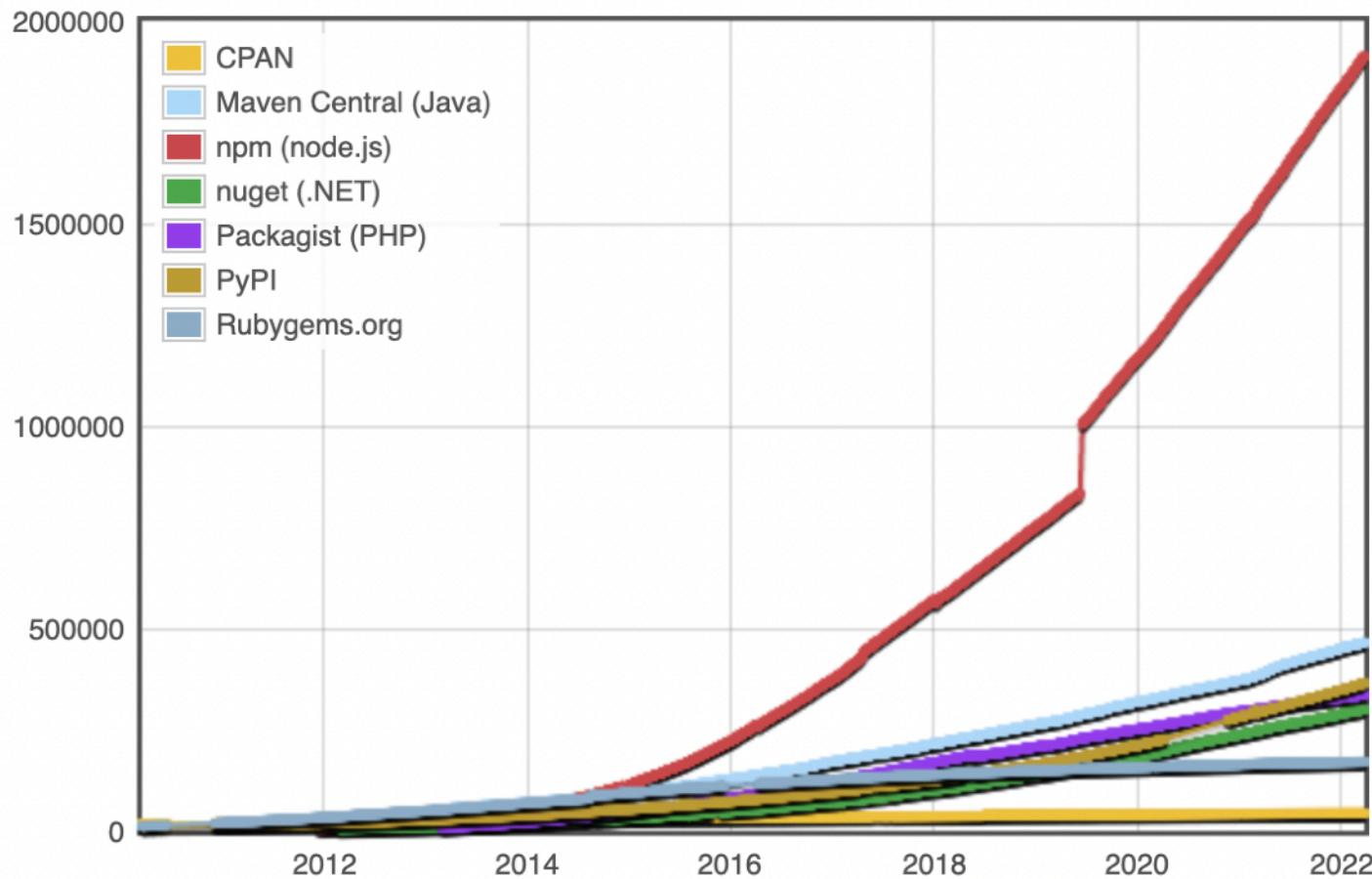




опыт

O NPM, ou Node Package Manager, é um gerenciador de pacotes totalmente escrito em JavaScript e criado por Isaac Z. Schlueter, um dos principais desenvolvedores do Node.js

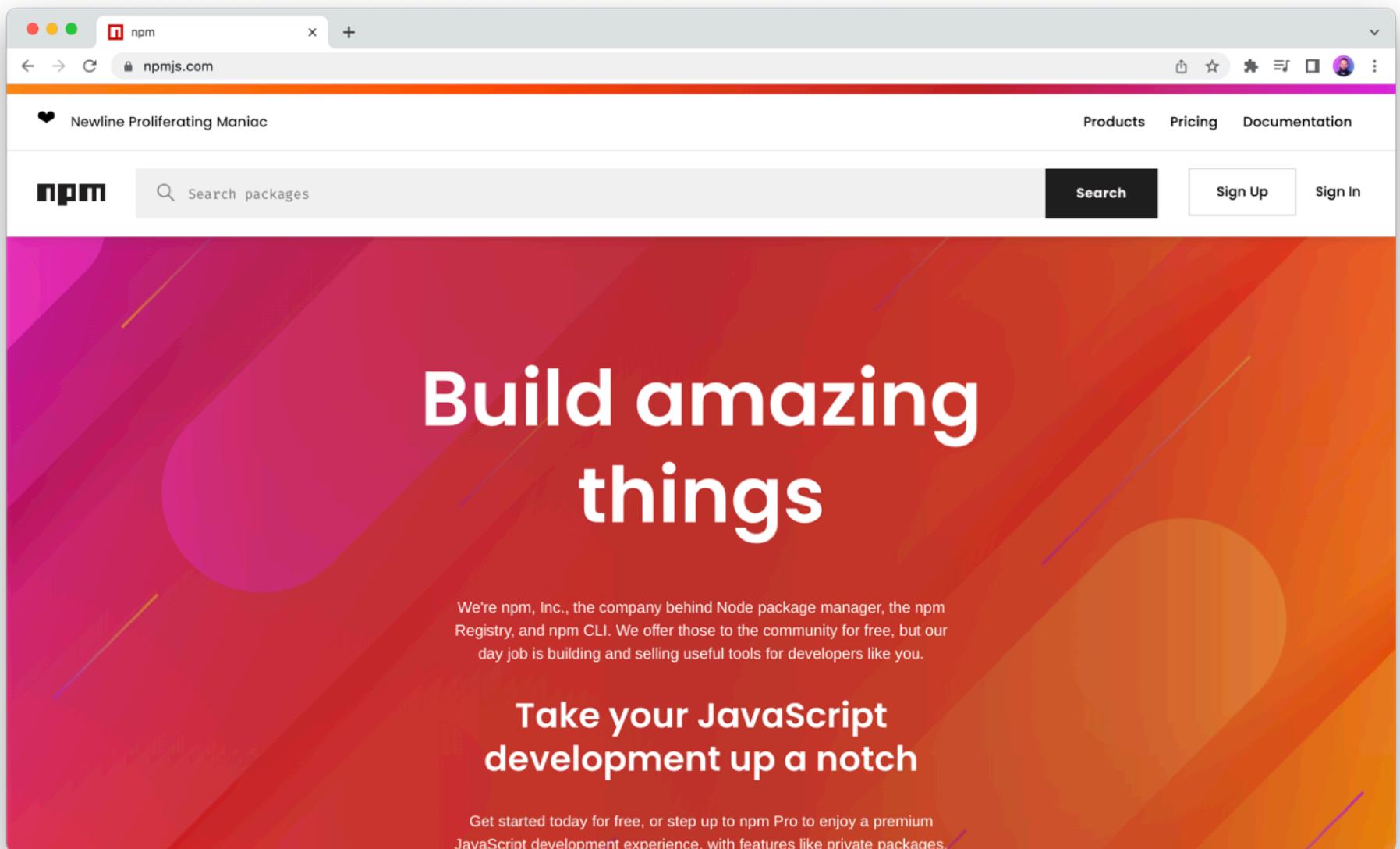


<http://www.modulecounts.com>

	<b>Mar 17</b>	<b>Mar 18</b>	<b>Mar 19</b>	<b>Mar 20</b>	<b>Mar 21</b>	<b>Mar 22</b>	<b>Mar 23</b>	<b>Avg Growth</b>
<a href="#">Clojars (Clojure)</a>	28823	28825	28826	28827	28828	28831	28833	2/day
<a href="#">CPAN</a>	43253	43256	43257	43258	43260	43263	43265	2/day
<a href="#">CPAN (search)</a>								5/day
<a href="#">CRAN (R)</a>	19011	18997	18998	18984	18994	18998	19015	2/day
<a href="#">Crates.io (Rust)</a>	79123	79180	79258	79303	79364	79457	79530	71/day
<a href="#">Crystal Shards</a>								2/day
<a href="#">Drupal (php)</a>	48380	48389	48393	48394	48398	48401	48406	4/day
<a href="#">DUB (dlang)</a>	2151	2151	2152	2153	2153	2153	2155	1/day
<a href="#">Gopm (go)</a>								7/day
<a href="#">Hackage (Haskell)</a>	16437	16440	16440	16441	16441	16441	16443	1/day
<a href="#">Hex.pm (Elixir/Erlang)</a>	12777	12781	12782	12785	12786	12790	12794	3/day
<a href="#">Julia</a>								0/day
<a href="#">LuaRocks (Lua)</a>	3070	3072	3072	3072	3072	3072	3072	1/day
<a href="#">Maven Central (Java)</a>	464183	464326	464477	464612	464739	464886	465098	158/day
<a href="#">MELPA (Emacs)</a>								2/day
<a href="#">Nimble (Nim)</a>	1828	1828	1829	1829	1829	1831	1831	1/day
<a href="#">npm (node.js)</a>	1908319	1908658	1909419	1909982	1910902	1912053	1913260	823/day
<a href="#">nuget (.NET)</a>	298501	298663	298899	298969	299255	299351	299476	167/day
<a href="#">Packagist (PHP)</a>	336077	336204	336276	336316	336418	336583	336708	107/day
<a href="#">Pear (PHP)</a>	603	603	603	603	603	603	603	0/day
<a href="#">Perl 6 Ecosystem (perl 6)</a>	1937	1937	1938	1938	1938	1938	1938	0/day
<a href="#">PyPI</a>	363582	363808	364007	364170	364369	364560	365379	288/day
<a href="#">Rubygems.org</a>	170733	170746	170754	170762	170767	170783	170800	12/day
<a href="#">Vim Scripts</a>	5536	5536	5536	5536	5536	5536	5536	0/day



O NPM é composto por 3 partes: site, Command Line Interface (CLI) e registry

A screenshot of the npmjs.com homepage displayed in a web browser. The page features a vibrant red and orange abstract background. In the center, the word "Build amazing things" is written in a large, white, sans-serif font. Above this text, there's a smaller paragraph about npm, Inc. Below it, another section encourages users to "Take your JavaScript development up a notch". At the bottom, a call to action invites users to "Get started today for free, or step up to npm Pro". The top navigation bar includes links for "Products", "Pricing", and "Documentation". A search bar is located at the top left, and a user profile icon is at the top right.

npm

↳ npm

↳ npmj.com

Newline Proliferating Maniac

Products Pricing Documentation

npm Search packages

Search

Sign Up Sign In

# Build amazing things

We're npm, Inc., the company behind Node package manager, the npm Registry, and npm CLI. We offer those to the community for free, but our day job is building and selling useful tools for developers like you.

## Take your JavaScript development up a notch

Get started today for free, or step up to npm Pro to enjoy a premium JavaScript development experience, with features like private packages.

express - npm

npmjs.com/package/express

Nomadic People Migration Products Pricing Documentation

Search packages Search Sign Up Sign In

## express DT

4.17.3 • Public • Published a month ago

Readme Explore BETA 30 Dependencies 60,411 Dependents 267 Versions

# express

Fast, unopinionated, minimalist web framework for [node](#).

npm v4.17.3 downloads 105M/month linux passing windows passing coverage 100%

```
const express = require('express')
const app = express()

app.get('/', function (req, res) {
  res.send('Hello World')
})
```

Install

```
> npm i express
```

Repository

github.com/expressjs/express

Homepage

expressjs.com/

Weekly Downloads

24,209,208

Version License

4.17.3 MIT

# npm init [-y]

Comando utilizado para **inicializar o projeto** e criar um package.json com as informações iniciais, se utilizado com a opção -y pula o modo interativo

fullstackjs — zsh — 80x24

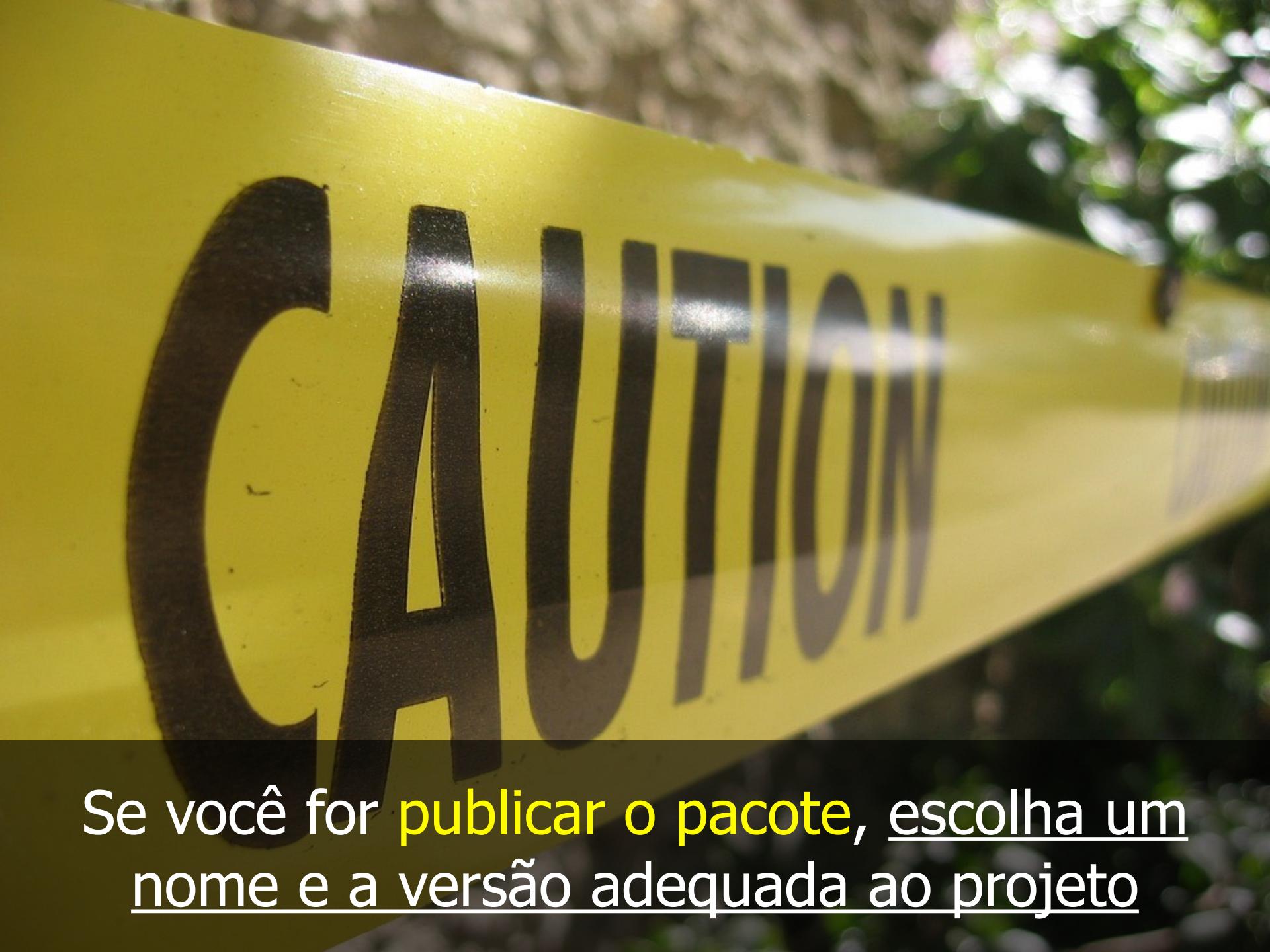
[rodrigobranas fullstackjs \$ npm init -y  
Wrote to /Users/rodrigobranas/development/workspace/branas/fullstackjs/fullstackjs/package.json:

```
{  
  "name": "fullstackjs",  
  "version": "1.0.0",  
  "description": "",  
  "main": "index.js",  
  "scripts": {  
    "test": "echo \\\"Error: no test specified\\\" && exit 1"  
  },  
  "keywords": [],  
  "author": "",  
  "license": "ISC"  
}
```

rodrigobranas fullstackjs \$ █

package.json — fullstackjs

```
{}
1  {
2    "name": "fullstackjs",
3    "version": "1.0.0",
4    "description": "",
5    "main": "index.js",
6    "scripts": {
7      "test": "echo \\\"Error: no test specified\\\" && exit 1"
8    },
9    "keywords": [],
10   "author": "",
11   "license": "ISC"
12 }
```



CAUTION

Se você for publicar o pacote, escolha um nome e a versão adequada ao projeto

O package.json é o **descriptor principal** de um projeto Node.js, responsável por conter as informações, configurações, dependências, scripts e muito mais

# Propriedades do package.json

**name, version, description, author, contributors, homepage, keywords, license, funding:** importante quando um projeto é publicado como dependência no npm

**main:** identifica o entrypoint do projeto

**config:** configurações utilizadas pelo projeto

**scripts:** scripts utilizados pelo projeto

**dependencies:** dependências

**devDependencies:** dependências de desenvolvimento

**peerDependencies:** dependências compatíveis

**bundleDependencies:** dependências empacotadas

**optionalDependencies:** dependências opcionais

**private:** não permite publicar o projeto no npm

**engines:** requisitos mínimos da versão do Node.js

**os:** requisitos de sistema operacional

**cpu:** requisitos de cpu



Já ouviu falar em semantic versioning ou  
SemVer?

package\_name@major.minor.patch

# MAJOR.MINOR.PATCH

MAJOR: breaking changes

MINOR: add functionalities with backwards compatibility

PATCH: bug fixes

# SemVer

1.2.0 is 1.2.0

$\wedge$ 1.2.0 is  $\geq 1.2.0 < 2.0.0$

$\sim$ 1.2.0 is  $\geq 1.2.0 < 1.3.0$

$\wedge$ 1 is 1

$\sim$ 1.2 is 1.2

\* most recent version

```
npm install package_name@version
```

Comando utilizado para instalar um determinado pacote, podendo ser pelo nome, url, tarball e também informando a versão específica

aliases: i, add, ins, insta, isnt, isntal, isntall

# Parâmetros

- P, --save-prod: salva em dependencies (é a opção padrão)
- D, --save-dev: salva em devDependencies, deve ser utilizado para dependências utilizadas na máquina local
- O, --save-optional: salva em optionalDependencies
- no-save: Não salva no package.json  
(comportamento padrão anterior)
- E, --save-exact: salva a versão exata
- B, --save-bundle: salva em bundleDependencies
- g, --global: instala de forma global (utilizado para CLI e pacotes similares)

```
[rodrigobranas fullstackjs $] fullstackjs — zsh — 80x24  
[rodrigobranas fullstackjs $] npm i typescript  
added 1 package, and audited 2 packages in 2s  
found 0 vulnerabilities  
[rodrigobranas fullstackjs $] cat package.json  
{  
  "name": "fullstackjs",  
  "version": "1.0.0",  
  "description": "",  
  "main": "index.js",  
  "scripts": {  
    "test": "echo \\\"Error: no test specified\\\" && exit 1"  
  },  
  "keywords": [],  
  "author": "",  
  "license": "ISC",  
  "dependencies": {  
    "typescript": "^4.6.2"  
  }  
}  
[rodrigobranas fullstackjs $]
```



CAUTION

Evite instalar pacotes em modo **global**  
com a opção `-g`, `--global`

# package-lock.json

Ao instalar pacotes é possível utilizar algumas opções no versionamento como ^ e ~, permitindo atualizar de acordo com lançamento de novos minors ou patches e isso cria um risco de segurança

O package-lock.json **registra a versão exata que está sendo utilizada e sempre baixa a mesma, incluindo também o checksum da dependência, garantindo que é a mesma que foi instalada originalmente**

## { package-lock.json X

```
3450      "resolved": "https://registry.npmjs.org/typesarray/-/typesarray-1.1.0.tgz",
3451      "integrity": "sha512-zdu8XMNEDepKKR+XY0XAVPtWui0ly0NtohUscw+UmaHiAWT8hrV1rr//H6V+0DvJ3Q0Q19S979M0laL
3452      "dependencies": {
3453        "is-typedarray": "^1.0.0"
3454      }
3455    },
3456    "node_modules/typescript": {
3457      "version": "4.6.2",
3458      "resolved": "https://registry.npmjs.org/typescript/-/typescript-4.6.2.tgz",
3459      "integrity": "sha512-HM/hFigTBHZhLXshn9sN37H085+hQGeJHJ/X7LpBWLID/fbc2acUMfU+lGD98X81sKP+pFa9f0DZmC
3460      "bin": {
3461        "tsc": "bin/tsc",
3462        "tsserver": "bin/tsserver"
3463      },
3464      "engines": {
3465        "node": ">=4.2.0"
3466      }
3467    },
3468    "node_modules/universalify": {
3469      "version": "0.1.2",
3470      "resolved": "https://registry.npmjs.org/universalify/-/universalify-0.1.2.tgz",
3471      "integrity": "sha512-rBJeI5CXAlmy1pV+617WB9J63U6XcazHHF2f2dbJix4XzpUF0RS3Zbj0FGIOCAva5P/d/GB0YaACQ1
3472      "engines": {
3473        "node": ">= 4.0.0"
3474      }
3475    },
3476  }
```

# npm install

Utilizado para instalar todas as dependências do projeto

```
[rodrigobranas fullstackjs $] fullstackjs — zsh — 80x24  
[rodrigobranas fullstackjs $] npm install  
  
added 334 packages, and audited 335 packages in 1s  
  
28 packages are looking for funding  
  run `npm fund` for details  
  
found 0 vulnerabilities  
rodrigobranas fullstackjs $ ]
```

```
npm uninstall package_name
```

Comando utilizado para desistalar um determinado pacote, sendo removido de todos os lugares do package.json como dependencies, devDependencies, optionalDependencies

aliases: remove, rm, r, unlink

```
[rodrigobranas fullstackjs $] fullstackjs — zsh — 80x24
[rodrigobranas fullstackjs $] npm r typescript
removed 1 package, and audited 1 package in 133ms
found 0 vulnerabilities
[rodrigobranas fullstackjs $] cat package.json
{
  "name": "fullstackjs",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC"
}
[rodrigobranas fullstackjs $]
```

`npm run script_name`

Comando utilizado para **executar um script**, sendo importante para automatizar atividades como transpilar, rodar testes, gerar pacotes

aliases: `run-script`

The screenshot shows the Visual Studio Code interface with the following details:

- EXPLORER** sidebar: Shows the project structure under "FULLSTACKJS".
  - dist
  - node\_modules
  - src
    - Calculator.ts
    - index.ts
  - test
    - Calculator.test.ts
  - package-lock.json
  - package.json (highlighted in blue)
  - tsconfig.json
- Editor Area**: The "package.json" file is open.

```
{  
  "name": "fullstackjs",  
  "version": "1.0.0",  
  "description": "",  
  "main": "index.js",  
  "scripts": {  
    "compile": "tsc",  
    "test": "tsc && jest ./dist",  
    "start": "tsc && node ./dist/src"  
  },  
  "keywords": [],  
  "author": "",  
  "license": "ISC",  
  "dependencies": {  
    "@types/jest": "^27.4.1",  
    "jest": "^27.5.1",  
    "typescript": "^4.6.2"  
  }  
}
```
- Bottom Bar**: Shows "OUTLINE" and other UI elements.

```
[rodrigobranas fullstackjs $ npm run compile
```

```
> fullstackjs@1.0.0 compile
> tsc
```

```
[rodrigobranas fullstackjs $ ls -la
```

```
total 536
```

drwxr-xr-x	9	rodrigobranas	staff	288	Mar 24 08:12	.
drwxr-xr-x	6	rodrigobranas	staff	192	Mar 24 07:47	..
drwxr-xr-x	5	rodrigobranas	staff	160	Mar 24 08:12	dist
drwxr-xr-x	261	rodrigobranas	staff	8352	Mar 24 08:09	node_modules
-rw-r--r--	1	rodrigobranas	staff	257381	Mar 24 08:10	package-lock.json
-rw-r--r--	1	rodrigobranas	staff	366	Mar 24 08:10	package.json
drwxr-xr-x	3	rodrigobranas	staff	96	Mar 24 08:11	src
drwxr-xr-x	3	rodrigobranas	staff	96	Mar 24 08:11	test
-rw-r--r--	1	rodrigobranas	staff	11038	Mar 24 08:10	tsconfig.json
rodrigobranas		fullstackjs \$				



fullstackjs — zsh — 80x24

[rodrigobranas fullstackjs \$ npm run test

```
> fullstackjs@1.0.0 test  
> tsc && jest ./dist
```

**PASS** dist/test/Calculator.test.js

✓ Deve testar a calculadora

**Test Suites:** 1 passed, 1 total

**Tests:** 1 passed, 1 total

**Snapshots:** 0 total

**Time:** 0.154 s

Ran all test suites matching ./dist/i.

rodrigobranas fullstackjs \$



fullstackjs — zsh — 80x24

[rodrigobranas fullstackjs \$ npm run start

```
> fullstackjs@1.0.0 start
> tsc && node ./dist/src
```

4

rodrigobranas fullstackjs \$

`npx package_name`

Executa o binário de um determinado pacote, sem a necessidade de instalar globalmente

```
[rodrigobranas fullstackjs $ jest  
zsh: command not found: jest  
[rodrigobranas fullstackjs $ npx jest ./dist/test  
  PASS dist/test/Calculator.test.js  
    ✓ Deve testar a calculadora (1 ms)  
  
Test Suites: 1 passed, 1 total  
Tests:       1 passed, 1 total  
Snapshots:   0 total  
Time:        0.121 s, estimated 1 s  
Ran all test suites matching ./\dist\test/i.  
rodrigobranas fullstackjs $
```

```
rodrigobranas fullstackjs $ npx tsc --init
```

Created a new tsconfig.json with:

```
target: es2016
module: commonjs
strict: true
esModuleInterop: true
skipLibCheck: true
forceConsistentCasingInFileNames: true
```

You can learn more at <https://aka.ms/tsconfig.json>

```
rodrigobranas fullstackjs $
```

TS

Existem outras opções ao npm como ied, pnpm, npmd e Yarn, sendo o Yarn uma das alternativas mais utilizadas.

Todas elas utilizam os mesmos repositórios do npm mas oferecem uma experiência diferente, eventualmente melhorando a performance e os mecanismos de cache

O Yarn, que significa Yet Another Resource Negotiator, foi criado pelo Facebook em 2016 com o **objetivo de ser uma alternativa rápida, estável e segura ao npm** depois que eles tiveram problemas em projetos com uma grande quantidade de dependências

Entre as **principais funcionalidades** do Yarn estão:

- **cache** de dependências de forma centralizada permitindo inclusive rodar projetos sem node\_modules
- utilização de **plugins** para extender as funcionalidades
- aumento da **segurança** (**yarn.lock** e **checksum** para **validar o conteúdo das dependências**)
- uso de diferentes **protocolos** além do HTTPS
- **paralelização** durante a instalação

```
yarn add package_name
```

Utilizado para instalar dependências, da forma  
forma que o npm install



fullstackjs — zsh — 80x24

```
[rodrigobranas fullstackjs $ yarn add typescript]
```

```
yarn add v1.22.18
```

```
warning package-lock.json found. Your project contains lock files generated by tools other than Yarn. It is advised not to mix package managers in order to avoid resolution inconsistencies caused by unsynchronized lock files. To clear this warning, remove package-lock.json.
```

```
[1/4] 🔎 Resolving packages...
[2/4] 🚛 Fetching packages...
[3/4] ⚡ Linking dependencies...
[4/4] 🏗 Building fresh packages...
```

```
success Saved 1 new dependency.
```

```
info Direct dependencies
```

```
  └─ typescript@4.6.2
```

```
info All dependencies
```

```
  └─ typescript@4.6.2
```

```
✨ Done in 1.18s.
```

```
rodrigobranas fullstackjs $
```

**yarn install**

Utilizado para instalar todas as dependências do projeto



fullstackjs — zsh — 80x24

```
[rodrigobranas fullstackjs $ rm -rf node_modules]
```

```
[rodrigobranas fullstackjs $ yarn install]
```

```
yarn install v1.22.18
```

```
warning package-lock.json found. Your project contains lock files generated by tools other than Yarn. It is advised not to mix package managers in order to avoid resolution inconsistencies caused by unsynchronized lock files. To clear this warning, remove package-lock.json.
```

```
[1/4] 🔎 Resolving packages...
[2/4] 🚚 Fetching packages...
[3/4] ⚡ Linking dependencies...
[4/4] 🏺 Building fresh packages...
```

```
✨ Done in 1.01s.
```

```
rodrigobranas fullstackjs $
```

**yarn remove package\_name**

Utilizado para remover dependências, da forma  
forma que o npm uninstall



fullstackjs — zsh — 80x24

```
[rodrigobranas fullstackjs $ yarn remove typescript
```

```
yarn remove v1.22.18
```

```
[1/2] 🗑️ Removing module typescript...
```

```
[2/2] ✕ Regenerating lockfile and installing missing dependencies...
```

```
[-----] 0/650
```

```
success Uninstalled packages.
```

```
✨ Done in 0.43s.
```

```
rodrigobranas fullstackjs $
```

Command	npm	yarn
Install dependencies	<code>npm install</code>	<code>yarn</code>
Install package	<code>npm install [package]</code>	<code>yarn add [package]</code>
Install dev package	<code>npm install --save-dev [package]</code>	<code>yarn add --dev [package]</code>
Uninstall package	<code>npm uninstall [package]</code>	<code>yarn remove [package]</code>
Uninstall dev package	<code>npm uninstall --save-dev [package]</code>	<code>yarn remove [package]</code>
Update	<code>npm update</code>	<code>yarn upgrade</code>
Update package	<code>npm update [package]</code>	<code>yarn upgrade [package]</code>
Global install package	<code>npm install --global [package]</code>	<code>yarn global add [package]</code>
Global uninstall package	<code>npm uninstall --global [package]</code>	<code>yarn global remove [package]</code>