Index content of your <u>hexo</u> website in <u>Algolia</u> and add search within minutes.

hexo-algolia is an hexo plugin provided by the community. This is what you get when you install it:

- 1. a **new command**, hexo algolia, to index the content of your website
- 2. a theme helper to include Algolia search client
- 3. another **theme helper** to configure the Algolia search client
- The hexo algolia command can be run manually on your computer and on a continuous integration system like [Travis CI][-ci-node].
 - Browse the <u>CHANGELOG</u> to learn what has changed between versions.
- Compatible with node>=4.0.0.

hexo theme

Install

```
1 $ npm install --save hexo-algolia
```

Public Facing Search Options

You can configure Algolia integration to your hexo website with the _config.yml file:

```
algolia:
applicationID: 'applicationID'
apiKey: 'apiKey'
indexName: '...'
```

Config Key	
applicationID	Your Algolia Application ID
аріКеу	A Search-Only API key
indexName	The name of the Algolia index to use

These configuration values are **accessible from your hexo theme**, to be used with <u>Algolia JavaScript client</u>.

Hexo Theme Setup

Helpers are provided to make your life easier.

Include Algolia JavaScript Client

The algolia_search theme helper adds the <u>Algolia search client</u> to your pages.

```
1 <%- algolia_search() %>
```

Renders as:

```
1 <script src="/assets/algolia/algoliasearchLite.min.js" async></script>
```

Configure Algolia JavaScript Client

You can make your index configuration available to your page and client-side scripts by adding the algolia_config() hexo helper in the <head> of your document.

```
1 <%- algolia_search_config() %>
```

Once done, you can retrieve Algolia configuration by querying the data_attribute of the algolia:search meta tag.

```
const algoliaConfig =
  document.querySelector('meta[property="algolia:search"]').dataset;

const client = algoliasearch(algoliaConfig.applicationId,
  algoliaConfig.apiKey);

const index = client.initIndex(algoliaConfig.indexName);
```

Display Search Results

It is now up to you to use the aforementioned example to trigger a search and display the results in your page.

If you need some help, have a look at the <u>search client doc</u> and <u>the tutorials</u>.

Indexing Content

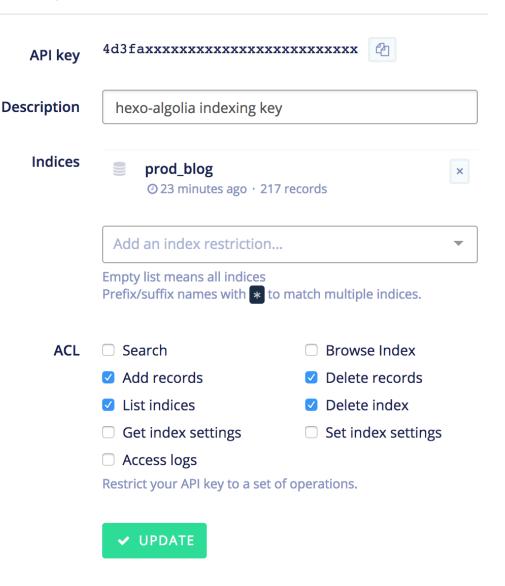
Content is indexed with the help of the hexo algolia command.

```
1 $ ./node_modules/.bin/hexo algolia
```

API Key

A separate API Key must be provided as an **environment variable** named HEXO_ALGOLIA_INDEXING_KEY. Create it with **these limited write access** permissions: Add records, Delete records, List indices, Delete index.

```
$ export HEXO_ALGOLIA_INDEXING_KEY=...
2 $ ./node_modules/.bin/hexo algolia
```



X

Usage

```
$ ./node_modules/.bin/hexo help algolia
2
   Usage: ./node_modules/.bin/hexo algolia
   Description:
   Index your content in Algolia Search API
   Options:
     --dry-run
                     Does not push content to Algolia (default: false).
      --flush
                      Resets the Algolia index before starting the indexation
    (default: false).
10
     --indexing-key An algolia API key with add/delete records permissions.
11
              It should be different than the search-only API key configured in
   _config.yml.
12
     --layouts
                     A comma-separated list of page layouts to index (default:
    "page").
```

Security Concerns

Never store your *Admin API Key* as apikey in the _config.yml file: it would give full control of your Algolia index to others and you don't want to face the consequences.

<u>Please read Algolia Security guide thoroughly</u> if you need some more informations about this.

License

MIT