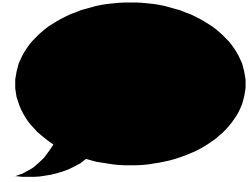


Testy

**A minimal testing
tool designed for
teaching**





iHi!

I'm Nahuel Garbezza

I make high-quality software at
10Pines, and I teach
Object-Oriented Design and
Test-Driven Development

1.

Context

How this tool was born?

Challenge: teaching using JS



Object-Oriented Design

Test-Driven Development



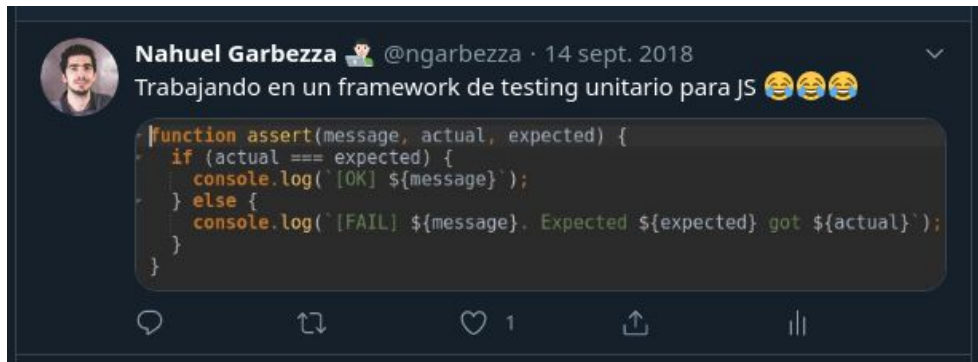
*Today we're going to
implement our own testing
tool!*

“Object-Oriented 2” class, UCA, 2018.

**I didn't know
where we could
go with this...**

...but I started with the `assert`

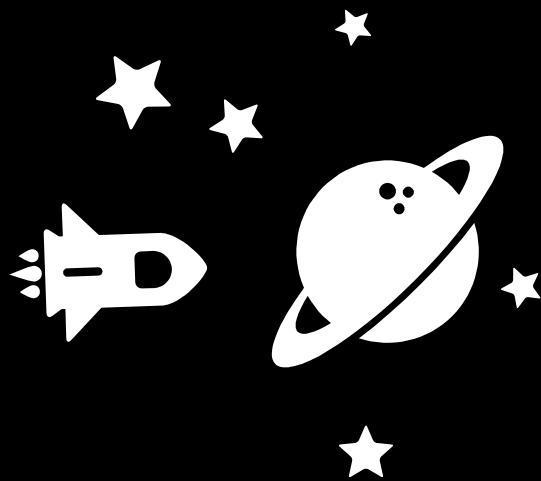
“Working on a testing framework for JS”



2.

Vision

¿What do we need? ¿Why another testing tool?



SIMPLICITY

Design Principles Behind Smalltalk

<https://www.cs.virginia.edu/~evans/cs655/readings/smalltalk.html>

Simplicity



Zero dependencies

The tool must be easy to install and configure. And it should run fast.

Jest has 328 dependencies that take 42 MB* of your node_modules

* as of 2021-10-21

Understandable and debuggable code

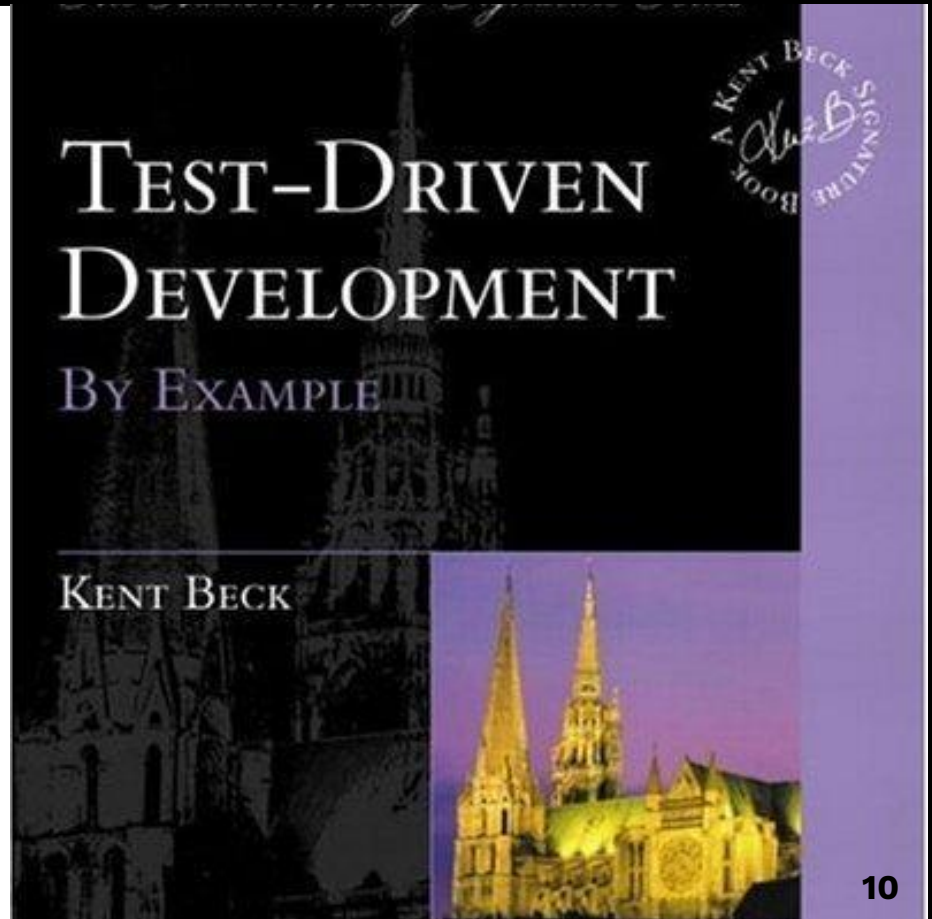
We (particularly students) should be able to navigate through the code and understand what it does. No metaprogramming involved.

Minimum functionality needed

It's not the goal to “copy” features from other testing tools.

Made for learn...

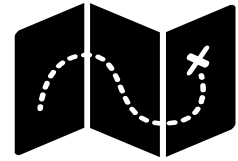
...to test, and to work with TDD.



3.

Testy, today

¿What do we use it for? ¿What features does it have?



Github (code, docs, issues)

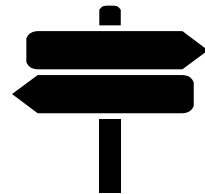
<https://github.com/ngarbezza/testy>

NPM

```
npm i --save-dev @pmoo/testy
```

Current version

5.1.0



Open Source

Documentation

Translated README files.
Exhaustive documentation of each feature. Contributing guidelines.

CI / Coverage / Releases

Continuous integration with Github Actions. Coverage and quality analysis provided by CodeClimate. Automated releases to NPM.

Issue templates

Bugs, features, refactoring proposals and documentation tasks. .

Badges

Quickly visualizing the project status through badges in the README.

<https://shields.io/>

Changelogs / SemVer

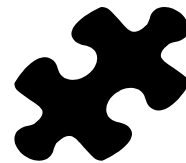
The project adheres to Keep a Changelog (<https://keepachangelog.com/en/1.0.0>) and Semantic Versioning (<https://semver.org/>)

Contributors bot

Bot that makes contributors visible and thank them for their work.

<https://github.com/all-contributors/all-contributors-bot>

Most important features



Suites and tests

Syntax to write tests organized in test suites.

Assertion language

Write different types of assertions using a fluent interface style.

Console output

Multilanguage support, different colors to represent test and suite statuses.

Pending tests

Support to flag tests as pending.

Fail Fast mode

Ability to configure the tool to fail in the first failed test.

Random order

Setting to ensure tests independence.

Unique functionality



Test without assertion considered error

We want to learn how to test well. Part of that learning process is to know if the tests we write are valid.

Undefined is not equal to anything

Compare undefined to undefined results in indetermination and it is considered a failure. There's `isUndefined()` assertion for explicit check for undefined.

Internal modules dependencies chart

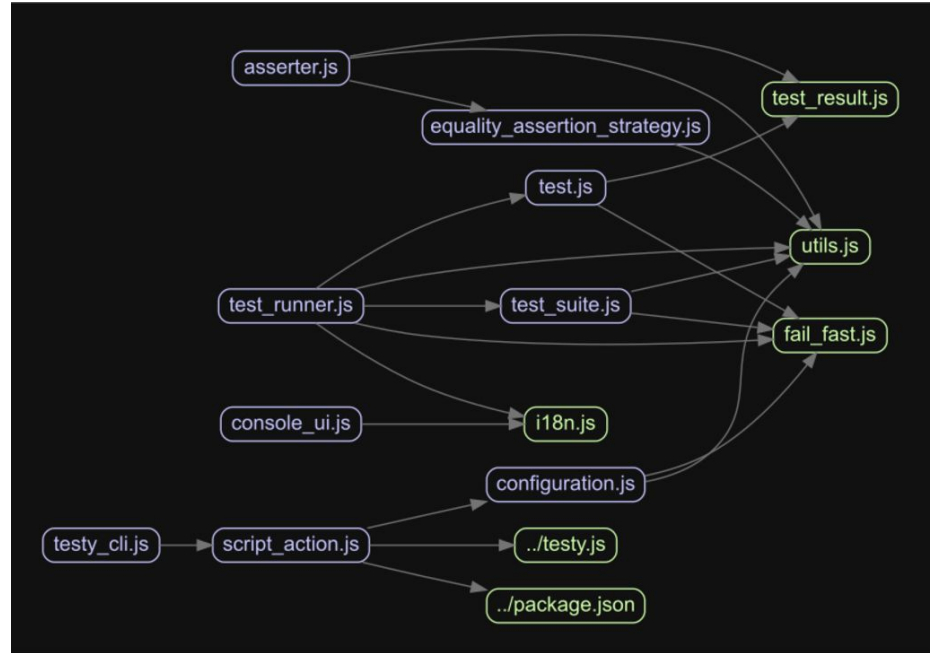


Image generated using Madge:
<https://github.com/pahen/madge>

4.

Demo

Let's see the tool in action



iThank you!

@ngarbezza (Twitter / Github)

<https://github.com/ngarbezza/testy>