidade Ativa

Tem uma comunidade de desenvolvedores vibrante e ativa, que oferece suporte e recursos valiosos.





Instalação e configuração do Node.js

A instalação do Node.js é um processo direto. O site oficial fornece instaladores para diferentes sistemas operacionais.

Download

Baixe o instalador adequado para seu sistema operacional.

1

Verificação

Verifique a versão instalada digitando 'node -v' no terminal.

3

2

Instalação

Execute o instalador e siga as instruções na tela.



Introdução ao NPM (Node Package Manager)

O NPM é o gerenciador de pacotes padrão do Node.js, facilitando a instalação, atualização e gerenciamento de bibliotecas e frameworks JavaScript.

1 Instalação

O NPM é instalado junto com o Node.js.

2 Busca de Pacotes

Use o comando 'npm search' para encontrar pacotes específicos.

3 Instalação de Pacotes

Use o comando 'npm install' para instalar pacotes.

4 Gerenciamento de Dependências

O NPM gerencia automaticamente as dependências de sua aplicação.

Gerenciamento de pacotes com NPM

O NPM oferece uma ampla gama de comandos para gerenciar seus pacotes, incluindo instalação, atualização, remoção e gerenciamento de dependências.

Instalação

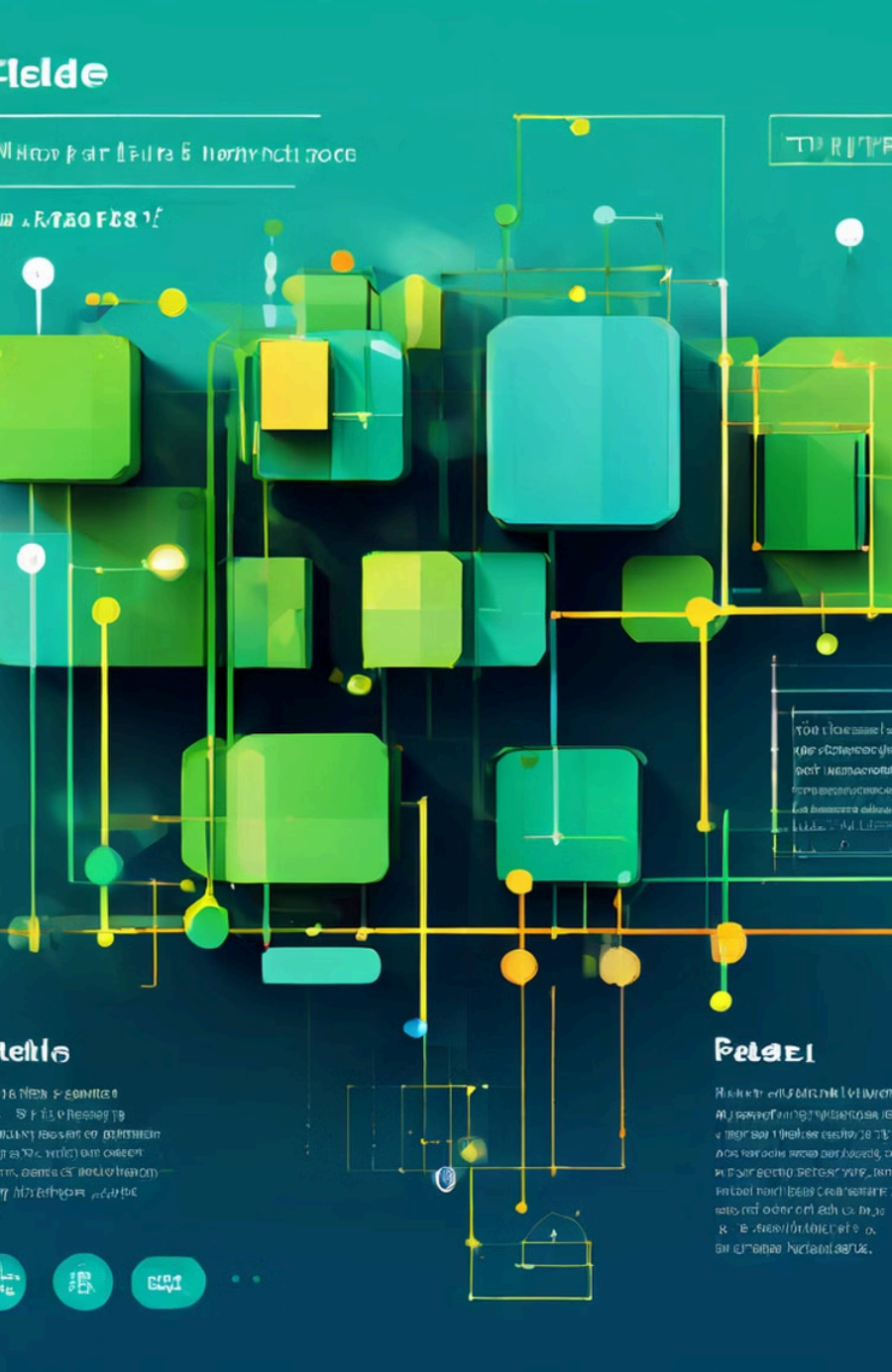
`'npm install package-name'`

Atualização

`'npm update package-name'`

Remoção

`'npm uninstall package-name'`



Criação de módulos em Node.js

Módulos são blocos de código reutilizáveis que organizam o código de sua aplicação em unidades lógicas.

1

Crie um Arquivo

Crie um arquivo JavaScript com o código do módulo.

2

Exporte Funções

Use 'module.exports' para exportar funções ou objetos.

3

Importe em Outro Arquivo

Use 'require' para importar o módulo em outro arquivo JavaScript.

Trabalhando com o sistema de arquivos em Node.js

O módulo 'fs' do Node.js fornece funcionalidades para interagir com o sistema de arquivos.

Operação	Descrição
'fs.readFile()'	Lê o conteúdo de um arquivo.
'fs.writeFile()'	Escreve dados em um arquivo.
'fs.rename()'	Renomeia um arquivo.
'fs.unlink()'	Exclui um arquivo.



Conclusão e próximos passos

Node.js é uma ferramenta poderosa para construir aplicações web robustas e escaláveis.



Explore a documentação

Explore a documentação oficial do Node.js para aprender mais sobre seus recursos.



Participe da comunidade

Participe da comunidade Node.js para trocar conhecimentos e obter ajuda.



Construa aplicações web

Comece a construir aplicações web simples usando Node.js e frameworks como Express.



Experimente frameworks

Experimente frameworks como Express, Koa e NestJS para construir aplicações web complexas.

