

Generating docs in the static website era



Alexandre Stanislawski

Software engineer / Algolia

@boblito



boblito



About me

- > Software engineer at Algolia
- > Lead of InstantSearch.js
- > Side topic: docs of our main libraries

Algolia + docs = <3

Dedicated documentation team

DocSearch (React, vue.js, Laravel...)

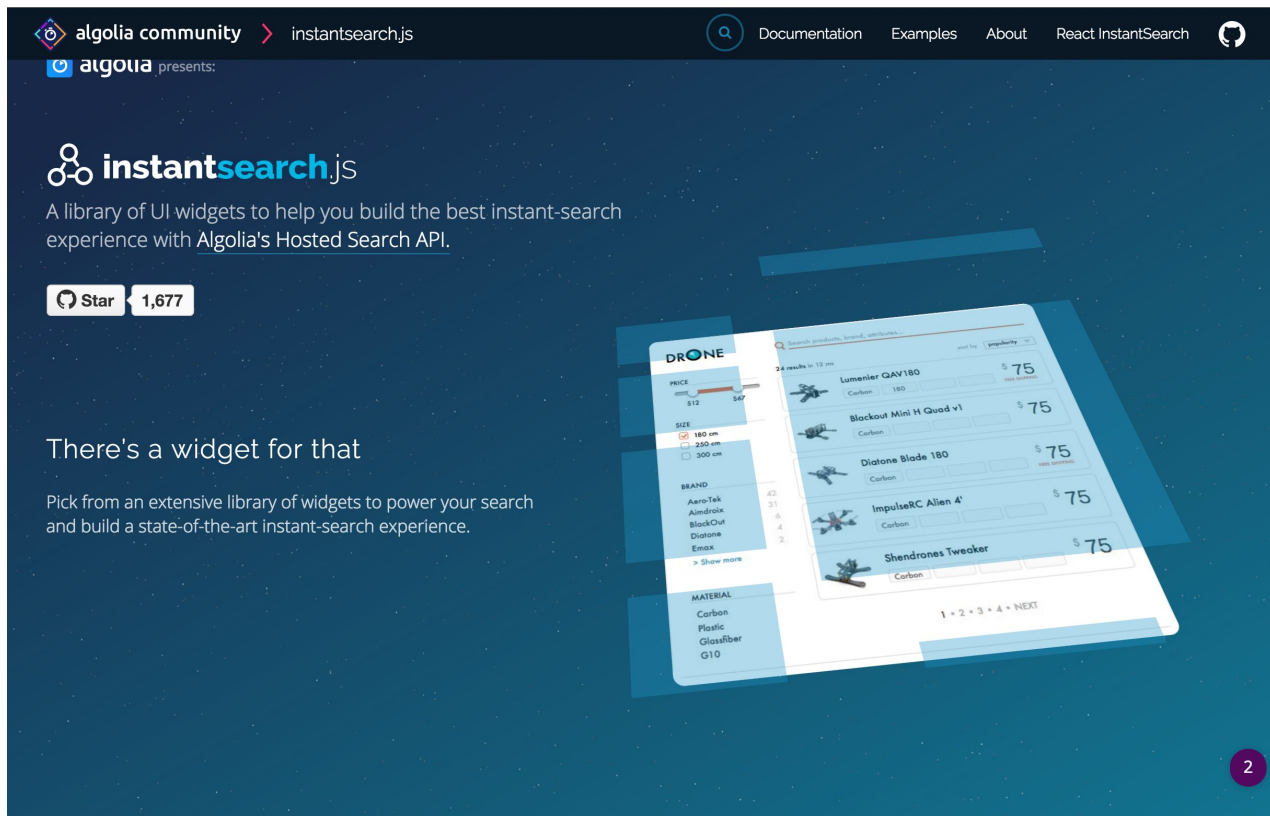
Main libraries have their documentation websites

And now a little story

2015: InstantSearch.js v1



Jekyll was cool



The screenshot shows the instantsearch.js documentation page. At the top, there's a navigation bar with 'algolia community' and 'instantsearch.js'. Below this, the Algolia logo is followed by 'presents:'. The main heading is 'instantsearch.js' with a subtext: 'A library of UI widgets to help you build the best instant-search experience with [Algolia's Hosted Search API](#).' Below the heading is a GitHub Star button showing '1,677' stars. The main content area has the text 'There's a widget for that' followed by a paragraph: 'Pick from an extensive library of widgets to power your search and build a state-of-the-art instant-search experience.' On the right side, there's a large image of a tablet displaying a search interface for drones. The interface includes a search bar, filters for price, size, brand, and material, and a list of drone products with their prices.

algolia community > instantsearch.js

algolia presents:

instantsearch.js

A library of UI widgets to help you build the best instant-search experience with [Algolia's Hosted Search API](#).

Star 1,677

There's a widget for that

Pick from an extensive library of widgets to power your search and build a state-of-the-art instant-search experience.

DRONE

PRICE
212 347

SIZE
180 cm
250 cm
300 cm

BRAND
Aero-Tek 42
Alundrox 31
BlockOut 6
Diatone 4
Emax 2
> Show more

MATERIAL
Carbon
Plastic
Glassfiber
G10

24 results in 12 ms

Product	Material	Price
Lumenier QAV180	Carbon	\$ 75
Blockout Mini H Quad v1	Carbon	\$ 75
Diatone Blade 180	Carbon	\$ 75
ImpulseRC Alien 4"	Carbon	\$ 75
Shendrones Tweaker	Carbon	\$ 75

1 • 2 • 3 • 4 • NEXT

JsDoc less so

Class: InstantSearch

InstantSearch()

The actual implementation of the InstantSearch. This is created using the `instantsearch` factory function.

Constructor

`new InstantSearch()`

Source: [lib/InstantSearch.js, line 35](#)

Fires:

- `Instantsearch#event:render` This event is triggered each time a render is done

Methods

`addWidget(widget) → {undefined}`

Add a widget

Parameters:

Name	Type	Description
<code>widget</code>	Widget	The widget to add to InstantSearch. Widgets are simple objects that have methods that map the search lifecycle in a UI perspective. Usually widgets are created by <code>[widget factories](widgets.html)</code> like the one provided with <code>InstantSearch.js</code> .

Source: [lib/InstantSearch.js, line 84](#)

Home

Modules

[widgets](#)
[instantsearch](#)

Externals

[HTMLElement](#)
[SearchParameters](#)

Classes

[InstantSearch](#)

Events

[render](#)

Global

[bemHelper](#)
[getContainerNode](#)
[hashUrlUtils](#)
[isDomElement](#)
[modernUrlUtils](#)
[prepareTemplateProps](#)
[urlSync](#)

jsDoc: 1 word, two things

> the tool

> the annotations in the code

Reasons for keeping annotations

- > **Exhaustivity**: all annotations are translated into doc
- > **Collocation**: doc lives with the code

Before it was cool...

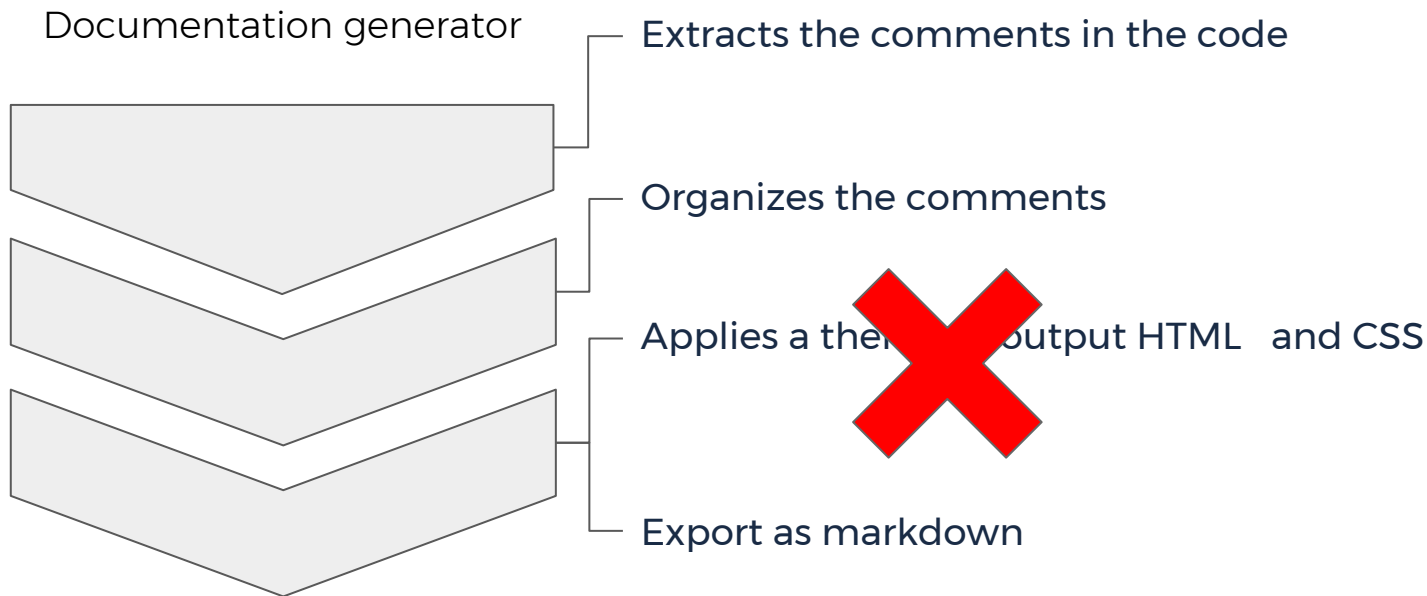
Documentation generators (like JSDoc) are the ancestors of static website generators

They do the same for docs than Jekyll does for blogs.

And they haven't aged well

- > poor / unique templating system
- > no css preprocessor
- > no modern js stack

Integrate the JSDoc in Jekyll?



Not the only one!

Lo 4.17.4

Search

Array

- [_.chunk](#)
- [_.compact](#)
- [_.concat](#)
- [_.difference](#)
- [_.differenceBy](#)
- [_.differenceWith](#)
- [_.drop](#)
- [_.dropRight](#)
- [_.dropRightWhile](#)
- [_.dropWhile](#)
- [_.fill](#)
- [_.findIndex](#)
- [_.findLastIndex](#)
- [_.first -> head](#)
- [_.flatten](#)
- [_.flattenDeep](#)
- [_.flattenDepth](#)
- [_.fromPairs](#)
- [_.head](#)
- [_.indexOf](#)
- [_.initial](#)
- [_.intersection](#)
- [_.intersectionBy](#)

“Array” Methods

`_.chunk(array, [size=1])`

[source](#) [npm package](#)

Creates an array of elements split into groups the length of size. If array can't be split evenly, the final chunk will be the remaining elements.

Since

3.0.0

Arguments

array (Array): The array to process.
[size=1] (number): The length of each chunk

Returns

(Array): Returns the new array of chunks.

Example

```
_.chunk(['a', 'b', 'c', 'd'], 2);
// => [['a', 'b'], ['c', 'd']]

_.chunk(['a', 'b', 'c', 'd'], 3);
// => [['a', 'b', 'c'], ['d']]
```

[Try in REPL](#)

Our solution

1. A script generates markdown files from *.js
2. The markdown can then be integrated to Jekyll

And in our case, we used [jsdoc-to-markdown](#)

Introduction

Setup

Light build

React InstantSearch

Initialization

Adding Widgets

Start

Events

Widgets

Basics

- ☐ [searchBox](#)
- ☐ [hits](#)
- ☐ [Infinite hits](#)
- ☐ [hitsPerPageSelector](#)

Navigation

- ☐ [pagination](#)
- ☐ [menu](#)
- ☐ [hierarchicalMenu](#)

Filters

- ☐ [refinementList](#)
- ☐ [numericRefinementList](#)
- ☐ [toggle](#)
- ☐ [rangeSlider](#)
- ☐ [priceRanges](#)
- ☐ [numericSelector](#)
- ☐ [starRating](#)
- ☐ [clearAll](#)
- ☐ [currentRefinedValues](#)

Sort

- ☐ [sortBySelector](#)

Metadata

Initialization

To initialize the `instantsearch.js` library, you need an Algolia account with a configured and non-empty index.

You can find your Algolia credentials on the [credentials page of your dashboard](#).

Use the `APPLICATION_ID` `appId`, the search only `API_KEY` `apiKey` and an index name `indexName` to configure the required parameters of the `instantsearch` function.

If you don't have any indices yet, learn how to push your data with the [Algolia getting started guide](#).

Url synchronization

You can synchronise the current search with the browser url. It provides two benefits:

- Working back/next browser buttons
- Copy and share the current search url

To configure this feature, pass `urlSync: true` option to `instantsearch()`.

The `urlSync` option has more parameters, see the `instantsearch` function documentation.

```
instantsearch(options);
```

Parameters

<code>options.appId</code>	(string)
The Algolia application ID	
<code>options.apiKey</code>	(string)
The Algolia search-only API key	
<code>options.indexName</code>	(string)
The name of the main index	
<code>options.numberLocale</code> ...	(string)
<code>options.searchFunction</code> ...	(function)
<code>options.createAlgoliaClient</code> ...	(function)
<code>options.searchParameters</code> ...	(object)
<code>options.urlSync</code> ...	(Object boolean)
<code>options.urlSync.mapping</code> ...	(object)
<code>options.urlSync.threshold</code> ...	(number)
<code>options.urlSync.trackedParameters</code> ...	(Array.<string>)
<code>options.urlSync.useHash</code> ...	(boolean)
<code>options.urlSync.urlHistoryState</code> ...	(function)

* Required

Let's fast forward to
now

Using modern tools for docs

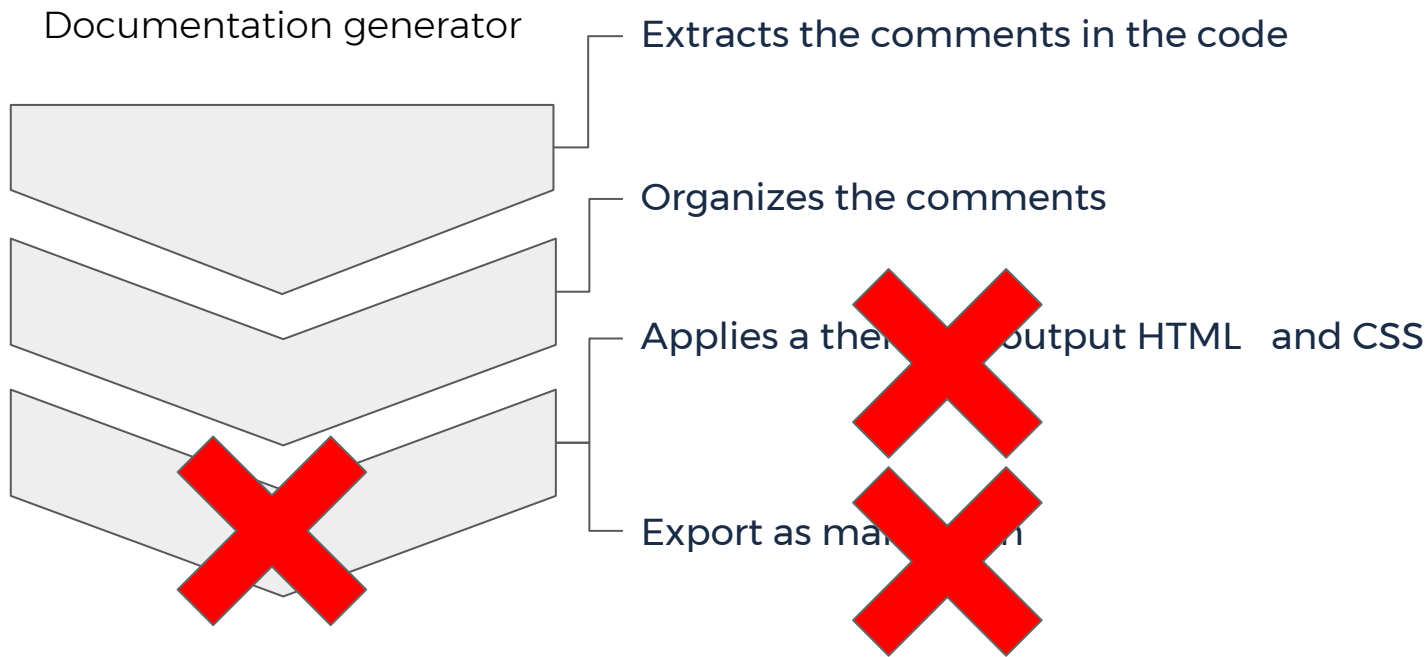
> Metalsmith

- Not focussed on blogs
- Written in JS
- Easy to build plugins

> **PUG + Sass** (+ webpack + handlebars)

> And a custom plugin for **documentation.js**

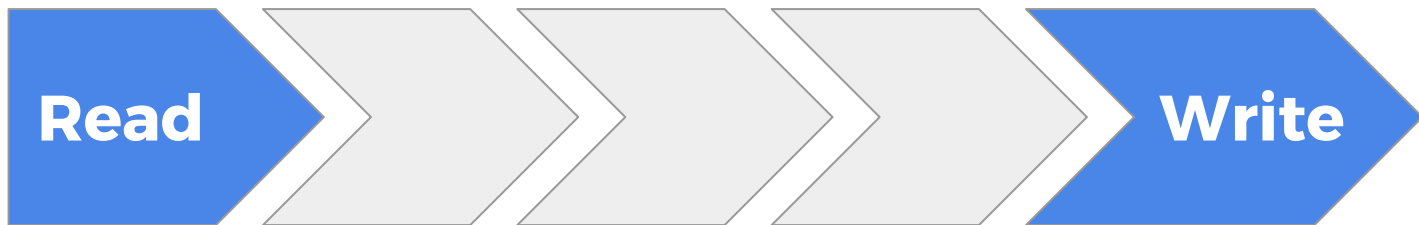
Bypassing the output



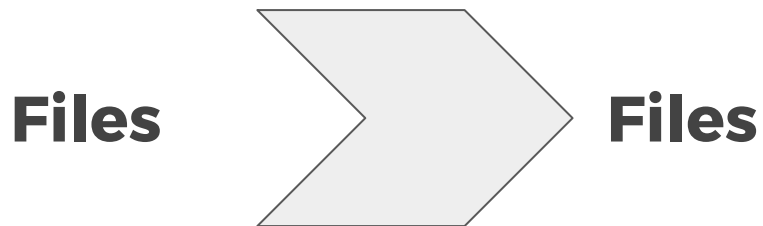
Documentation.js

- > Fully compatible with jsDoc comments
- > Under active development
- > Built-in javascript API

Metalsmith architecture



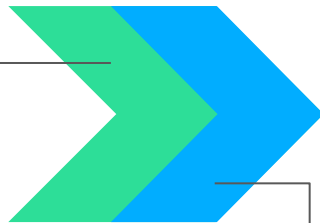
Metalsmith plugin



Documentation.js plugin

Extracts JS
documentation
data

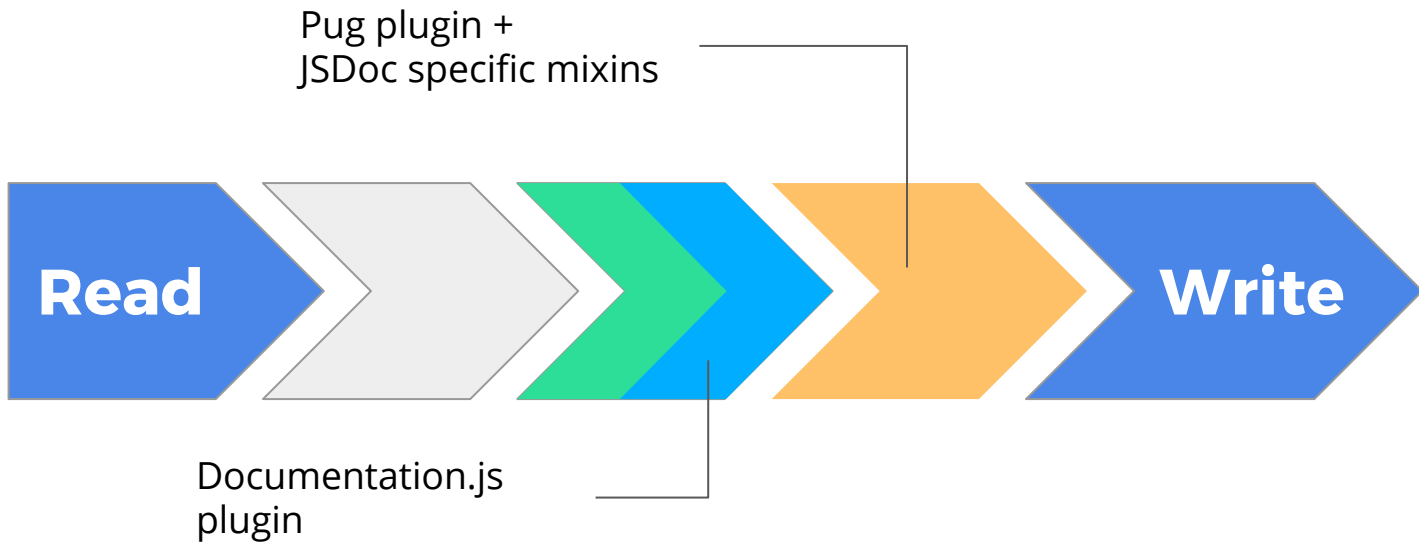
Files



Files

Organizes
documentation
in files

Our pipeline



Results

- > Editorial content with reference doc
- > Tools you know
- > Requires some time investment

Get more from your doc!

Easy contributions

 algolia >  community > InstantSearch.js

 [Getting Started](#) [Guides](#) [API](#) [Demos](#) [v1 docs](#) 

widgets ► analytics

[EDIT THIS PAGE](#)

[Introduction](#)

[Common parameters](#)

[analytics](#)

● **Description**

[Usage](#)

[Options](#)

[Example](#)

github.com/algolia/instantsearch.js/edit/develop/src/widgets/analytics/analytics.js

Description

The analytics widget pushes the current state of the search to the analytics platform of your choice. It requires the implementation of a function that will push the data.

This is a headless widget, which means that it does not have a rendered output in the UI.

Usage

Usage

```
const search = instantsearch(/* parameters */);
```

Topic oriented documentation

- > The API can be separated in concepts
- > Organizing topics with the learning curve in mind
- > Driven by the business logic of the doc based on the jsDoc data

Runnable example



InstantSearch.js



Getting Started Guides ▾ API ▾ Demos 

Example

Code

Run

```
search.addWidget(  
  instantsearch.widgets.menu({  
    container: '#categories',  
    attributeName: 'hierarchicalCategories.lvl0',  
    limit: 10,  
    templates: {  
      header: 'Categories'  
    }  
  })  
);
```

Menu widget example

CATEGORIES

Appliances 4,306

Audio 1,570

Cameras & Camcorders 1,369

Car Electronics & GPS 1,208

Cell Phones 3,291

Computers & Tablets 3,563

Health, Fitness & Beauty 923

Office & School Supplies 617

TV & Home Theater 1,201

Video Games 505

Introduction

Common parameters

analytics

clearAll

currentRefinedValues

hierarchicalMenu

hits

hitsPerPageSelector

infiniteHits

menu

Usage



InstantSearch.js



Getting Started Guides • API • Demos



Usage

Usage

```
const search = instantsearch(/* parameters */);

const widget = instantsearch.widgets.currentRefinedValues({
  container: string|HTMLElement,
  attributes: [Array<CurrentRefinedValuesAttributes>],
  onlyListedAttributes: [boolean],
  clearAll: ["before"|"after"|boolean],
  templates: [CurrentRefinedValuesTemplates],
  transformData: [CurrentRefinedValuesTransforms],
  autoHideContainer: [boolean],
  cssClasses: [CurrentRefinedValuesCSSClasses],
  collapsible: [boolean|{collapsed: boolean}],
  clearsQuery: [boolean],
}); CurrentRefinedValuesWidgetOptions);

search.addWidget(widget);
```

Options

CurrentRefinedValuesWidgetOptions

container `string|HTMLElement`

CSS Selector or HTMLElement to insert the widget

Introduction

Common parameters

analytics

clearAll

currentRefinedValues

Description

● Usage

Options

Example

hierarchicalMenu

hits

hitsPerPageSelector

infiniteHits

menu

numericRefinementList

numericSelector

Extend the semantic

[algolia](#) > [community](#) > [React InstantSearch](#)[Q](#) [Getting Started](#) • [Learn](#) • [API](#) • [Examples](#) • [🔗](#)

Connectors ▶ connectHierarchicalMenu

Connectors

[connectAutoComplete](#)
[connectBreadcrumb](#)
[connectCurrentRefinements](#)
[connectHierarchicalMenu](#)
[connectHighlight](#)
[connectHits](#)
[connectHitsPerPage](#)
[connectInfiniteHits](#)
[connectMenu](#)
[connectMultiRange](#)
[connectPagination](#)
[connectPoweredBy](#)
[connectRange](#)
[connectRefinementList](#)
[connectScrollTo](#)

Description

Add instant-search and autocomplete to your React and React Native apps with a few lines of code. Powered by Algolia.

Exposed props

Props exposed on the connected component:

attributes*	type: <code>string</code>	
List of attributes to use to generate the hierarchy of the menu. See the example for the convention to follow.		
...

Reflexions and future

Current stack

- > Metalsmith is small and very easy to hack
- > Maybe other tools could fit too
- > Other languages could benefit from this approach

Code semantics

- > jsDoc is typing without verifications
- > documentation.js support flow as typing
- > Double bonus: Writing type once + checks

Conclusion

- > Documentation is key for a successful library
- > You don't have to rely on old stacks
- > Documentation is data, you can harness it



Thank you

Questions?



Alexandre Stanislawski

Software engineer / Algolia

@boblito



boblito



Reference

- [InstantSearch.js documentation](#)
- [react-instantsearch documentation](#)

Tools

- [Documentation.js](#)
- [Metalsmith](#)
- [jsdoc-to-markdown](#)
- [jsdoc](#)

Some interesting blog post on the doc topic

- [Making docs more inspiring](#)
- [In defense of jsDoc](#)