推荐 - HTML、CSS、JavaScript 优秀的编辑器

PricingSupportDocsNewsBlogContact

Integrated Continuous Testing Tool for JavaScript

Wallaby.js is an intelligent and super fast **test runner for JavaScript** that continuously runs your tests. It reports **code coverage** and other results directly to your code editor, **immediately as you change your code**. Wallaby.js is **insanely fast**, because it **only executes tests affected by your code changes** and runs your tests **in parallel**. The tool provides a **huge productivity boost** whether you are doing **TDD/BDD** or using any other approach.

Where can I download wallaby.js?

Download

Wallaby.js for products based on IntelliJ Platform (WebStorm, PhpStorm, RubyMine, IntelliJ IDEA or PyCharm).

Install

To get the version of wallaby.js for Visual Studio Code, please install the Wallaby.js extension.

Install

To get the version of wallaby.js for Atom editor, please install the Atom-Wallaby package.

Download

Wallaby.js for Visual Studio 2013 (Community, Professional, Premium, Ultimate).

Download

Wallaby.js for Visual Studio 2015 (Community, Professional, Enterprise).

Download

Wallaby.js for Visual Studio 2017 (Community, Professional, Enterprise).

Install

To get the version of wallaby.js for Sublime Text, please install the Wallaby package via Sublime Text Package Manager.

Open

Once you have wallaby.js installed and running in any of the supported editors, you may also use Wallaby.js App to get **coverage reports and the realtime bird's eye view of your project's tests connected to your editor**.

To **install** the downloaded wallaby.js extension in your editor and **get started**, please follow these instructions.

Trial versions available for download are fully functional and should work for 30 days. Note that in the trial mode wallaby will sometimes prompt you to restart your editor to get another trial session. Please contact us if you **would like to arrange a trial period** for yourself or your team **without activation prompts and restarts**.

Who is using wallaby.js?

Thousands of individual professional developers and companies all over the world have already become more productive with wallaby.js. Even software development tools vendors, such as our friends from JetBrains, are trusting wallaby.js to help building other development tools.

JET BRAINS

LOCKHEED MARTIN

RIGHTCROWD
EVERYONE ON SITE IS RIGHT

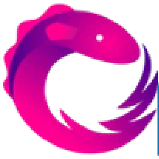
MLC®

etraveli

KAREN

APPLIED

INSILICO



LeisureLink™

parrotly

myBuilder.com

NEOCLES

Please contact us if you would like your company's or project's logo to be listed.

What are our users saying?

> Trying out @WallabyJS, an amazing JavaScript test runner that works in @code! https://t.co/wRsT1uNfDW
>
> — Scott Hanselman (@shanselman) January 13, 2016

> Testing @angular and @typescriptlang using the Angular CLI inside VS @code with @wallabyjs ... I love it! pic.twitter.com/IAnNIPfq0V
>
> — John Papa (@John_Papa) February 20, 2017

> Why wait until hitting save to know if your #JavaScript test is right?
>
> Get real-time test results with Wallaby: https://t.co/cShmvUUP9j
>
> — Cory House (@housecor) April 7, 2016

> Coding with @wallabyjs easily cut my development-to-complete time in half. Can't recommend it enough.
>
> — Sean Matheson (@controlplusb) April 4, 2016

> .@wallabyjs is a continuous test runner. Check out the plugin overview in our blog: http://t.co/9KwGzfWjN9
>
> — JetBrains WebStorm (@WebStormIDE) April 24, 2015

> #Javascript devs, have you met #WallabyJS? Slick, powerful, and...well, fun! https://t.co/mBqTZsnu1kpic.twitter.com/C05qdBBgfs
>
> — Microsoft Developer (@msdev) February 1, 2016

> Beautiful Seamless JavaScript Testing in 10 Minutes with Wallaby.js: https://t.co/R8LlVXibm4pic.twitter.com/JpviFJLWUy
>
> — JavaScript Daily (@JavaScriptDaily) September 14, 2015

> Worked on reducing down the 30 second wait for 1800 angular unit tests to run in Karma. Got it down to, well, instant with @wallabyjs :D
>
> — Andrew Shelton (@Sheltonial) March 4, 2015

> Got @wallabyjs to continuously run Jest tests in @WebStormIDE and it's the best thing ever □ □
>
> □ Sample config: https://t.co/OWP4zaXnuEpic.twitter.com/AFvvqjHsMO
>
> — kitze (@thekitze) November 25, 2016

> BLOG: Better #JavaScript & @typescriptlang Testing Experience with @code and @wallabyjshttps://t.co/B35ObaaHX8pic.twitter.com/AOPx0AQz6K
>
> — Andrew Connell (@andrewconnell) September 6, 2016

> Refactoring using @wallabyjs is a dream!
>
> — Pisman Kwinten (@KwintenP) November 11, 2016

> Since I've bought @wallabyjs my code coverage went sky-high. Never thought that this would happen.
>
> — Damian Kamiński (@_dkaminski_) March 21, 2016

> Whoaa. The company I work for just bought me a @wallabyjs license. Unit testing has never been so easy. #javascript

— Edvins Antonovs (@edvinsantonovs) November 14, 2016

WallabyJS just gave me super powers.https://t.co/b9wEC7mIDR

— Matthew Thornton (@Thornton_Matt) March 20, 2016

im only writing tests because @wallabyjs is awesooome :)

— Arnelle Balane (@arnellebalane) March 4, 2016

Thank you @wallabyjs for your great service. An answer AND solution to my question after 10 minutes already! #awesome#tdd

— Brecht Billiet (@brechtbilliet) September 29, 2016

Not a fan of testing? With https://t.co/EVC6zHAaO6#wallaby you will be

— LondonJavascript (@thelondonjs) September 6, 2016

For me barrier to doing TDD has always been better tooling esp in large projects, @wallabyjs however has changed that for me

— Nimesh Gurung (@nimgrg) October 4, 2016

Thank you @wallabyjs, an unlivable-without tool

— Alpha Shuro (@alphashuro) February 17, 2016

Just bought my @wallabyjs freedom licence. Its rare that a tool can have such a big impact on my workflow so quickly, I highly recommend it!

— James Henry (@MrJamesHenry) July 26, 2016

Once you try @code + @wallabyjs is hard to live without them.. #typescript#javascriptpic.twitter.com/NjHt6pMiOR

— Remo H. Jansen (@OweR_ReLoaDeD) February 15, 2016

@wallabyjs Seriously great tool, esp running only changed tests/code. I have 1000+ tests & Karma was beginning to be really slow. Thanks.

— Christopher Carson (@chriscarson) December 11, 2016

Well damn, @wallabyjs is just plain awesome for JS testing and coverage. They can expect to sell us a bunch of licenses soon!

— Hilke Heremans (@HHeremans) January 28, 2016

This is my #hailcorporate moment, but if you're javascripting, you should do yourself a favor and buy a @wallabyjs licence. Tests != pain

— □ (@mutantantihero) December 10, 2016

Thank you @wallabyjs for an amazing productivity boost!

— Cathal Coffey (@coffeycathal) January 27, 2016

Set up @wallabyjs with typescript for the VS @Code extension I'm working on. There should be something like this for every language, wow

— Cody Hoover (@hoovercj) January 15, 2016

Configured @wallabyjs with @typescriptlang@browserify and @code - took a few minutes - works amazingly! #javascript#typescript#html5

— Łukasz Wójciak (@lukaszwojciak) January 12, 2016

A: "ReSharper, NCrunch, and WallabyJs are this"
Q: "What are the 3 Visual Studio plugins every developer should use?"

— Calvin Allen (@CalvinAllen_) April 14, 2016

@wallabyjs is so rapid I now find waiting about half a second for NCrunch quite tedious when switching to my C# tests.

— Jamie Humphries (@_jamiehumphries) July 6, 2015

Just purchased a license for @wallabyjs to use with @code. 100% worth the money, so pumped to make #TDD a more natural part of my workflow.

— Ben White (@whtouche) April 17, 2016

JS dev? Do yourself a favor and check out @wallabyjs. Fantastic tool.

— Jake Prather (@JakeXSV) December 30, 2015

Developing @reactjs components and testing them with @wallabyjs: my #mocha tests are running live in @AtomEditor. I really love @wallabyjs

— Pierre Arnaud (@epsitec) November 4, 2015

Being extra productive with @wallabyjs#TypeScript and #VisualStudio@codehttps://t.co/e6A1Jht0Xb#javascriptpic.twitter.com/X2PG76AEMU

— Remo H. Jansen (@OweR_ReLoaDeD) February 25, 2016

Now I can see realtime test coverage for RxJS in editor with @wallabyjs , great help to increase test coverage. pic.twitter.com/0gO6Cm06Fi

— OJ Kwon (@_ojkwon) October 29, 2015

can't say enough good about @wallabyjs. Not only is it an amazing tool (whole dept now using), but @ArtemGovorov is great with support.

— Justin Obney (@justinobney) October 8, 2015

Ok, Wallaby.js is just frikken spectacular. Writing tests is both fluid and pleasurable now. The trial times out, so purchase it. Worth it.

— Sean Matheson (@controlplusb) February 9, 2016

Some tools are worth paying for: @wallabyjs is definitely one of them

— Steve Greatrex (@SteveGreatrex) May 8, 2015

Using @wallabyjs a few days now, already greatly increased efficiency of my #tdd cycle. #amazing

— Michael de Wit (@mjwdewit) April 24, 2015

#javascript TDD with @wallabyjs is sooo much more fun than with Karma! Thx @ArtemGovorov for the quick fix of the bug I reported

— Patrick Hund (@wiekatz) April 13, 2015

The best thing that happened to javascript development, testing suddenly became fun. #javascript#tddhttp://t.co/quEnBHKjLS@wallabyjs

— João Pedro Serra (@joaaoserra) October 14, 2015

First time of using @wallabyjs all day long at my professional work. Just DOPE! Works like a charm w/TypeScript,ES6,Mocha,Sinon,Chai,Atom.io

— Jens Krause (@sectore) September 2, 2015

@dan_abramov not sure what text editor you are using, but @wallabyjs for atom is incredible/game changing.

— Aaron Jensen (@aaronjensen) August 28, 2015

If you're a JavaScript dev, you have to check out @wallabyjs, you will not be disappointed, put your dev workflow on steroids.

— Wesley Higbee (@g0t4) August 19, 2015

I just did a decently complex refactor on my JS app — and using wallaby.js made it 10x faster.

— Bryan Mills (@diffkid) August 12, 2015

Build break? Forget to change your tests? Well, maybe let Wallaby keep an eye on your code as your write it. http://t.co/zswe5u0hau#testing

— Egor Kloos (@dutchcelt) August 14, 2015

ahh, there is nothing like the sweet smell of 100% code coverage :)
Thx @wallabyjs for creating such a great tool pic.twitter.com/1YywqoEaD1

— Dinis Cruz (@DinisCruz) August 8, 2015

Thanks @wallabyjs for the awesome swag to give away at our testing #angular talk at #ndclondon next week you rock! pic.twitter.com/Mh56BvsRGN

— Duncan Hunter (@dunchunter) January 13, 2017

Just renewed my @wallabyjs and upgraded to all editors. Will it save me $70 of time this year? It will save me $70 of time *this week*!

— Dave Brotherstone (@bruderstein) May 19, 2016

@iamstarkov tried istanbul/isparta. Didn't work well to me. Ended up using @wallabyjs. The best tool money can buy.

— Nikolay Matrosov (@nikthespirit) July 22, 2015

That moment you realise you've forgotten to start @wallabyjs and you've been wasting your time by manually running your unit tests.

— Rich Somerfield (@richie5um) November 18, 2016

Spent the day learning/writing jasmine #angularjs unit tests, have learnt! @wallabyjs makes the experience awesome

— Ken Ross (@kzhen) July 21, 2015

god damn, @WallabyJS is a FAST continuous test runner. Love it! #wallabyjs#jstests#continuoustestrunner#javascript#typescript#melike

— Mikkel Damm (@MikkelDamm) June 23, 2015

Most impressive JS test runner I've seen. http://t.co/93a7KYmjxC Worth a good look.

— Shawn McKay (@Sh_McK) June 21, 2015

@TylorS167 about a year in now and I literally cringe when I don't have it... @wallabyjs □

— Justin Obney (@justinobney) December 23, 2016

WOW.. 1 minute into @wallabyjs and I'm reeaaally impressed. Now for some Babel ES6/7 preprocessing

— Einar Ingebrigtsen (@Einarl) June 18, 2015

Man, @wallabyjs is a great test runner. Good support too. 2 hours from initial bug report to them pushing a fix live. :)

— Cody Hatch (@Codayus) June 17, 2015

VERY interesting. @Wallabyjs is *continuous JavaScript testing* for Visual Studio, running as you type. Cool! http://t.co/5c6bJWvV1A

— Scott Hanselman (@shanselman) June 5, 2015

Very impressed with the @wallabyjs team. Fixed an issue with Visual Studio extension within 2 hours! http://t.co/uB6TfEpmzr

— Jamie Humphries (@_jamiehumphries) May 28, 2015

Great example of using @wallabyjs to debug quite a nasty bug introduced by Angular 1.6.0 upgrade (see code comments)https://t.co/7nwKdrPYaNpic.twitter.com/uAkJkNQYdA

— Dinis Cruz (@DinisCruz) December 14, 2016

one of the best javascript engineering support tools you can buy: @wallabyjs

— Manuel Alabor (@swissmanu) May 13, 2015

.@wallabyjs is pretty slick...easy to setup/configure. Installed, configured, and was writing/running tests in ~10 minutes.

— Calvin Allen (@CalvinAllen_) December 12, 2015

Great to see @wallabyjs supporting more IDEs. It’ s the tool that made TDD JS pain free and possible for me https://t.co/8V1icyUYgB

— Tim Reynolds (@timjreynolds) November 18, 2015

thanks to @ArtemGovorov and @wallabyjs for supporting #mochajs

— Christopher Hiller (@b0neskull) January 16, 2017

> @ModernWebUIhttp://t.co/pxKck6GdEd is def going to change Javascript development, it is amazeballs
>
> — Joshua Gross (@joshuagross) March 23, 2015
>
> Wow. @wallabyjs plugin for @WebStormIDE is a game changer.
>
> — Justin Mandzik (@Justin_Mandzik) March 7, 2015

Show more happy users

How much does wallaby.js license cost?

There are no free tools, unless your time has no value. You may find our pricing and licensing details on the pricing page.

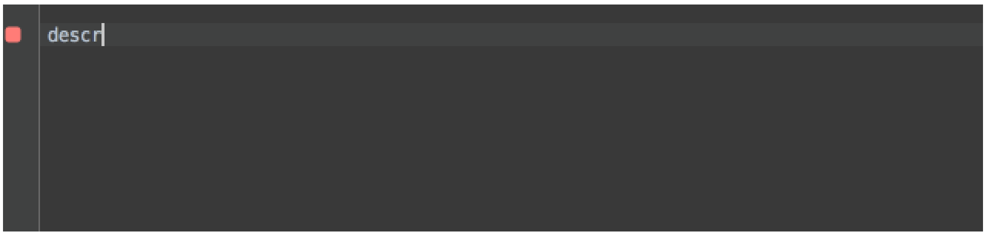
Do I need to use some new APIs in my tests or change my tests/code to run them with wallaby.js?

No. Wallaby.js just makes your code editor smarter and your tests to run faster. No need to switch and invest into new frameworks, commands, or APIs - just use your existing testing libraries.

There's **no vendor, API or framework lock-in when using wallaby.js**. You are getting a productivity boost with wallaby.js, but you will always be able to run your tests without the tool.

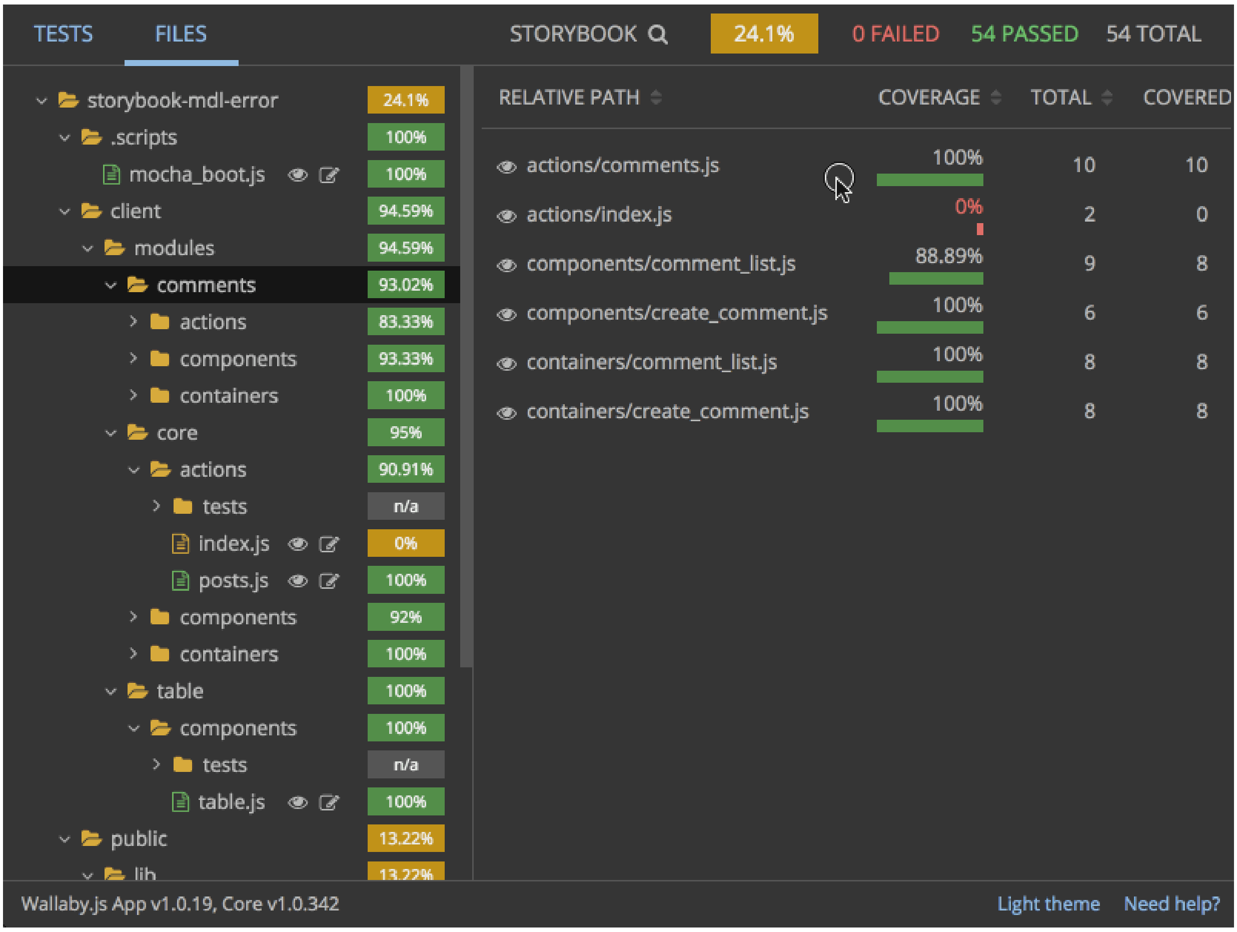
What is wallaby.js again?

Wallaby.js is an intelligent and super fast test runner deeply integrated into your editor or IDE, that enables highly productive JavaScript unit testing workflow. It runs your unit tests as you type and displays various results (including code coverage for JavaScript, TypeScript, and CoffeeScript) right inside your code editor. Wallaby.js is great for doing **JavaScript TDD (Test-driven development) or BDD (Behavior Driven Development)**, while also working for other approaches.



```
descr
```

Once you have wallaby.js installed and running in any of the supported editors, you may also use Wallaby.js App to get the **realtime reports and analysis** connected to your editor.

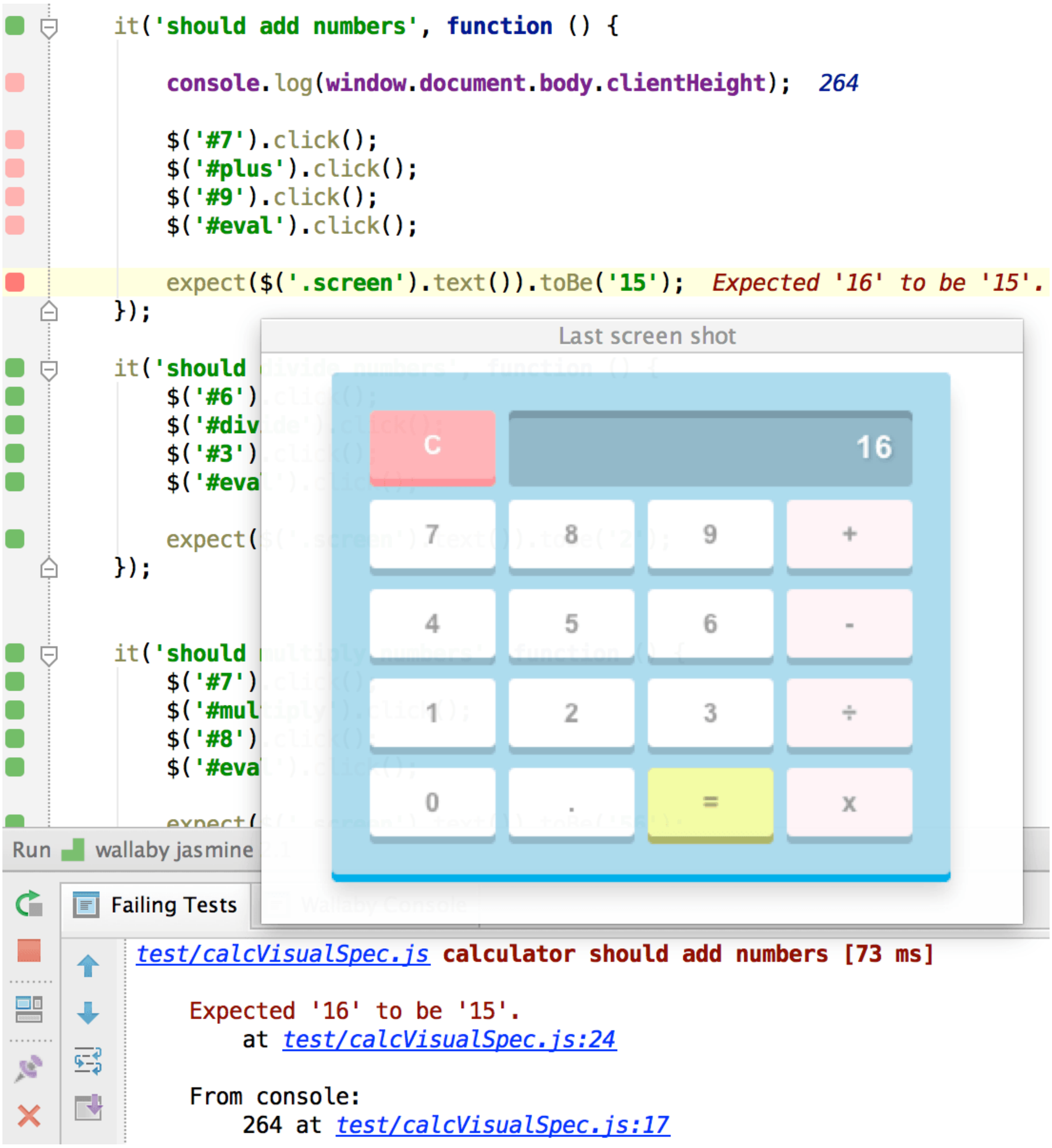


storybook-mdl-error	24.1%
.scripts	100%
mocha_boot.js	100%
client	94.59%
modules	94.59%
comments	93.02%
actions	83.33%
components	93.33%
containers	100%
core	95%
actions	90.91%
tests	n/a
index.js	0%
posts.js	100%
components	92%
containers	100%
table	100%
components	100%
tests	n/a
table.js	100%
public	13.22%
lib	13.22%

RELATIVE PATH	COVERAGE	TOTAL	COVERED
actions/comments.js	100%	10	10
actions/index.js	0%	2	0
components/comment_list.js	88.89%	9	8
components/create_comment.js	100%	6	6
containers/comment_list.js	100%	8	8
containers/create_comment.js	100%	8	8

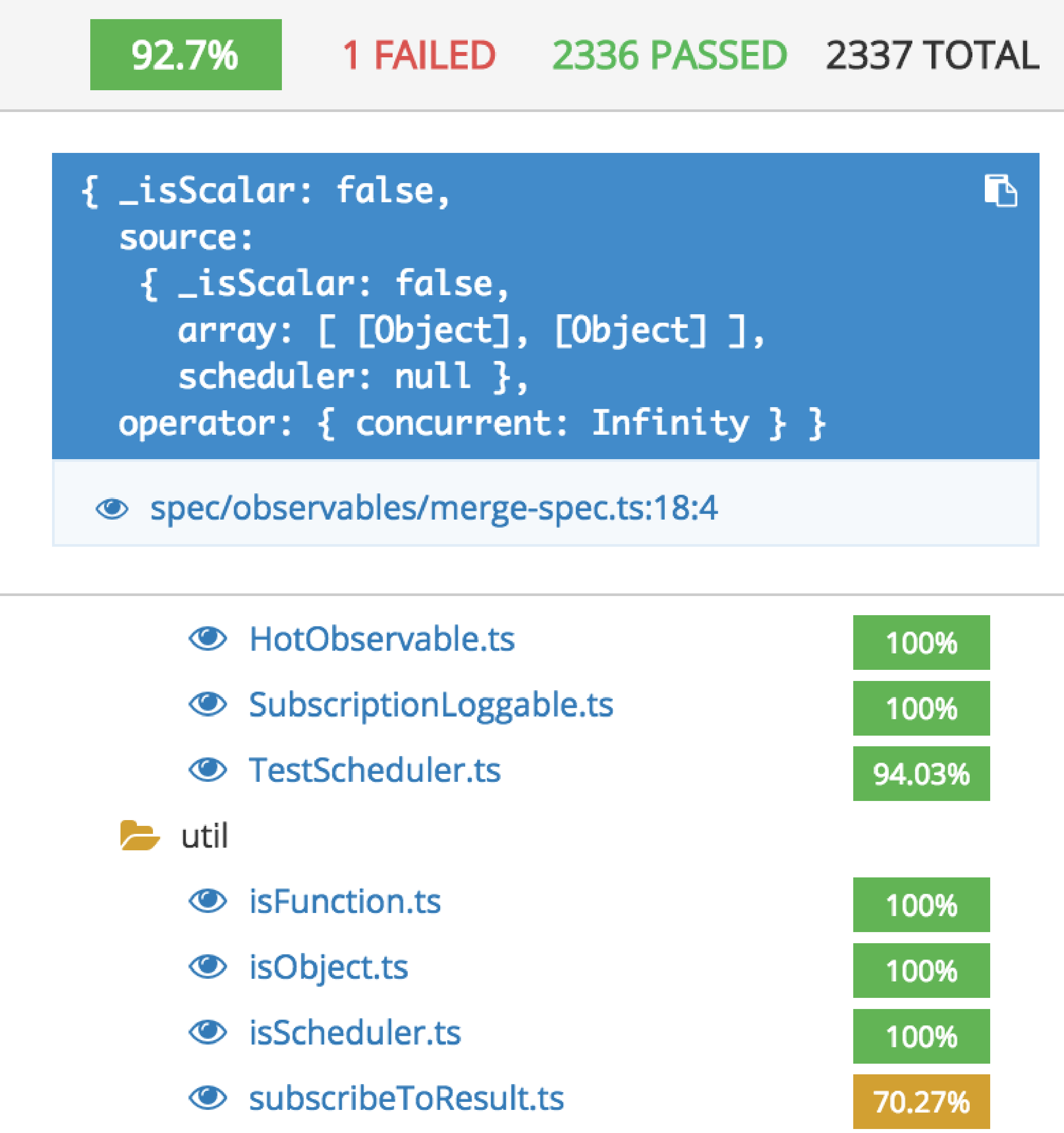
Wallaby.js App v1.0.19, Core v1.0.342

What about a list of main features?



```
it('should add numbers', function () {

    console.log(window.document.body.clientHeight);  264

    $('#7').click();
    $('#plus').click();
    $('#9').click();
    $('#eval').click();

    expect($('.screen').text()).toBe('15');  Expected '16' to be '15'.
});

it('should divide numbers', function () {
    $('#6').click();
    $('#divide').click();
    $('#3').click();
    $('#eval').click();

    expect($('.screen').text()).toBe('2');
});


it('should multiply numbers', function () {
    $('#7').click();
    $('#multiply').click();
    $('#8').click();
    $('#eval').click();

    expect($('.screen').text()).toBe('56');
```

```
test/calcVisualSpec.js calculator should add numbers [73 ms]

    Expected '16' to be '15'.
        at test/calcVisualSpec.js:24

    From console:
        264 at test/calcVisualSpec.js:17
```



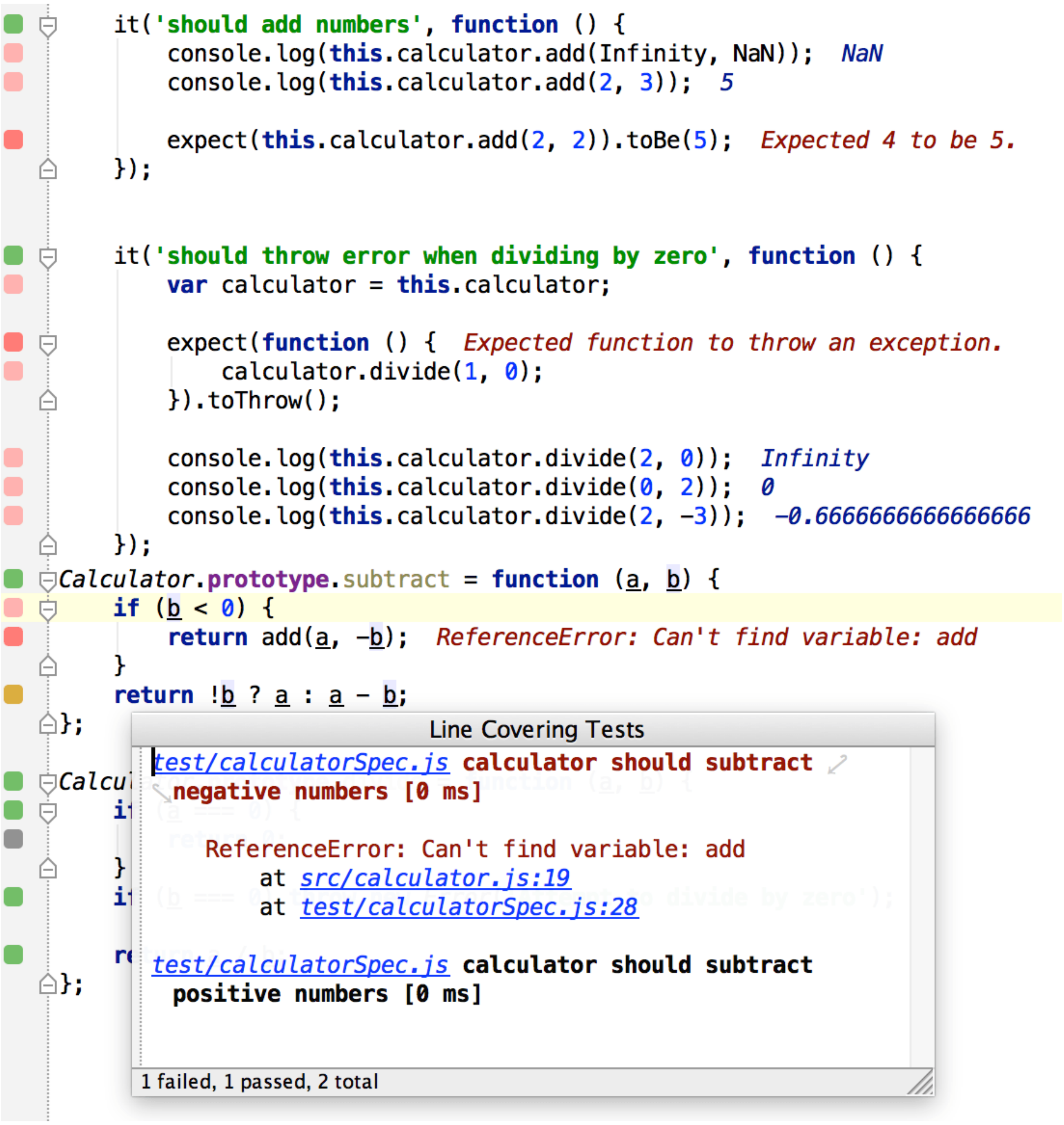
92.7% 1 FAILED 2336 PASSED 2337 TOTAL

```
{ _isScalar: false,
  source:
   { _isScalar: false,
     array: [ [Object], [Object] ],
     scheduler: null },
  operator: { concurrent: Infinity } }
```

spec/observables/merge-spec.ts:18:4

HotObservable.ts	100%
SubscriptionLoggable.ts	100%
TestScheduler.ts	94.03%
util	
isFunction.ts	100%
isObject.ts	100%
isScheduler.ts	100%
subscribeToResult.ts	70.27%

- Supports continuous testing in your code editor:
  - **WebStorm, IntelliJ IDEA, PhpStorm, RubyMine, PyCharm**;
  - **Visual Studio Code;**
  - **Atom Text Editor;**
  - **Visual Studio 2013 (Update 4-5),**
    Visual Studio 2015,
    Visual Studio 2017;
  - **Sublime Text.**
- Supports **browser code unit testing** (via PhantomJs, Electron or node.js) and **node.js unit testing**.
- Shows live test coverage.
- Provides beautiful realtime **test and coverage reports**.
- Shows failed expectations, errors and console.log messages inline, where and when they occur.
- Supports many testing frameworks (**Jasmine, QUnit, Mocha, Jest, and AVA**).
- Supports **ES6, ES7, JSX**.
- Supports **TypeScript, CoffeeScript, Flow**.
- Supports **Webpack and Browserify**.
- Extensible via preprocessors, compilers and more.
- Runs tests affected by code changes.
- Runs tests in parallel.
- Runs selected test(s).
- Captures test execution screenshots.



```
it('should add numbers', function () {
    console.log(this.calculator.add(Infinity, NaN));  NaN
    console.log(this.calculator.add(2, 3));  5

    expect(this.calculator.add(2, 2)).toBe(5);  Expected 4 to be 5.
});


it('should throw error when dividing by zero', function () {
    var calculator = this.calculator;

    expect(function () {  Expected function to throw an exception.
        calculator.divide(1, 0);
    }).toThrow();

    console.log(this.calculator.divide(2, 0));  Infinity
    console.log(this.calculator.divide(0, 2));  0
    console.log(this.calculator.divide(2, -3));  -0.6666666666666666
});
Calculator.prototype.subtract = function (a, b) {
    if (b < 0) {
        return add(a, -b);  ReferenceError: Can't find variable: add
    }
    return !b ? a : a - b;
};

Calcu
    if

    }
    if

    re
};
```

Line Covering Tests

```
test/calculatorSpec.js calculator should subtract
 negative numbers [0 ms]

    ReferenceError: Can't find variable: add
        at src/calculator.js:19
        at test/calculatorSpec.js:28

test/calculatorSpec.js calculator should subtract
 positive numbers [0 ms]
```

1 failed, 1 passed, 2 total

Where can I find more information about wallaby.js, such as tutorials and samples?

You can find more information, detailed tutorials and sample project links in our documentation.

How is it different from Karma, Mocha runner, etc.?

With other test runners you either need to run your tests manually, or configure them to run the tests when you save your source code files. Those runners execute all of your tests, and, as your project grows, it takes more and more time to complete the task. The test execution results are displayed somewhere outside of your code editor, and constant context switching is a productivity killer.

Wallaby.js runs your unit tests **immediately as you type**, no need to run anything manually or even save files. The tool calculates and **runs the minimum required number of tests affected by your code changes**, so no matter how large your project grows - the feedback is almost instant. Test execution results, including code coverage, are displayed and updated in real time right where you need it — in your code editor, **next to the line of code that you're editing**.

Can I use wallaby.js with Jasmine, Mocha, QUnit, AVA, or Jest?

Yes. You can use your favorite testing framework, any mocking framework (such as Sinon.js), any assertion library (such as Chai.js, Expect.js, Should.js, Jasmine built-in one, etc.) with wallaby.js.

Does wallaby.js support Angular.js, React, Backbone.js, Aurelia, etc.?

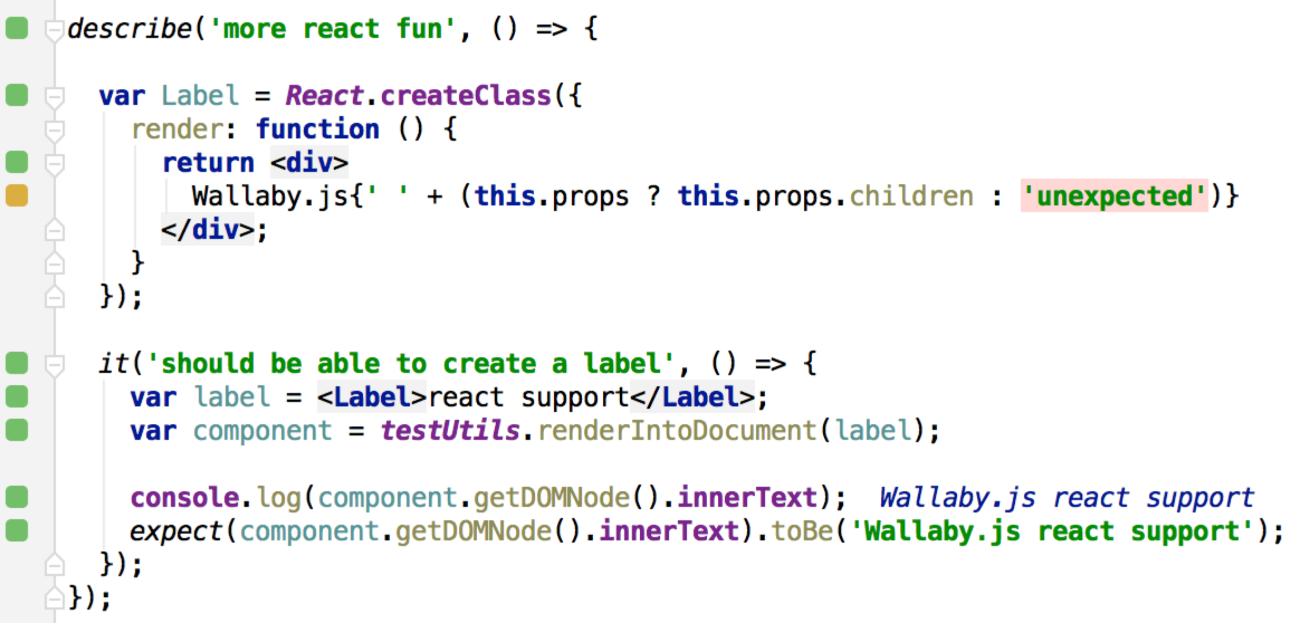
Yes. As long as you can write tests for your application, wallaby.js can run them. Check out our tutorials for testing Angular.js and React applications.

What about node.js?

Wallaby.js supports node.js testing. In fact, wallaby.js is written in node.js and we write tests for wallaby.js using wallaby.js.

What about ES6, ES7 or JSX?

Wallaby.js supports ES6, ES7 and JSX. ES6 is supported both natively for engines that can execute it and via code transpilers, such as Babel. You can see realtime coverage for your generators, classes, arrow functions and much more with wallaby.js. For JSX, not only does wallaby.js understand its syntax, but it also displays coverage inside JavaScript expressions within JSX elements.



```
describe('more react fun', () => {

  var Label = React.createClass({
    render: function () {
      return <div>
        Wallaby.js{' ' + (this.props ? this.props.children : 'unexpected')}
      </div>;
    }
  });

  it('should be able to create a label', () => {
    var label = <Label>react support</Label>;
    var component = testUtils.renderIntoDocument(label);

    console.log(component.getDOMNode().innerText);  Wallaby.js react support
    expect(component.getDOMNode().innerText).toBe('Wallaby.js react support');
  });
});
```

What about TypeScript, CoffeeScript, Flow?

Wallaby.js has zero-configuration support for TypeScript and CoffeeScript. Babel is also fully supported.

What about Browserify or Webpack?

Wallaby.js supports both Browserify and Webpack via **open source plugins**. These plugins not only use incremental compilation, but also leverage browser caching to provide the best performance.

What about code preprocessors like in Karma?

Wallaby.js supports Karma-like preprocessors. Simply provide a function invoking an existing node module.

Do I have to install any standalone applications with wallaby.js?

No. All you need to install is the wallaby.js plugin for your code editor, period. **No new command line tools or browser plugins**. Our goal is not to add more things between you and the feedback from your tests. In fact, it's the opposite: wallaby.js eliminates any unnecessary context switching, so it's just you and your test feedback, right where and when you write your code.

Do I have to use TDD with wallaby.js or do I have to write my code first?

Wallaby.js is agnostic of any software development techniques. It helps you be more productive whether you're doing JavaScript TDD, writing tests after, or using any mixed approach.

Who stands behind wallaby.js?

At the core of our team are professional JavaScript developers and professional development tools makers. Not only do we know and love the language and the ecosystem, but we have also dedicated our careers to making programmers happier and more productive.

Where is your changelog, and what does the roadmap for wallaby.js look like?

Our changelog is in our public repository.

Wallaby.js roadmap:

- Implementing better test debugging.
- Implementing CI server support.
- Support for more testing frameworks.
- Adding more features to Wallaby.js App.
- Simplifying wallaby.js configuration process.
- Are we missing something? Share your idea or vote for other submitted ideas.

How can I contact you?

Please feel free to send us an email to **[email protected]** or a message on twitter @WallabyJs and we will get back to you shortly.

How can I subscribe to wallaby.js news?

Follow us on twitter @WallabyJs to receive all the latest news.

We also send out project updates every now and then. Have a look at our past newsletters to decide if they’ re worth subscribing to.

Email Address: