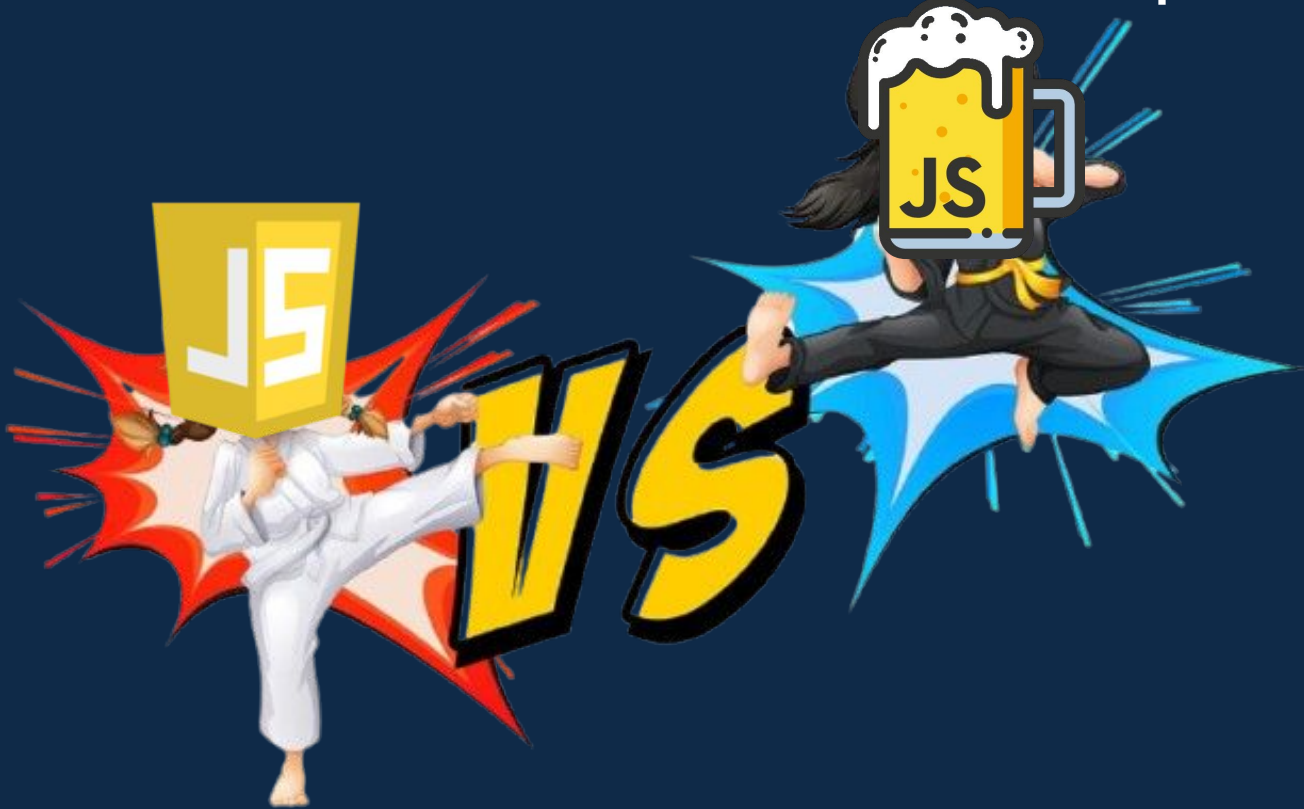


Artes Marciales con JavaScript



SERGIO GARZÓN

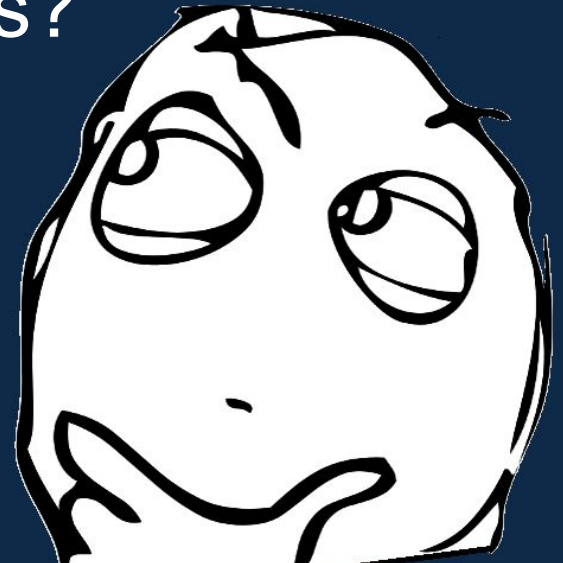
- Desarrollador de Software
- Desarrollador de Videojuegos
- Docente de Programación





Automatización de pruebas de API

¿Por qué el título de la charla es
“Artes Marciales con JavaScript”
si me estas mostrando una
herramienta para testear APIs?
No entiendo...





Karate Framework / Karate DSL



Karate Labs

Empresa de software
Sugar Land, Texas





**Karate
Labs**

Solutions ▾

Integrations

Plugins ▾

Resources ▾

Pricing

Why Karate ▾



Sign up

Open-Source Test Automation Platform


Unifying API testing, API Performance testing, API Mocks and UI testing.


GET STARTED

WATCH VIDEO


Sitio web: <https://www.karatelabs.io/>



 karatelabs

Q Type  to search

[Overview](#) [Repositories 13](#) [Projects](#) [Packages](#) [People 2](#)



Karate Labs
Simplicity is Key
Verified


[237 followers](#) <https://karatelabs.io> [company/karatelabs](#) [@karatelabs](#) [@getKarate](#)

[Sponsor](#) [Follow](#)

README .md







Since the first version of Karate was released seven years ago, we have successfully made testing fun and collaborative.

Our 8500 GitHub stars and adoption by the Global 2000 is clear indication that Karate is a top choice for developers, QA teams, and enterprise customers.



**Karate Labs**

Karate was created by a developer - keeping Developer Experience first and foremost. We made test-scripts simpler, more readable, and more maintainable by relying on an innovative DSL (Domain Specific Language), JSON support and JavaScript engine.

Staying true to our mission of making test automation fun & collaborative, we continue to invest in elevating the Developer Experience.

 Home	 How To	 Tools	 Enterprise	 Support	 Follow Us
Web Site Documentation	Get Started	Visual Studio Code	Integrations	Stack Overflow	LinkedIn
	Kick Start Videos	IntelliJ IDEA	GitHub Codespaces	Contact Us	Twitter
	Examples	Xplorer			YouTube

People



Top languages

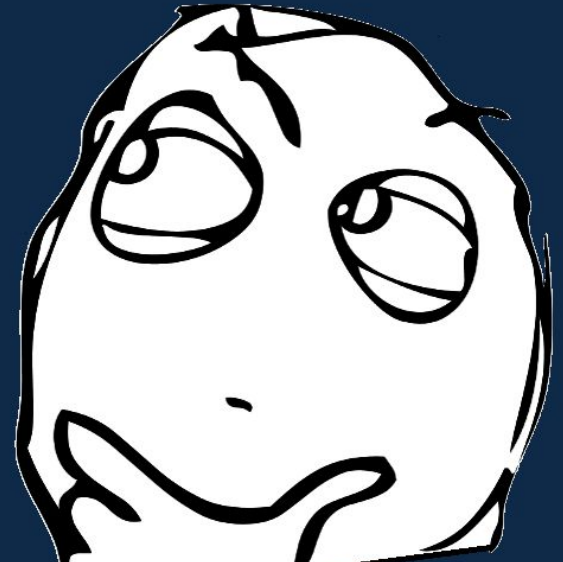
● Java ● Gherkin ● JavaScript ● HTML

[Report abuse](#)

Sitio web: <https://github.com/karatelabs>

 Interesante....

Pero ¿Para que me sirve?

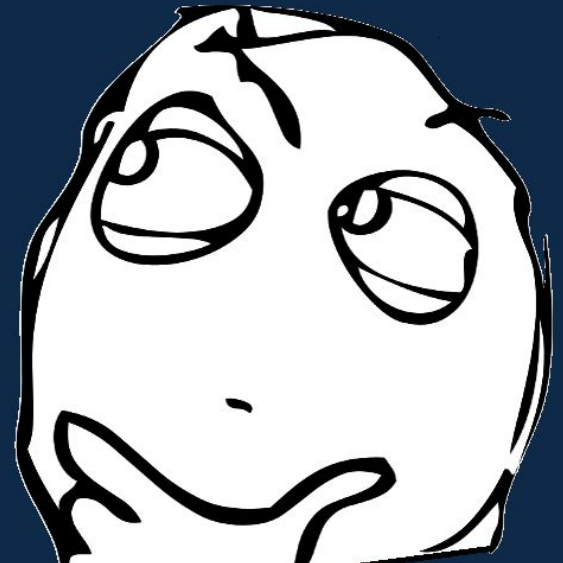


KARATE sirve para muchas cosas:

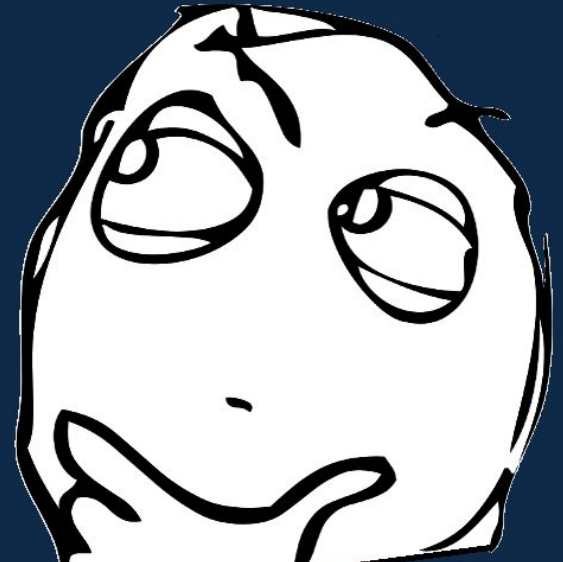
- Nos permite automatizar pruebas de APIs
- Crear script pruebas de APIs
- Simular APIs
- Realizar pruebas de UI
- Automatizar pruebas web, de performance y demás.



¿Y es fácil? 🤯



Obvio, no es difícil, es fácil
aprenderlo, pero puede volverse
complejo, depende del escenario.



KARATE utiliza:

- Utiliza Cucumber (Software para procesar y ejecutar pruebas de APIs)
- Cucumber utiliza el lenguaje Gherkin (Lenguaje DSL - Lenguaje de dominio específico)
- Cucumber se centra en la sintaxis BDD (Behavior Driven Development / Desarrollo dirigidos por comportamiento)



KARATE utiliza:

- Está escrito en el lenguaje de programación de Java
- Lo podemos utilizar en cualquier proyecto, no necesariamente de Java
- Es de código abierto, es gratuito, pero en su mayoría



Y lo más importante de todo:

- Karate tiene funciones internas, escritas en JavaScript, porque cuenta con el motor Karate-JS, que está basado en el motor GraalVM.
- Se pueden crear más funciones de JavaScript para realizar configuraciones. Dentro del archivo `.feature` o en un archivo `.js`



Repo del motor Karate-JS

The screenshot shows the GitHub repository page for **karatelabs / karate-js**. The repository is public and has 21 stars, 7 forks, and 3 watchers. The main branch is **main**, with 1 branch and 2 tags. The repository was last updated 5 months ago by **pvrthomas**, who has made 120 commits. The file list includes **.github/workflows**, **karate-js**, **test-core**, **test-ecma**, **test-perf**, **.gitignore**, **LICENSE**, **README.md**, and **pom.xml**. The **README.md** file is selected, showing the **maven-central v0.1.1** badge. The right sidebar contains sections for **About** (Lightweight JavaScript engine for the JVM), **Releases** (2 tags), **Packages** (No packages published), and **Languages** (Java 97.0%, Lex 2.6%, JavaScript 0.4%).

karatelabs / karate-js

Public

Watch 3 Fork 7 Star 21

main 1 Branch 2 Tags

Go to file Add file Code

pvrthomas support dollar symbol within backtick / string templates 4d7cfeb · 5 months ago 120 Commits

.github/workflows	public release	last year
karate-js	support dollar symbol within backtick / string templates	5 months ago
test-core	support dollar symbol within backtick / string templates	5 months ago
test-ecma	add version to maven deploy plugin	last year
test-perf	add version to maven deploy plugin	last year
.gitignore	public release	last year
LICENSE	Initial commit	last year
README.md	string regex match, search and replace implemented	7 months ago
pom.xml	add version to maven deploy plugin	last year

README MIT license

maven-central v0.1.1

About

Lightweight JavaScript engine for the JVM

- Readme
- MIT license
- Activity
- Custom properties
- 21 stars
- 3 watching
- 7 forks

Report repository

Releases

2 tags

Packages

No packages published

Languages

- Java 97.0%
- Lex 2.6%
- JavaScript 0.4%

Sitio web: <https://github.com/karatelabs/karate-js>

Se puede integrar con varias tecnologías:



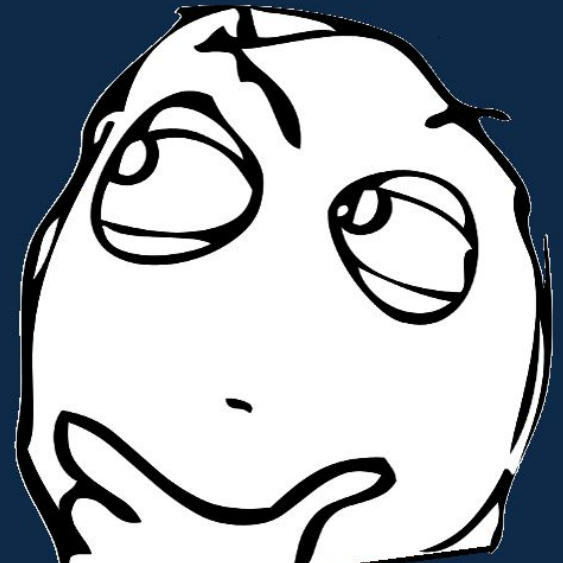
Sitio web: <https://www.karatelabs.io/integrations>

Karate está escrito en Java y está relacionado con la Java Virtual Machine

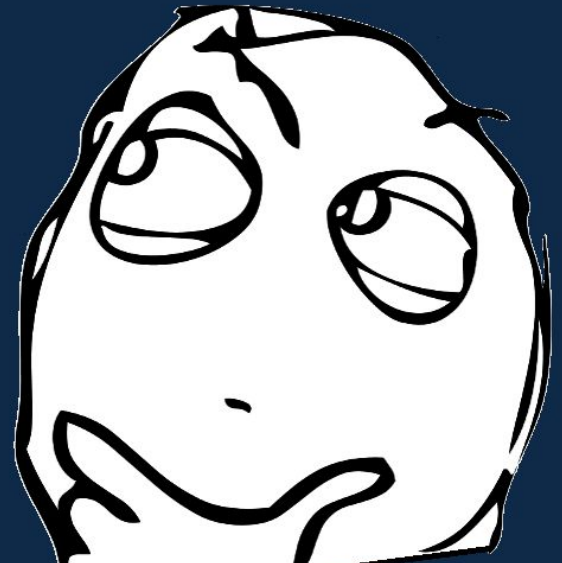




Interesante todo lo que me contas
¿Y tenes alguna demostración para
ver?



Si, mas vale papa. 😊



Si tenemos un proyecto de Java con Maven o Gradle es posible, de hecho mucho mejor porque Karate DSL está escrito en java




Teniendo el JDK instalado:



En el repo nos da el comando para generar un proyecto de Maven

```
mvn archetype:generate \  
-DarchetypeGroupId=io.karatelabs \  
-DarchetypeArtifactId=karate-archetype \  
-DarchetypeVersion=1.5.1 \  
-DgroupId=com.mycompany \  
-DartifactId=myproject
```

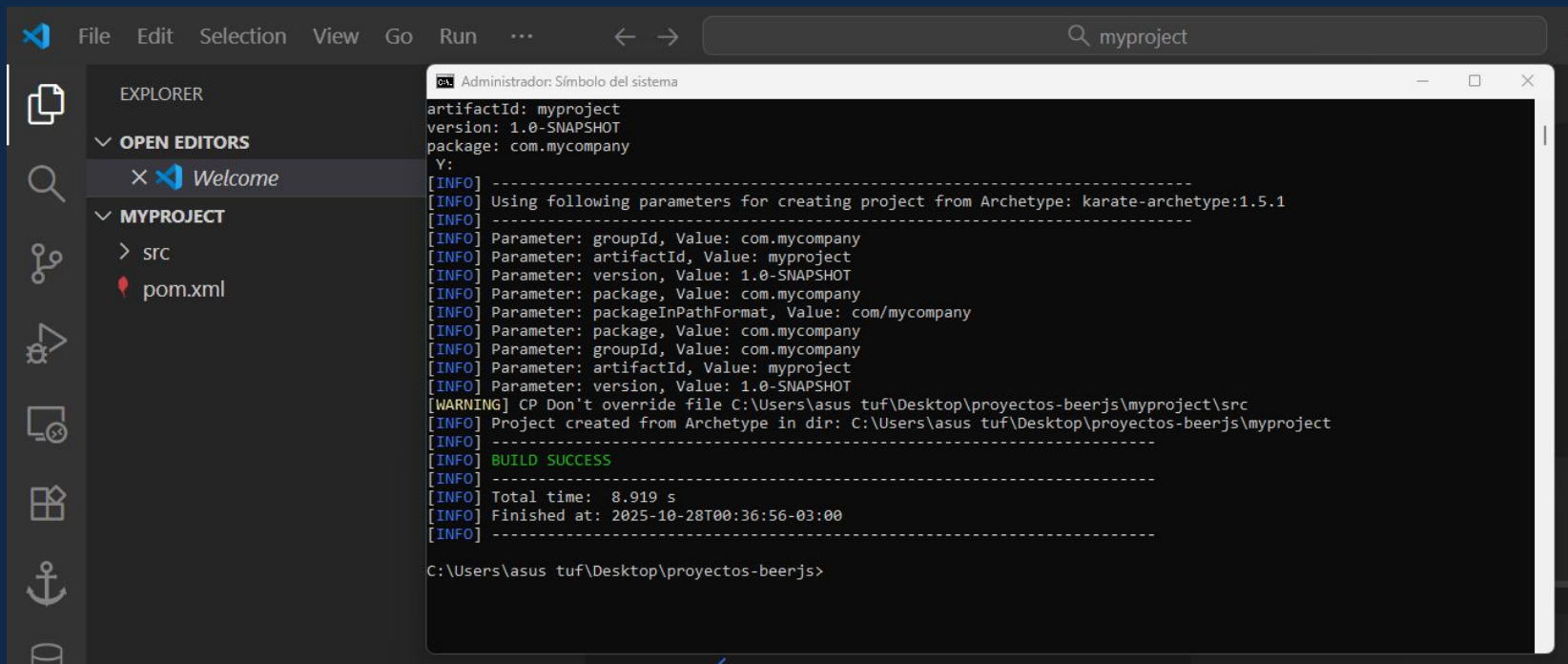


También con Gradle:

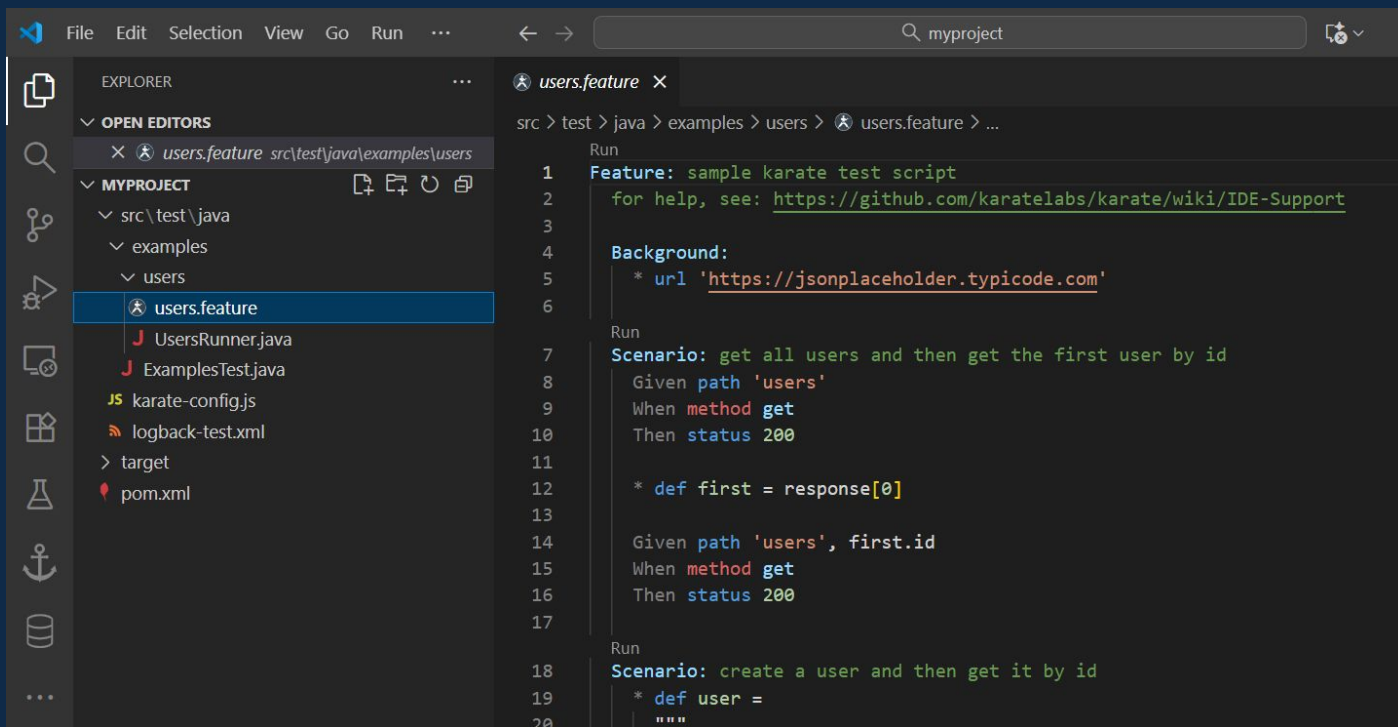
```
testImplementation 'io.karatelabs:karate-junit5:1.5.1'
```



Creamos el proyecto de java con Maven



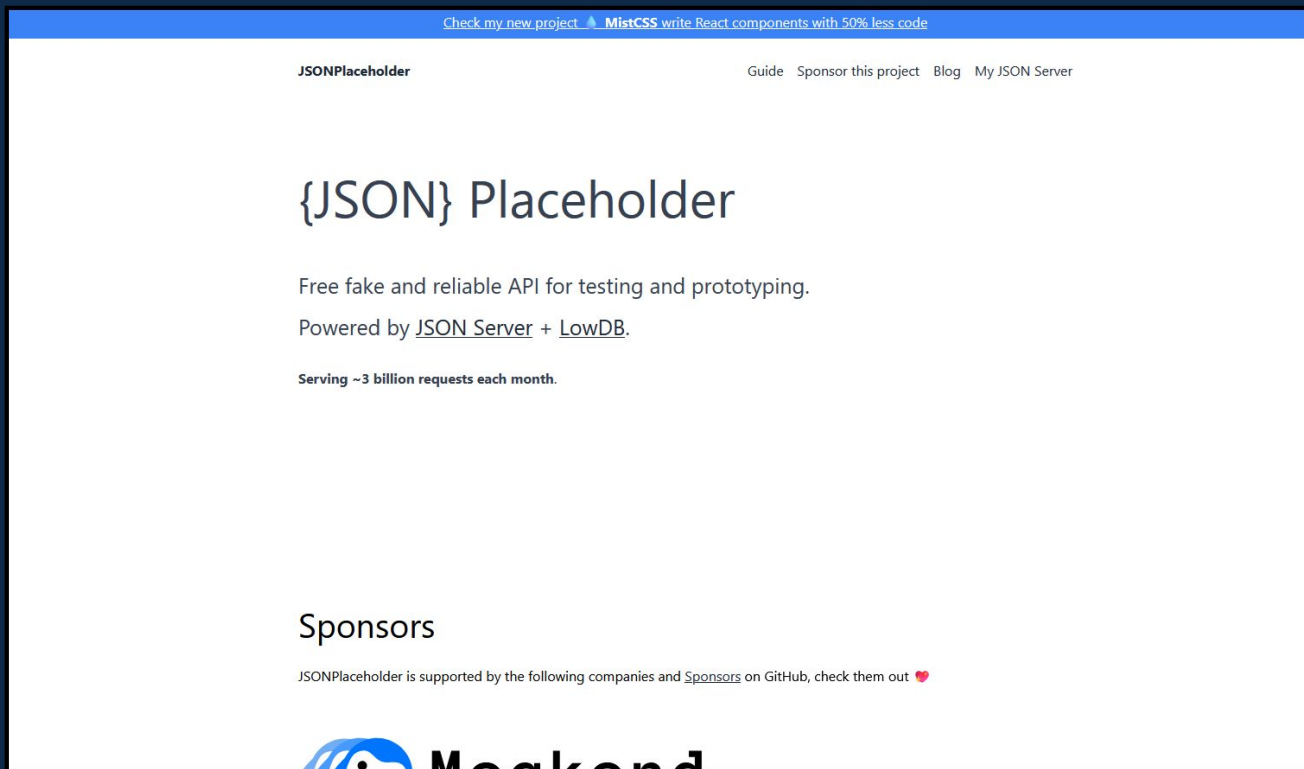
En la estructura del proyecto tenemos los escenarios (archivos .feature)



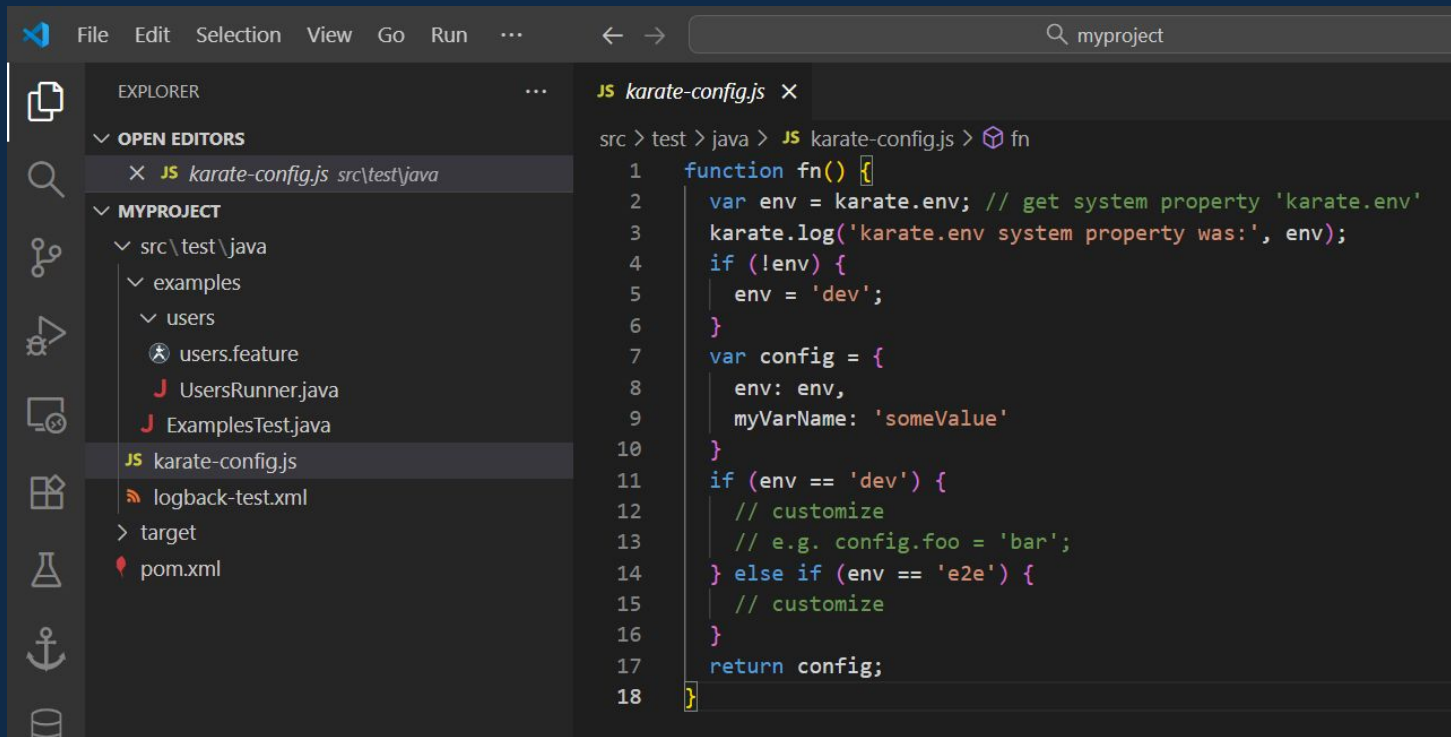
The screenshot shows an IDE interface with a dark theme. On the left, the 'EXPLORER' sidebar displays the project structure under 'MYPROJECT'. The path is 'src\test\java\examples\users', and the file 'users.feature' is selected. Below it, other files like 'UsersRunner.java', 'ExamplesTest.java', 'karate-config.js', 'logback-test.xml', 'target', and 'pom.xml' are listed. The main editor area shows the content of 'users.feature'. The file path at the top of the editor is 'src > test > java > examples > users > users.feature > ...'. The code in the editor is as follows:

```
Run
1 Feature: sample karate test script
2   for help, see: https://github.com/karatelabs/karate/wiki/IDE-Support
3
4 Background:
5   * url 'https://jsonplaceholder.typicode.com'
6
7 Run
8 Scenario: get all users and then get the first user by id
9   Given path 'users'
10  When method get
11  Then status 200
12
13   * def first = response[0]
14
15  Given path 'users', first.id
16  When method get
17  Then status 200
18
19 Run
20 Scenario: create a user and then get it by id
21   * def user =
22     """
```


Ya nos trae por defecto una url de una API (users) de JSONPlaceholder



y el archivo de configuración en JavaScript que usa Karate
(archivo karate-config.js)



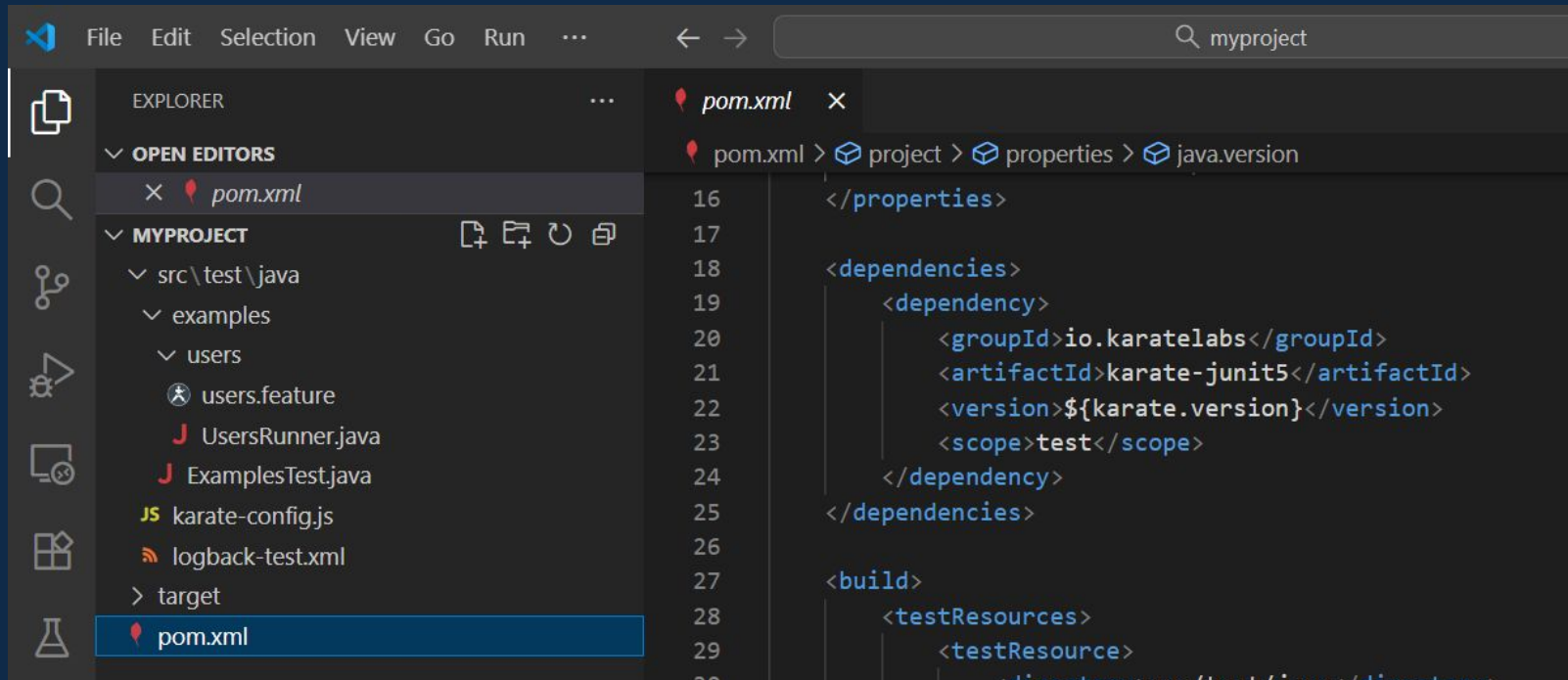
The screenshot shows the Visual Studio Code interface with a project named 'myproject'. The Explorer sidebar on the left displays the file structure:

- EXPLORER
 - OPEN EDITORS
 - JS karate-config.js src\test\java
 - MYPROJECT
 - src\test\java
 - examples
 - users
 - users.feature
 - UsersRunner.java
 - ExamplesTest.java
 - JS karate-config.js**
 - logback-test.xml
 - target
 - pom.xml

The main editor window shows the content of 'JS karate-config.js' with the following code:

```
src > test > java > JS karate-config.js > fn
1 function fn() {
2     var env = karate.env; // get system property 'karate.env'
3     karate.log('karate.env system property was:', env);
4     if (!env) {
5         env = 'dev';
6     }
7     var config = {
8         env: env,
9         myVarName: 'someValue'
10    }
11    if (env == 'dev') {
12        // customize
13        // e.g. config.foo = 'bar';
14    } else if (env == 'e2e') {
15        // customize
16    }
17    return config;
18 }
```

En el archivo pom.xml podemos ver que Karate Framework / DSL se integra con JUnit para hacer test de pruebas



The screenshot shows an IDE interface with the following components:

- Explorer (Left):** Displays the project structure. The 'pom.xml' file is selected at the bottom of the list. The project is named 'MYPROJECT'.
- Open Editors (Top Left):** Shows 'pom.xml' as the currently open file.
- Main Editor (Right):** Displays the content of 'pom.xml'. The breadcrumb navigation at the top of the editor indicates the path: 'pom.xml > project > properties > java.version'. The XML content is as follows:

```
16 </properties>
17
18 <dependencies>
19   <dependency>
20     <groupId>io.karatelabs</groupId>
21     <artifactId>karate-junit5</artifactId>
22     <version>${karate.version}</version>
23     <scope>test</scope>
24   </dependency>
25 </dependencies>
26
27 <build>
28   <testResources>
29     <testResource>
30       <directory>src/test/feature</directory>
```

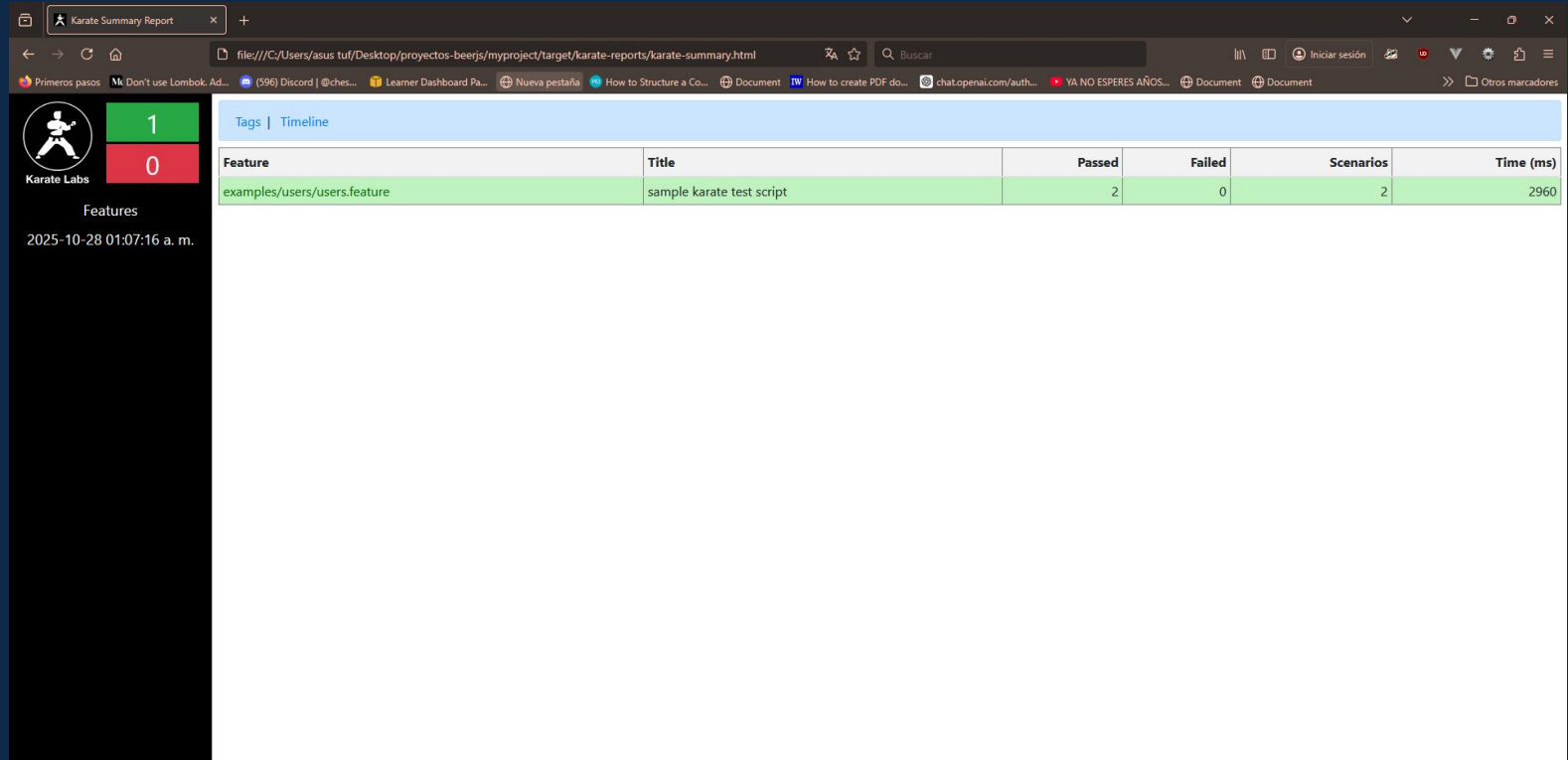
Para instalar las dependencias de Karate, junto con las otras, entonces ejecuto el comando de Maven

```
PS C:\Users\asus tuf\Desktop\proyectos-beerjs\myproject> mvn compile
[INFO] Scanning for projects...
[INFO]
[INFO] -----< com.mycompany:myproject >-----
[INFO] Building myproject 1.0-SNAPSHOT
```

Luego ejecuto el test con Maven

```
PS C:\Users\asus tuf\Desktop\proyectos-beerjs\myproject> mvn clean test
[INFO] Scanning for projects...
[INFO]
[INFO] -----< com.mycompany:myproject >-----
[INFO] Building myproject 1.0-SNAPSHOT
[INFO] from pom.xml
```

En la terminal se genera el log, y a su vez nos da el reporte de las pruebas que hizo



Karate Summary Report

file:///C:/Users/asus/tuf/Desktop/proyectos-beerjs/myproject/target/karate-reports/karate-summary.html

Tags | Timeline

Feature	Title	Passed	Failed	Scenarios	Time (ms)
examples/users/users.feature	sample karate test script	2	0	2	2960

Karate Labs

Features

2025-10-28 01:07:16 a. m.

Dentro de los archivos .feature de Karate, tenemos sintaxis de Cucumber

Run

Feature: Todo lo que contiene el test

Background: Precondiciones (url, token, variables de cada escenario y demas...)

Run

Scenario: Nombre del escenario

Given

#Dado

And

Y

When

Cuando

Then

Entonces

Run

Scenario: Nombre del escenario 2

Multiples escenarios

Dentro de los archivos .feature de Karate, es posible escribir JSON dentro de una variable:

```
Scenario: Escenario de eventos
  * def createNewEvent =
    """
    {
      "nombreEvento": "BeerJS",
      "tematica": "JavaScript"
    }
    """

  Given path '/api/events'
  And request createNewEvent
  When method post
  Then status 200
```

Dentro de los archivos .feature de Karate, es posible escribir funciones de javascript:

```
* def calcularSuma =  
  """  
    function(num1, num2){  
      var result = num1 + num2  
      return result  
    }  
  """  
* karate.log(calcularSuma(12, 2))
```

No se puede utilizar console.log() de javascript, si no que karate.log()

También es posible utilizar simplemente “print” para mostrar por consola valores:

```
* def calcularSuma =  
  """  
  |  
  | function(num1, num2){  
  |   var result = num1 + num2  
  |   return result  
  | }  
  |  
  | """  
* print "El resultado de la suma es: ", calcularSuma(12, 2)
```

Para poder ejecutar los test, se trata en conjunto la JVM y el motor de Karate-JS, por medio de esto se generan los reportes



Karate-JS

Ahora, sin la necesidad de que sea un proyecto de Java, y tener Maven o Gradle instalado, viene una extensión de VSCode:

Extension: Karate ✕



Karate v2.0.1

Karate Labs

16,545

★★★★☆ (5)

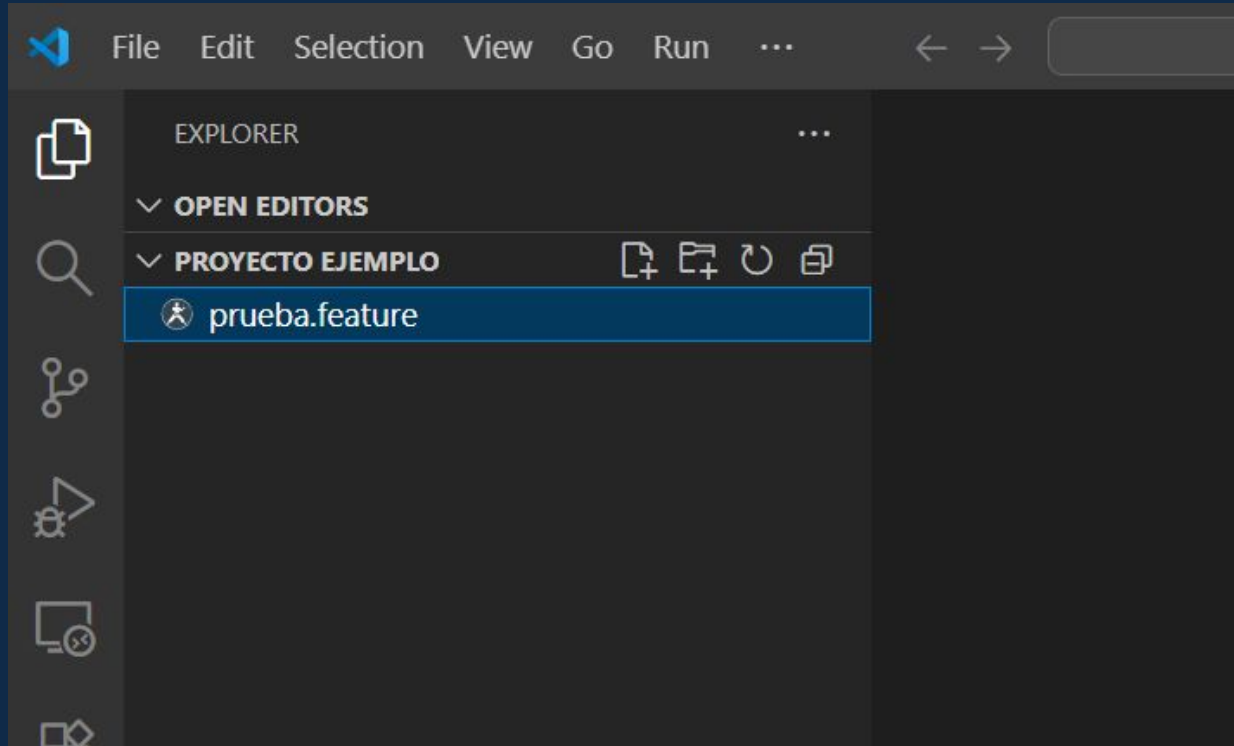
♥ Sponsor

Test Automation Made Simple

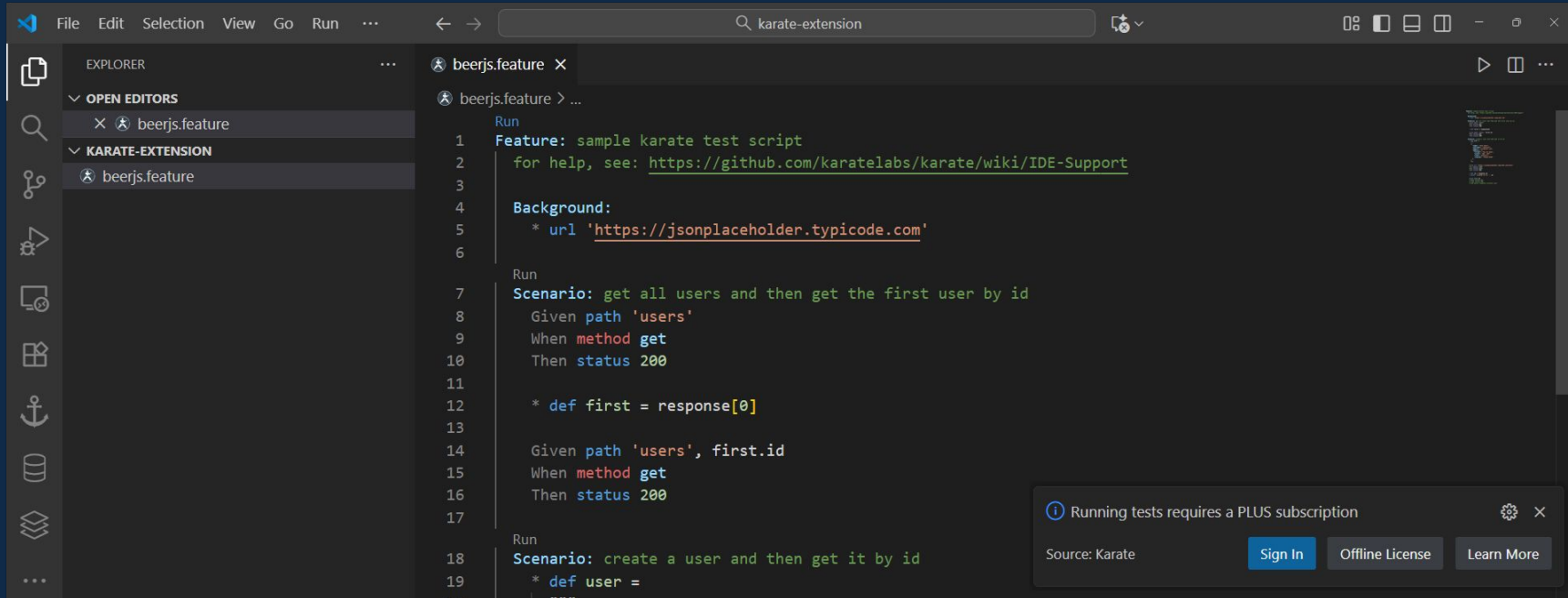
Install



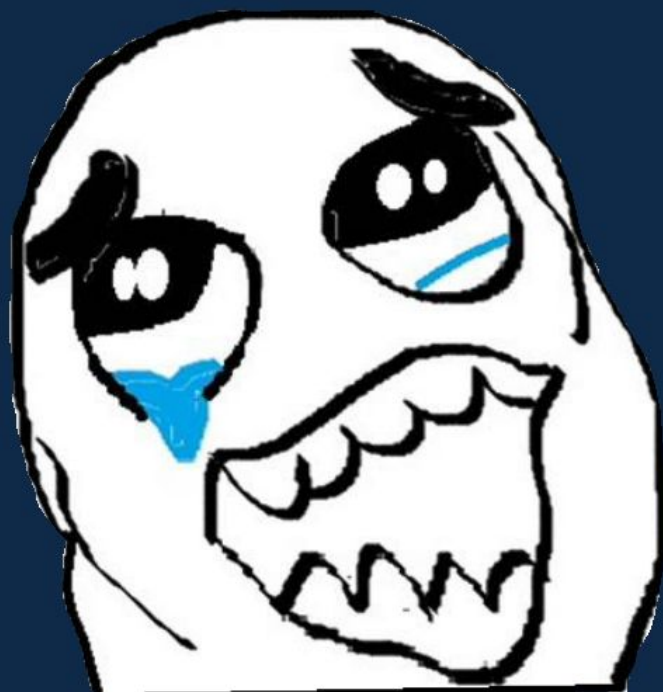
Se puede ejecutar directamente archivos de karate .feature y agregarle código



Se puede ejecutar directamente archivos de karate .feature desde el botón de Run



Pero el tema es que esta característica no la podemos utilizar gratuitamente, la han hecho paga hace 2 años atras (2023).



Otra opción para trabajar con Karate en proyectos de javascript
con NodeJS con Karate-npm



Repo del proyecto Karate-npm

karatelabs / karate-npm

Code Issues 1 Pull requests Actions Projects Security Insights

karate-npm Public

Watch 1 Fork 6 Star 12

main 1 Branch 0 Tags

Go to file Add file Code

ptrthomas	update readme for higher versions	32ac241 · last year	35 Commits
karate	test that #3 is resolved	3 years ago	
.gitignore	refactor for using jbang-npm	3 years ago	
.npmignore	refactor for using jbang-npm	3 years ago	
README.md	update readme for higher versions	last year	
karate.js	release for karate 1.5.0	last year	
package.json	release for karate 1.5.0	last year	
preinstall.js	refactor for using jbang-npm	3 years ago	
test.js	release for karate 1.5.0	last year	

README

@karatelabs/karate

Seamlessly use the power of [Karate](#) from Node / JS projects.

Usage

About

NPM for Karate

[www.npmjs.com/package/@karatelabs/k...](#)

- Readme
- Activity
- Custom properties
- 12 stars
- 1 watching
- 6 forks

Report repository

Releases

No releases published

Packages

No packages published

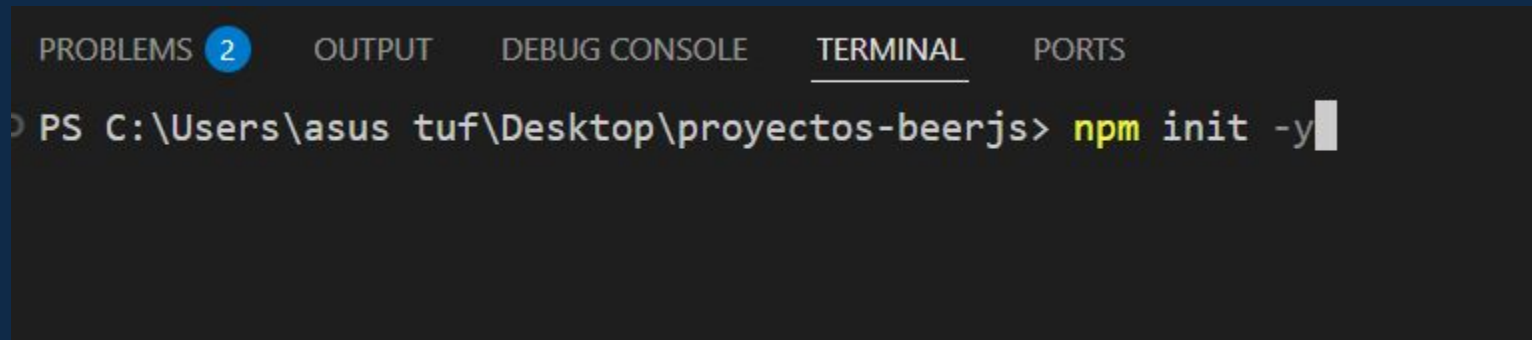
Contributors 2

- ptrthomas Peter Thomas
- steve1337

Sitio web: <https://github.com/karatelabs/karate-npm>

Empezamos a trabajar

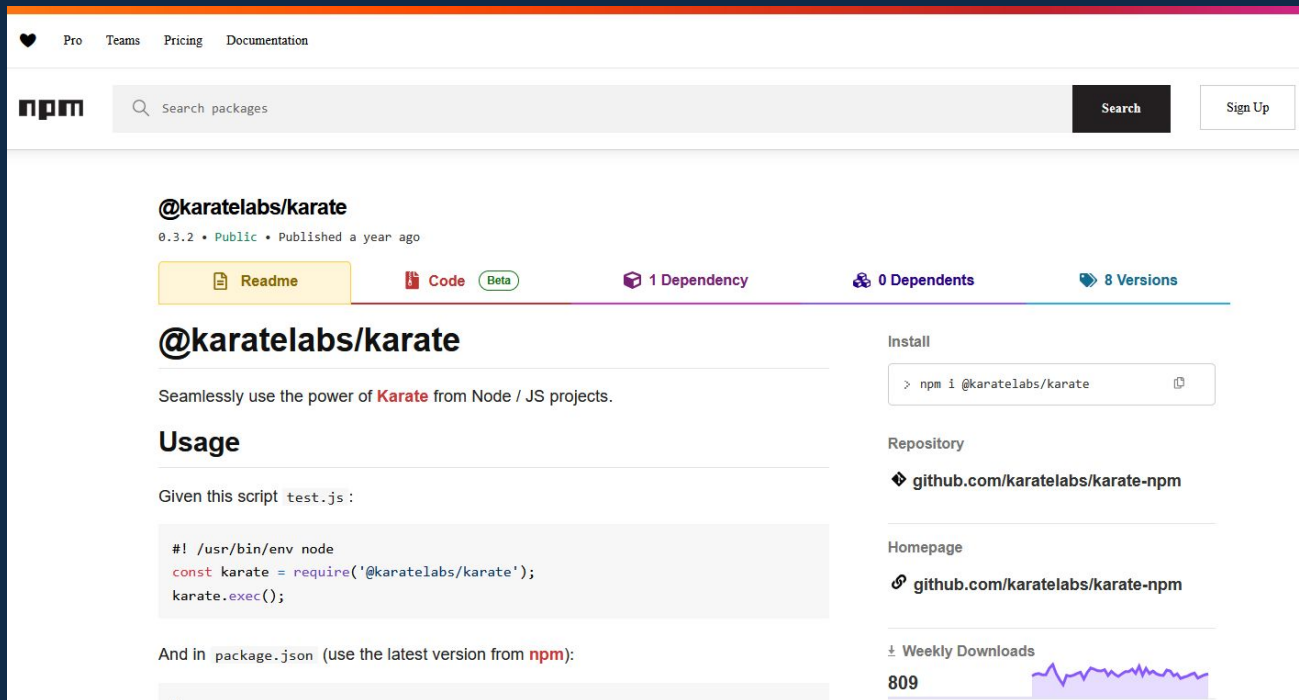
Creamos un proyecto de node



A screenshot of a Visual Studio Code terminal window. The terminal has a dark background with a light blue title bar. The title bar contains five tabs: 'PROBLEMS' with a blue circle containing the number '2', 'OUTPUT', 'DEBUG CONSOLE', 'TERMINAL' (which is selected and underlined), and 'PORTS'. The terminal content shows a PowerShell prompt 'PS' followed by the path 'C:\Users\asus tuf\Desktop\proyectos-beerjs>'. The command 'npm init -y' is being typed, with 'npm' in yellow and 'init -y' in white. A white cursor is at the end of the command.

```
PS C:\Users\asus tuf\Desktop\proyectos-beerjs> npm init -y
```

Primero debemos instalar la dependencia de Karate



The screenshot shows the npm package page for **@karatelabs/karate**. The page header includes navigation links: Pro, Teams, Pricing, and Documentation. Below the header is a search bar with the text "Search packages" and a "Search" button. The package name **@karatelabs/karate** is prominently displayed, along with its version **0.3.2**, status **Public**, and publication date **Published a year ago**. A horizontal bar contains links for **Readme**, **Code** (with a **Beta** badge), **1 Dependency**, **0 Dependents**, and **8 Versions**. The main content area is divided into two columns. The left column contains the package description: "Seamlessly use the power of **Karate** from Node / JS projects." followed by a **Usage** section. The usage section shows a sample script `test.js` and a sample `package.json` entry. The right column contains an **Install** section with a terminal command `> npm i @karatelabs/karate`, a **Repository** section with a link to `github.com/karatelabs/karate-npm`, a **Homepage** section with a link to `github.com/karatelabs/karate-npm`, and a **Weekly Downloads** section showing **809** downloads with a corresponding line graph.

Pro Teams Pricing Documentation

npm Search packages Search Sign Up

@karatelabs/karate
0.3.2 • Public • Published a year ago

Readme Code Beta 1 Dependency 0 Dependents 8 Versions

@karatelabs/karate

Seamlessly use the power of **Karate** from Node / JS projects.

Usage

Given this script `test.js`:

```
#!/usr/bin/env node
const karate = require('@karatelabs/karate');
karate.exec();
```

And in `package.json` (use the latest version from **npm**):

Install

```
> npm i @karatelabs/karate
```

Repository

github.com/karatelabs/karate-npm

Homepage

github.com/karatelabs/karate-npm

Weekly Downloads

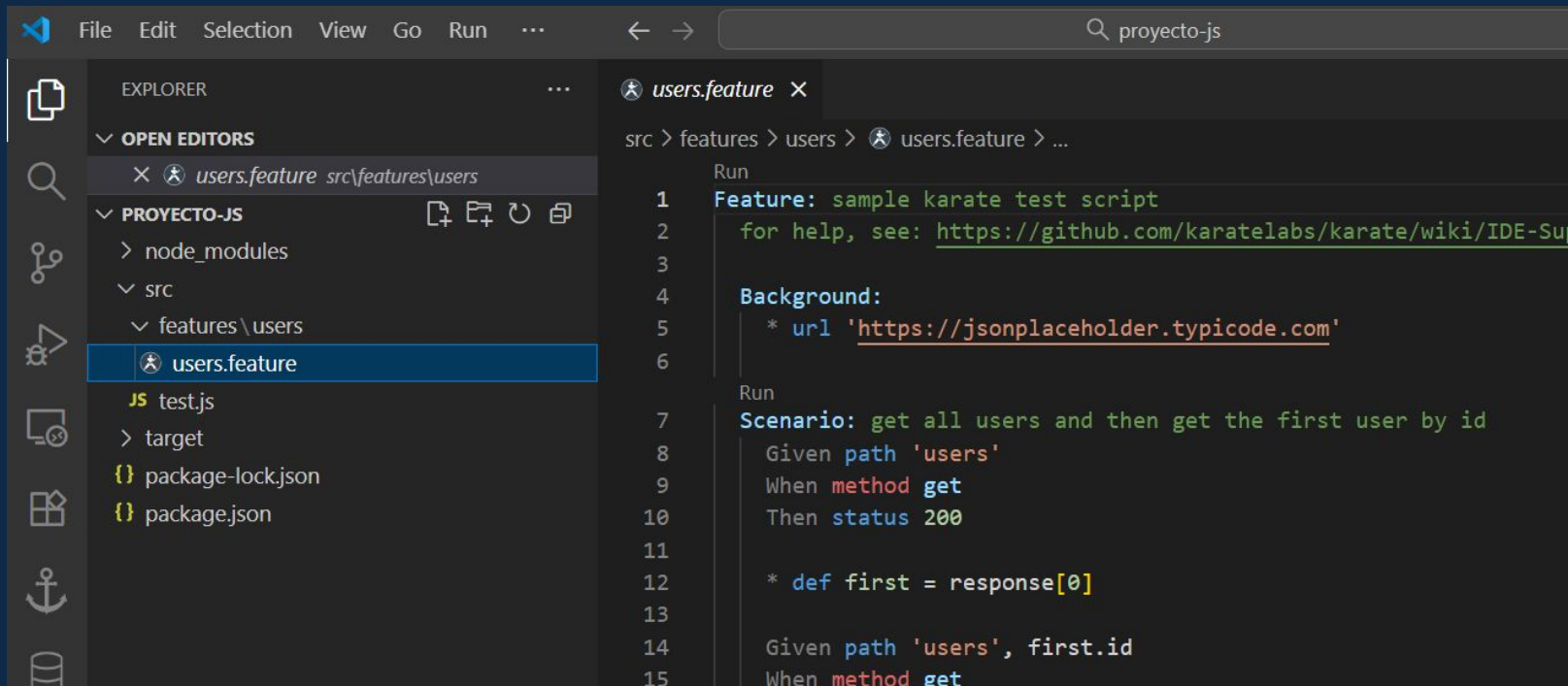
809

Link: <https://www.npmjs.com/package/@karatelabs/karate>

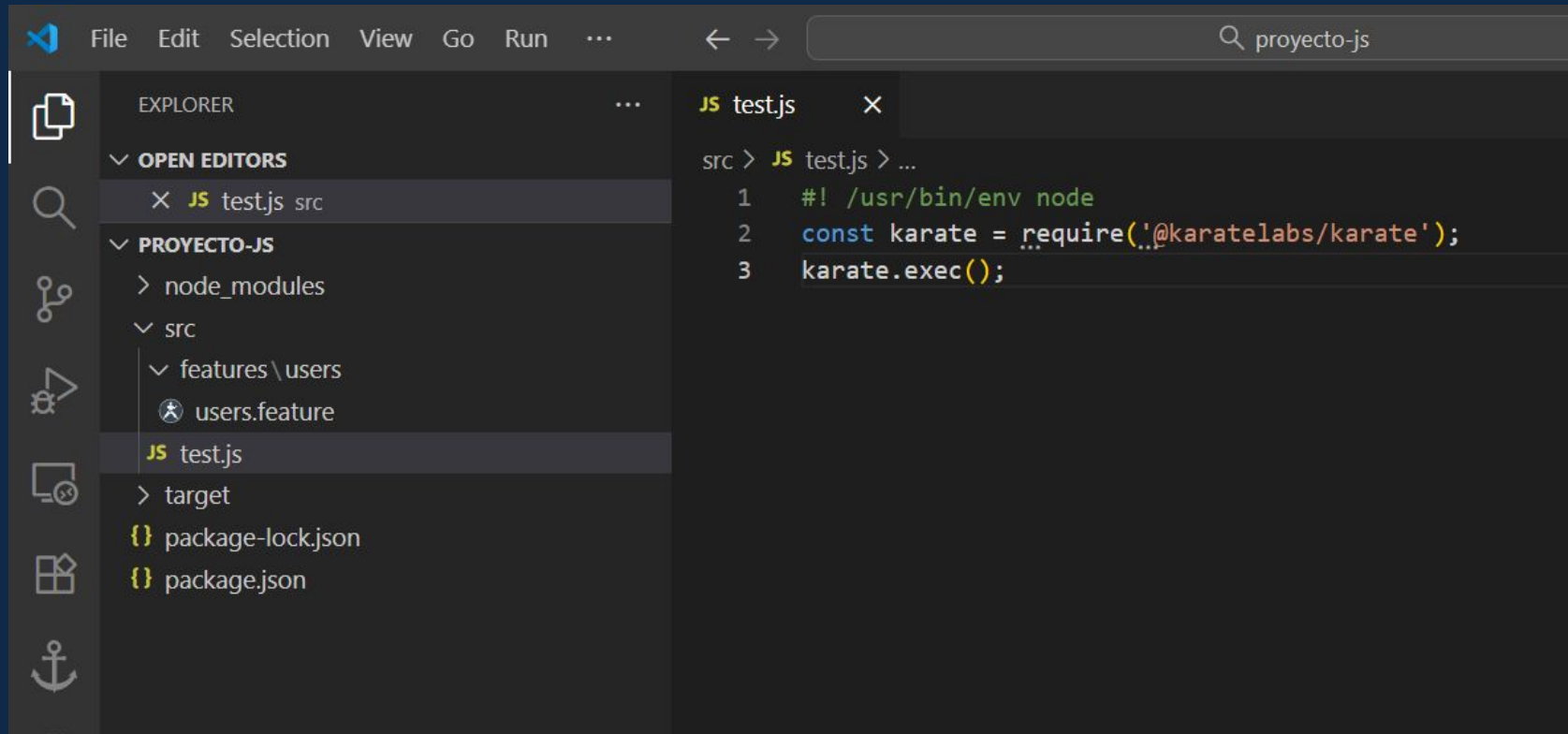
Se guarda en el package.json la dependencia de Karate

```
{ } package.json X
{ } package.json > ...
1  {
2    "name": "proyecto",
3    "version": "1.0.0",
4    "description": "",
5    "main": "index.js",
6    "scripts": {
7      "test": "node ./src/test.js"
8    },
9    "keywords": [],
10   "author": "",
11   "license": "ISC",
12   "dependencies": {
13     "@karatelabs/karate": "^0.3.2"
14   }
15 }
16
```

Agregamos un archivo de test de karate (.feature)



Agregamos un archivo de test de javascript



Para ejecutar los test de Karate, hay que instalar una dependencia que es @jbangdev/jbang que es un orquestador de JVM.



Dos formas de instalar:

- Consultar documentación
- Consultar la dependencia en la página de npm

JBang

Lets Students, Educators and Professional Developers create, edit and run self-contained source-only Java programs with unprecedented ease.

[Download](#)[Try](#)

LATEST BLOG POST

Java in Places You Do Not Expect It

October 17, 2025

```
//DEPS org.slf4j:slf4j-simple:1.7.25
//DEPS org.kohsuke.github-api:1.116
import org.kohsuke.github.*;
```

Embedded Dependencies

Automatic fetching of any dependency using `//DEPS` annotations directly from the

```
$ sdk install jbang
$ brew install jbangdev/tap/jbang
C:/> choco install jbang
```

Install & Run Anywhere

JBang installs and run on Windows, Linux, macOS, Docker and Github Actions as well as usable from Maven and Gradle plugins

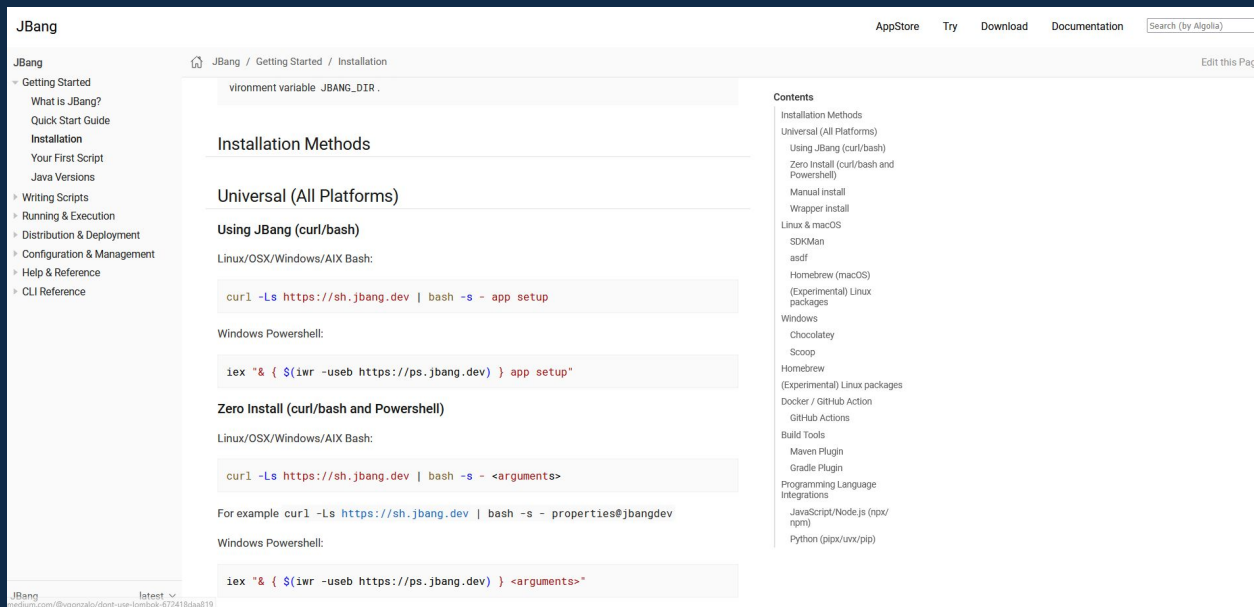
```
Downloading JBang...
Installing JBang...
Downloading JDK 11. Be Patient, ...
```

No Java ? No Problem!

Java will automatically be downloaded when needed.

Sitio web: <https://www.jbang.dev/>

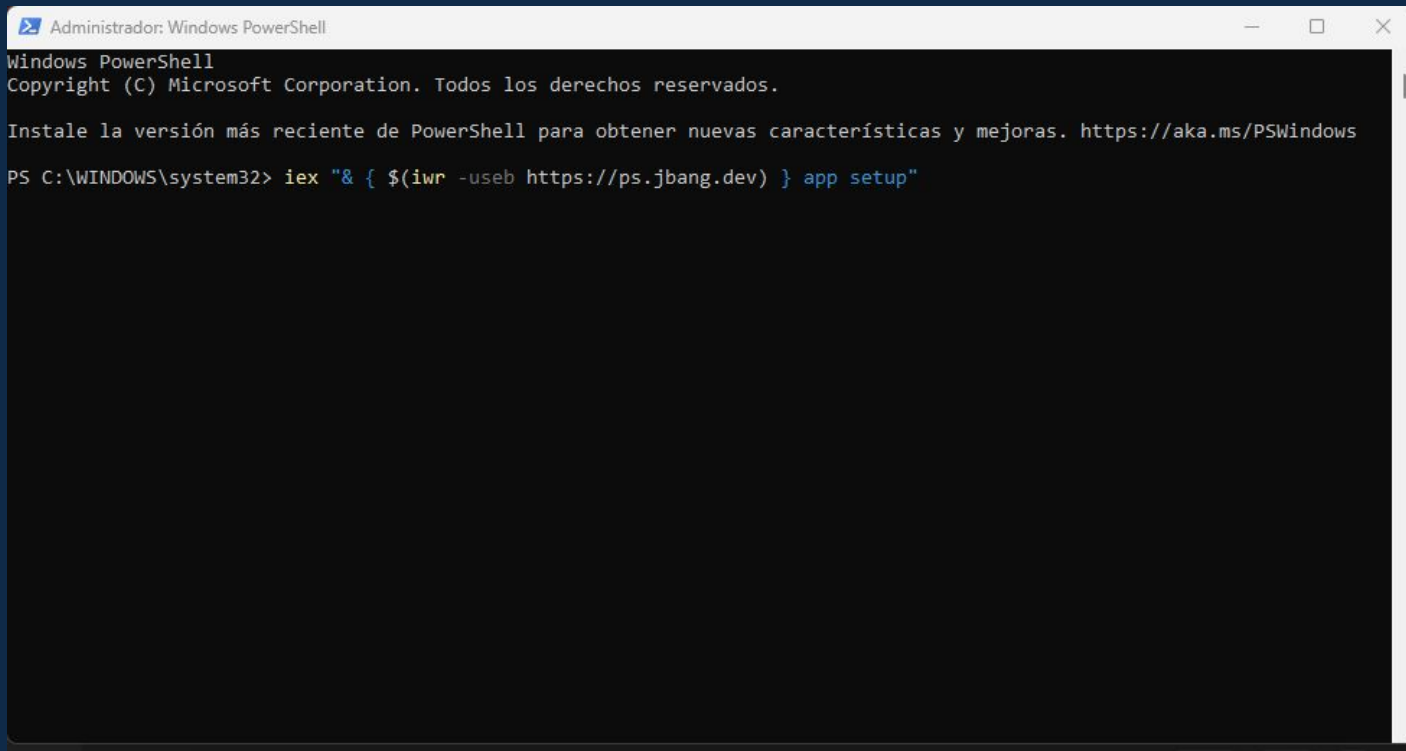
En la documentación nos salen los comandos para instalar JBang en terminal de Linux, PowerShell de Windows y demás



Link:

<https://www.jbang.dev/documentation/jbang/latest/installation.html>

En el PowerShell de Windows instalamos JBang

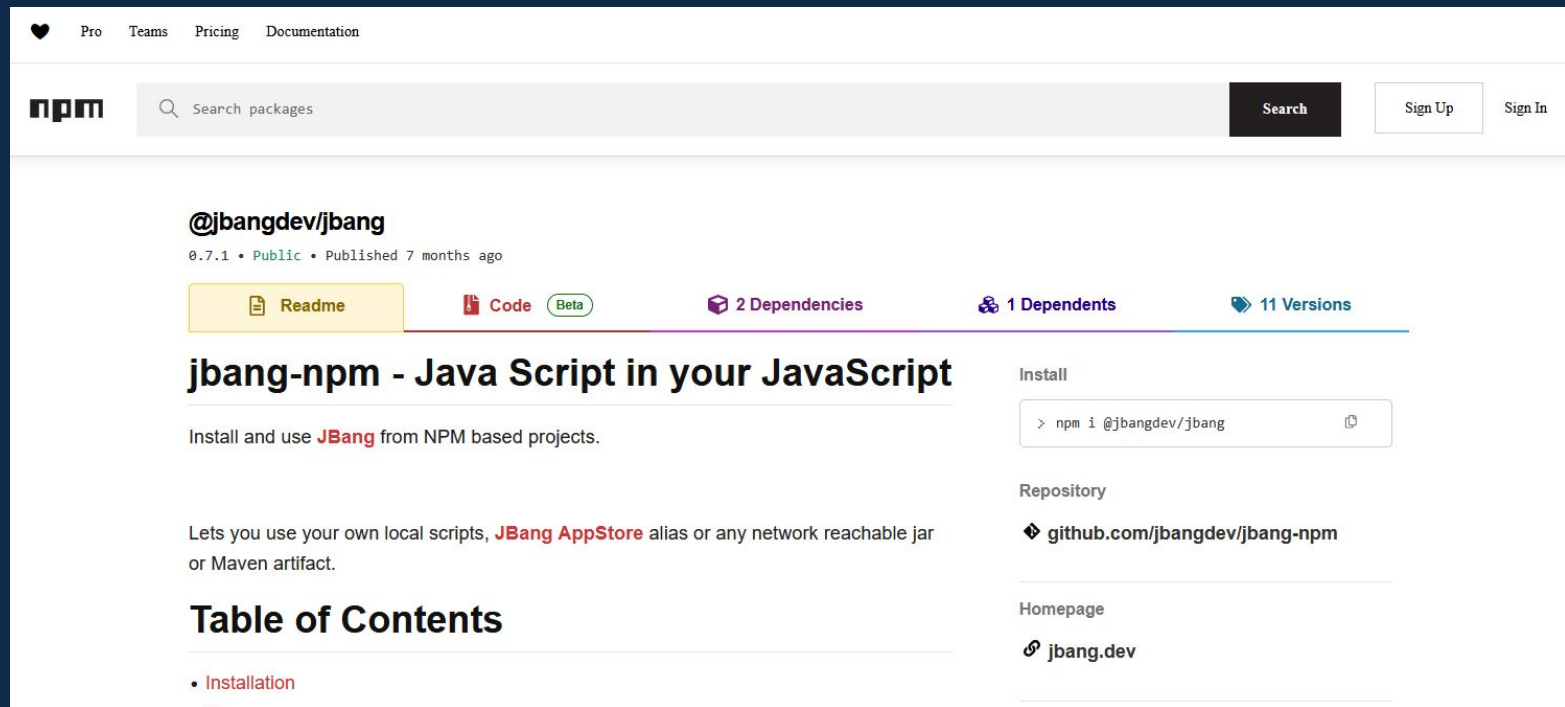


```
Administrador: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Instale la versión más reciente de PowerShell para obtener nuevas características y mejoras. https://aka.ms/PSWindows

PS C:\WINDOWS\system32> iex "& { $(iwr -useb https://ps.jbang.dev) } app setup"
```

Otra opción para instalar es desde la página de npm



The screenshot shows the npm package page for `@jbangdev/jbang`. The page header includes navigation links: Pro, Teams, Pricing, and Documentation. The main header features the npm logo, a search bar with the placeholder text "Search packages", and buttons for "Search", "Sign Up", and "Sign In".

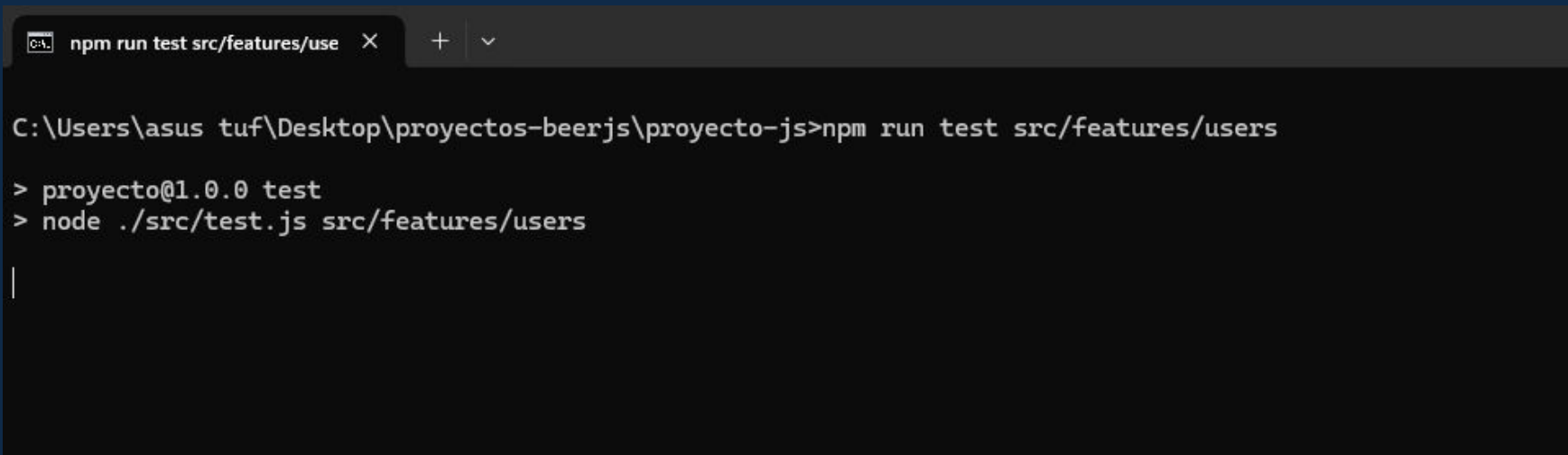
The package details section for `@jbangdev/jbang` shows version `0.7.1`, which is `Public` and was published `7 months ago`. It has `2 Dependencies`, `1 Dependents`, and `11 Versions`. The "Readme" tab is selected, showing the package title `jbang-npm - Java Script in your JavaScript` and a description: "Install and use **JBang** from NPM based projects."

The "Install" section shows the command `> npm i @jbangdev/jbang`. The "Repository" section lists `github.com/jbangdev/jbang-npm`. The "Homepage" section lists `jbang.dev`.

The "Table of Contents" section includes a link to the "Installation" section.

Link: <https://www.npmjs.com/package/@jbangdev/jbang>

Por último ejecutamos el test desde la terminal de Linux o Windows

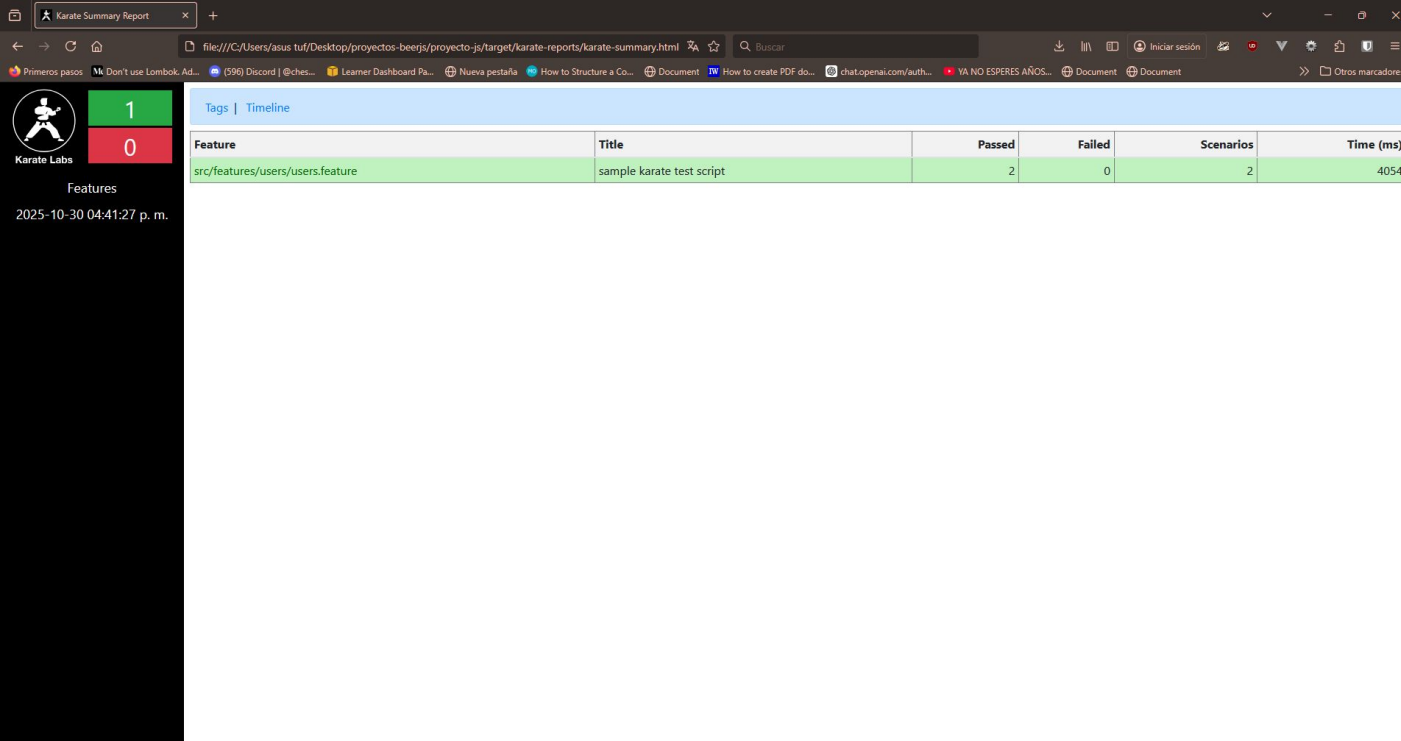
A screenshot of a Windows command prompt terminal window. The title bar shows a single tab with the text "npm run test src/features/use" and a close button. The terminal content shows the command "C:\Users\asus tuf\Desktop\proyectos-beerjs\proyecto-js>npm run test src/features/users" being entered. Below this, the output of the command is displayed: "> proyecto@1.0.0 test" followed by "> node ./src/test.js src/features/users". A vertical cursor is visible on the line following the second command.

```
C:\Users\asus tuf\Desktop\proyectos-beerjs\proyecto-js>npm run test src/features/users

> proyecto@1.0.0 test
> node ./src/test.js src/features/users

|
```

Podremos ver el log en la terminal y genera el reporte que es igual a Karate Framework



The screenshot shows a web browser displaying a Karate Summary Report. The browser's address bar shows the file path: `file:///C:/Users/asus.tuf/Desktop/proyectos-beerjs/proyecto-js/target/karate-reports/karate-summary.html`. The report is titled "Karate Summary Report" and includes a sidebar with the Karate Labs logo, a "Features" section, and a timestamp: "2025-10-30 04:41:27 p. m.". The main content area displays a table with the following data:

Feature	Title	Passed	Failed	Scenarios	Time (ms)
src/features/users/users.feature	sample karate test script	2	0	2	4054

Puedo utilizar otras API para realizar pruebas que no sea el genérico de JSONPlaceholder



{ api }

Puedo poner las API que yo quiera, pero tengo esta pagina que tiene muchísimas APIs para realizar pruebas, es un mercado de APIs:



RapidAPI: Sitio web para consultar APIs

The screenshot shows the RapidAPI website homepage. At the top, there's a navigation bar with the RapidAPI logo, links for 'Enterprises' and 'About Us', and a user profile icon. The main hero section features the headline 'Your dream API is on the market' and a subtext explaining that the platform is previously known as Rapid and Nokia API Hub. It includes a blue 'API hub' button. Below this, it states 'World's largest public API Hub'. Three statistics are displayed: '7M+ Developers', '82K+ APIs in the Hub', and '9B+ API calls per month'. The bottom section is titled 'Consume APIs' and includes a code snippet for using Axios in Node.js. To the right of the code, there are two API cards: 'Change Memo' and 'Raw Materials Schedule'. The bottom right corner has a small 'Powered by Cloudinary' logo.

Rapid
Nokia API Hub

Enterprises About Us

API Hub

Your dream API is on the market

Previously known as Rapid, Nokia API Hub is more than just a catalog for APIs. Find the perfect API using our discovery features or publish your own APIs and drive new API revenue streams.

API hub

**World's largest public
API Hub**

7M+
Developers

82K+
APIs in the Hub

9B+
API calls per month

Consume APIs

With indexed search functionality, discovering the perfect API match for your product roadmap is easier than

```
(Node.js) Axios  
Copy Code  
const axios = require("axios");  
  
const options = {  
  method: "GET",  
  url: "https://tank01-mlb-live-in-game-real-  
actions",  
}
```

Change Memo
Provide organizational change
Control today at \$12.99

Raw Materials Schedule
Get in line for API scheduling

Powered by Cloudinary

Link: <https://rapidapi.com/>



Rapid API



Iddo Gino

2015

NOKIA

2024



Rapid

Home Workspaces Explore Search Postman

Working locally in Scratch Pad. Switch to a Workspace

Scratch Pad New Import

Collections

- allanza_promocode
- auth-api
- demo
- Kiosco
- Kiosco backend
- Mercantil NN
- mineswesper
- New Collection
- POST New Request
- GET New Request
- NinjaExtraAPI
- Node-TS-Api-Base
- promotion
- PruebasPromotion
- redflag

Monitors

History

New Collection / New Request

GET https://zillow-com4.p.rapidapi.com/properties/search-coordinates?location=Houston%2C%20TX&eastLng=-94.517205&westLng=-96.1932233&southLat=29.170258&northLat=29.857524&status=forSale&sort=relevance

Params Authorization Headers (10) Body Pre-request Script Tests Settings

Headers 8 hidden

KEY	VALUE	DESCRIPTION
x-rapidapi-key	47dc174f4cmsh8332c65ae30cf8bp1e5cc0jsne264a9419954	
x-rapidapi-host	zillow-com4.p.rapidapi.com	
Key	Value	Description

Body Cookies Headers (14) Test Results

Pretty Raw Preview Visualize JSON

```

1  {
2    "data": [
3      {
4        "spid": 26886487,
5        "location": {
6          "latitude": 29.636186,
7          "longitude": -95.43865
8        },
9        "address": {
10         "streetAddress": "13114 White Heather Dr",
11         "zipcode": "77845",
12         "city": "Houston",
13         "state": "TX"
14       },
15       "media": {
16         "propertyPhotoLinks": {
17           "highResolutionLink": "https://photos.zillowstatic.com/fp/0b647577fb2d5839ea69e469d94d1081-p_f.jpg"
18         },
19         "thirdPartyPhotoLinks": {
20           "streetViewLink": "https://maps.googleapis.com/maps/api/streetview?size=1600x240&location=29.63618649188867,-95.4386529357422&key=AIzaSyBWY3WCARFNMqSe_k3LkfdGJaFgKwacoe
21             signature=95049-0MDVSp3HN38KZ7zCewYTi=",
22           "satelliteLink": "https://maps.googleapis.com/maps/api/staticmap?mobile=false&sensor=true&maptype=satellite&size=1600x480&zoom=19&center=29.63618649188867,-95.

```

Status: 200 OK Time: 2.88 s Size: 11735 KB Save Response

Find and Replace Console

Runner Trash



POSTMAN



Bruno
Reinventing the API Client



Insomnia



¿¿¿Preguntas???

Muchas Gracias



[Sergio Gabriel Garzón](#)



[sergiog90arg](#)