



# Vue实战

打造属于你的博客发布系统

@ Jrain

# About me



Jrain (/dʒ reɪn/)

华南师范大学光信息科学与技术

招联消费金融有限公司

Hiphop dancer



# Introduction

教你从**零**开始打造一款**博客发布系统**



# Basic Function

- 一个markdown文件代表一个页面（通过 [vue-markdown-loader](#)实现）
- 单页应用，自动根据文章列表来维护路由
- 提供标签分类，选择不同标签即可过滤出对应的文章
- 基于github-page，通过简单的命令即可部署

# Base on Markdown

Markdown是一种可以使用普通文本编辑器编写的标记语言，通过简单的标记语法，它可以使普通文本内容具有一定的格式。

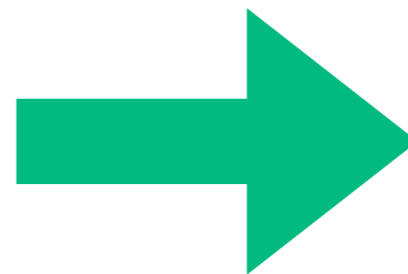
```
![img](../imgs/scion.png)

# SCION
A light tool to generate projects in an easy way.

# Installation
...
npm install scion-cli -g
...
or
...
git clone https://github.com/jrainlau/scion.git

cd scion && npm install

npm link
...
```



```
Scion

SCION

A light tool to generate projects in an easy way.

Installation

```
npm install scion-cli -g
```



or



```
git clone https://github.com/jrainlau/scion.git

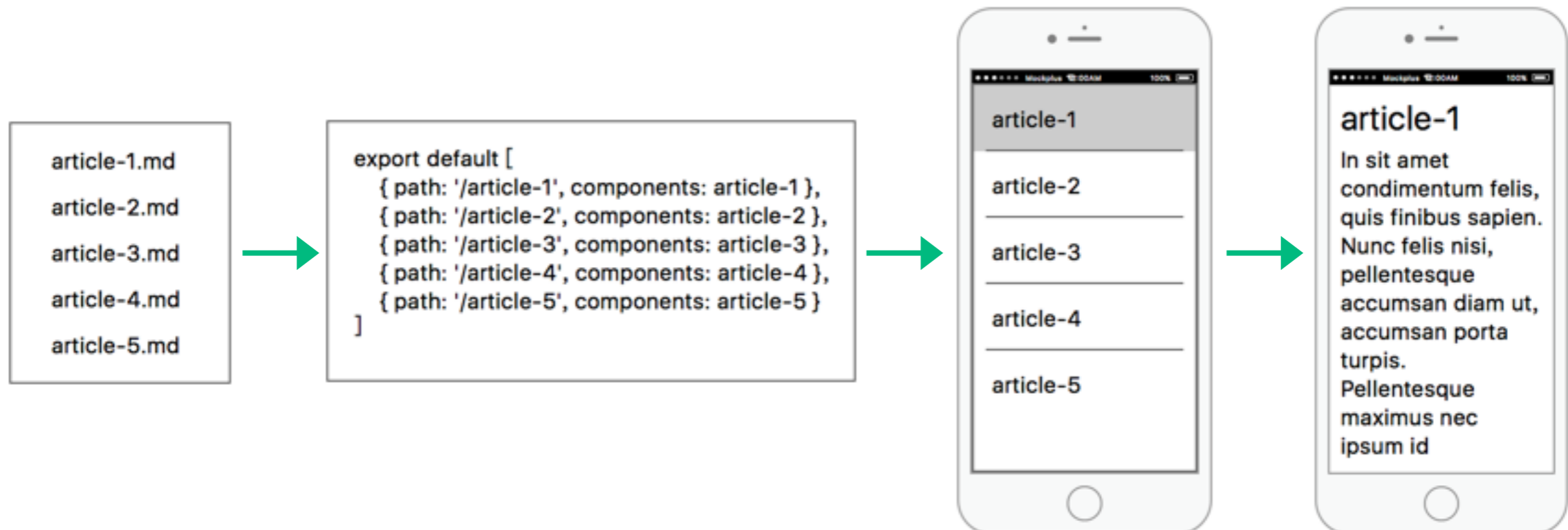
cd scion && npm install

npm link
```


```

# Auto routes

根据markdown文件自动维护路由

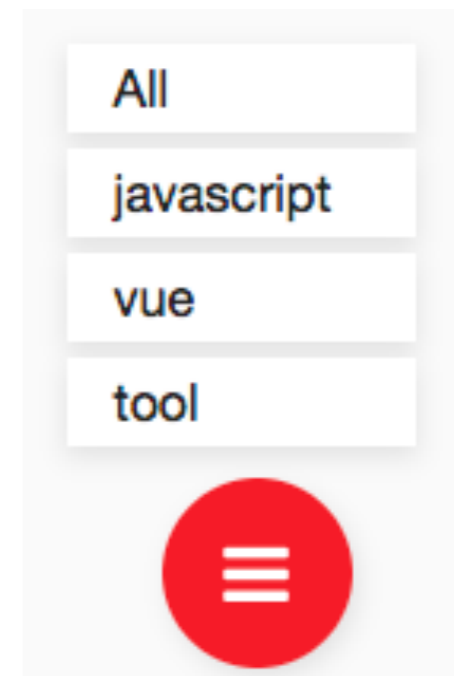
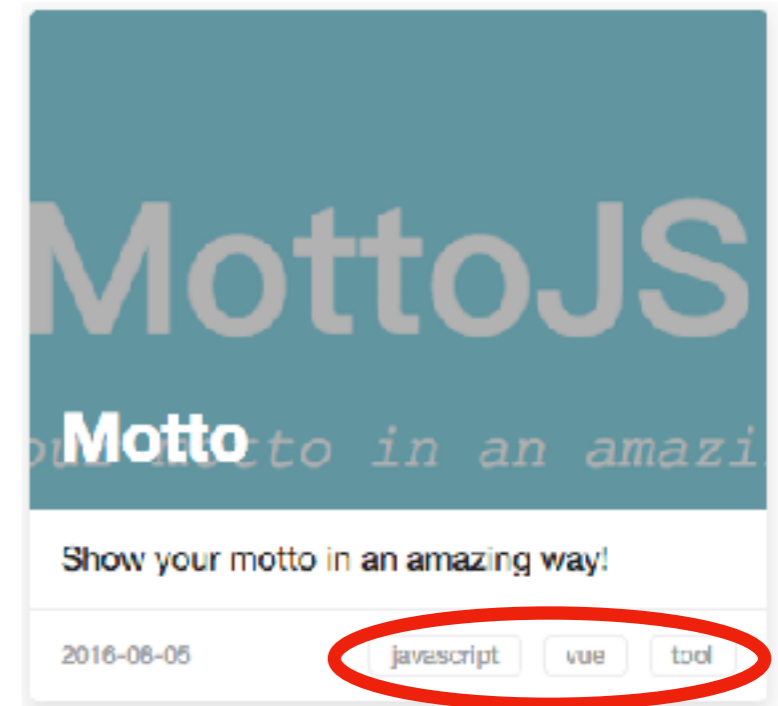


# Tags

每一篇文章都可以被定义多个**标签**,

这些标签会被**自动收集管理**,

点击某个标签则会**过滤**出相符合的文章。

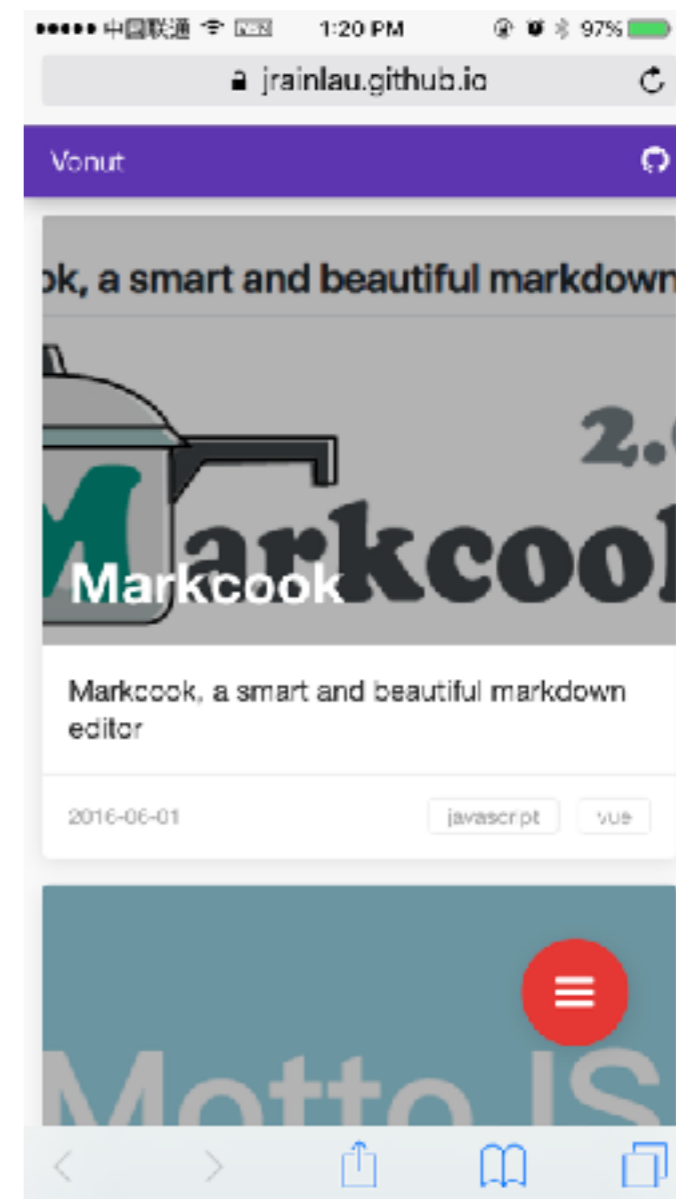


# Github Pages

Github Pages 是Github推出的静态网页托管服务

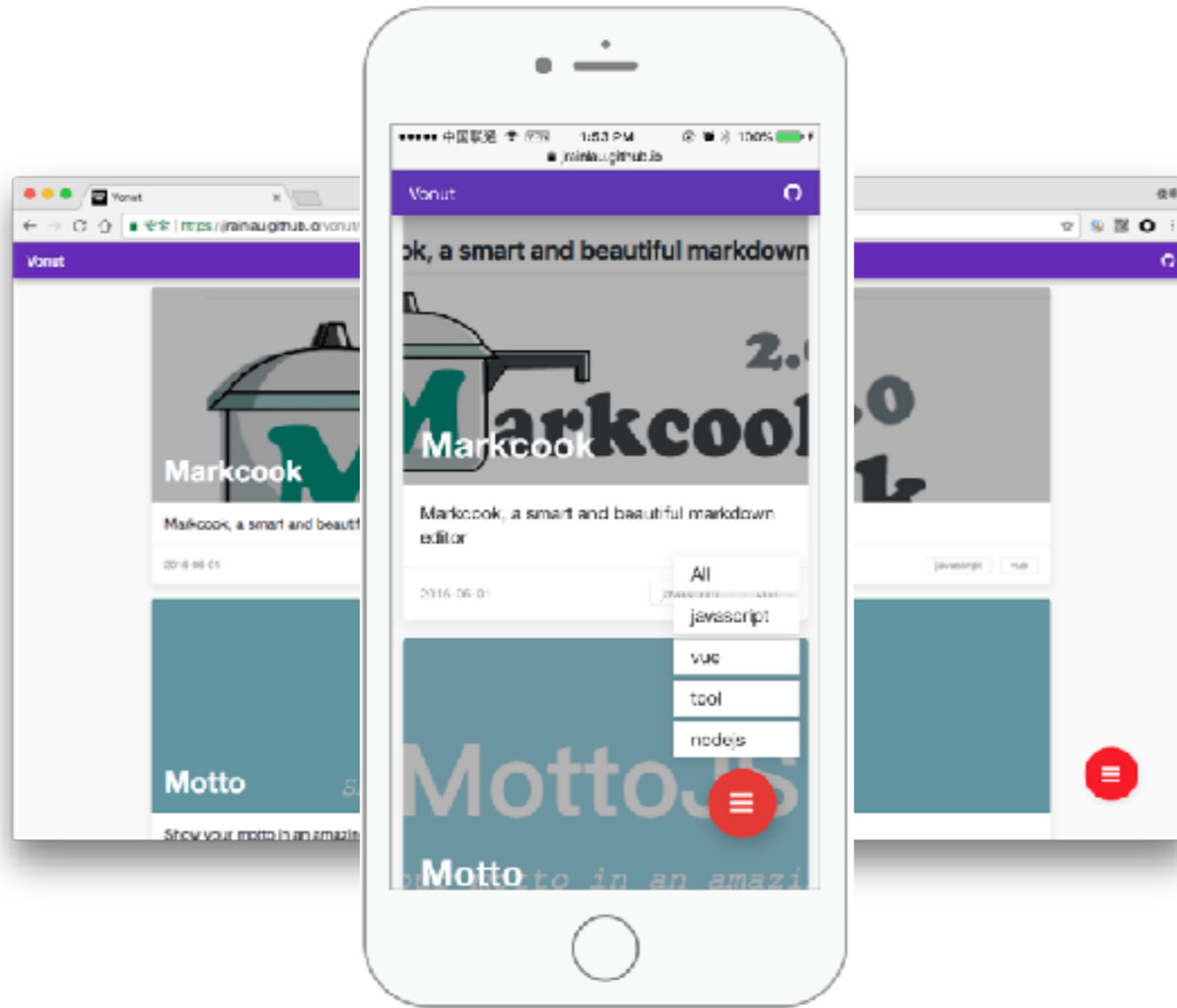
通过简单的设置，Github Pages会自动部署位于你github仓库指定目录下的静态资源文件，

最终通过一个url即可访问这个静态页面。





# UI



遵循Material Design设计风格

使用卡片的形式展现文章列表

通过阴影表达层级关系

响应式页面，移动端优先

# Technique



{less}



# Dev/Prod Environment

```
module.exports = {
  module: {
    rules: [...],
  },
  resolve: {
    extensions: ['.js', '.vue', '.json', '.css', '.less'],
    modules: [resolve('src'), 'node_modules'],
    alias: {
      '~src': resolve('src'),
      '~components': resolve('src/components'),
      '~pages': resolve('src/pages'),
      '~assets': resolve('src/assets'),
      '~store': resolve('src/store'),
      '~articles': resolve('articles')
    }
  }
}
```

Base

```
module.exports = merge(basicConfig, {
  entry: {
    app: './src/main.js'
  },
  output: {
    filename: 'js/[name].js',
    path: resolve('dist'),
    publicPath: '/'
  },
  module: {
    rules: [...],
  },
  devtool: 'inline-source-map',
  devServer: {
    contentBase: resolve('/'),
    compress: true,
    hot: true,
    inline: true,
    publicPath: '/',
    stats: 'minimal',
    host: ip.address()
  },
  plugins: [
    new webpack.HotModuleReplacementPlugin(),
    new webpack.NamedModulesPlugin(),
    new HtmlWebpackPlugin({
      filename: 'index.html',
      template: 'index.html',
      inject: true,
      chunks: ['app']
    })
  ]
})
```

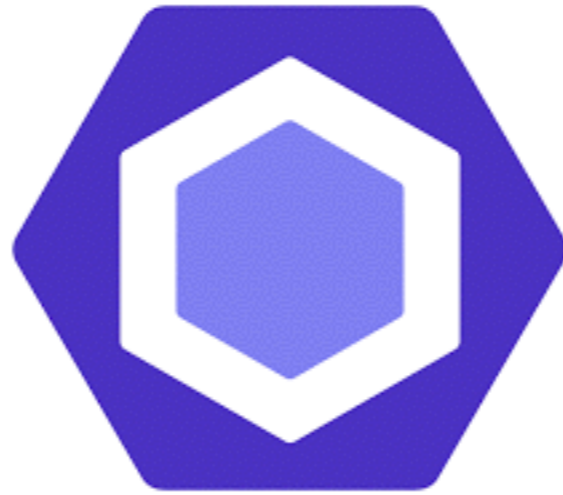
Dev

```
module.exports = merge(basicConfig, {
  entry: [
    'js/app': './src/main.js',
    'js/vendor': [...],
  ],
  devtool: 'source-map',
  output: {
    path: resolve('docs'),
    publicPath: '/output/',
    filename: '[name].[chunkhash:8].js',
    chunkFilename: 'js/[name].[chunkhash:8].js'
  },
  module: {
    rules: [...],
  },
  plugins: [
    new ExtractTextPlugin({
      filename: 'css/[name].[contenthash:8].css',
      allChunks: true
    }),
    new webpack.optimize.CommonsChunkPlugin({
      names: ['js/vendor', 'js/manifest']
    }),
    new HtmlWebpackPlugin({
      filename: 'index.html',
      template: 'index.html',
      inject: true,
      minify: {...},
      chunks: ['js/vendor', 'js/manifest', 'js/app'],
      chunksSortMode: 'dependency'
    }),
    new webpack.LoaderOptionsPlugin({}),
    new webpack.optimize.OccurrenceOrderPlugin(),
    new CleanWebpackPlugin(['docs'], {
      root: resolve('/')
    })
  ]
})
```

Prod

# Coding rules

**Javascript:**



+



**Css:**



# Markdown with JSON

demo-1.md

```
{
  "title": "Demo-1",
  "tags": ["demo", "test"],
  "date": "2016-07-01",
  "desc": "This is demo-1",
  "cover": "demo-1.png"
}

# Demo-1 Title

demo content...
```

demo-2.md

```
{
  "title": "Demo-2",
  "tags": ["demo", "test"],
  "date": "2016-07-01",
  "desc": "This is demo-2",
  "cover": "demo-2.png"
}

# Demo-2 Title

demo content...
```

articles.json

```
{
  "demo-1.md": {
    "title": "Demo-1",
    "tags": [
      "demo",
      "test"
    ],
    "date": "2016-07-01",
    "desc": "This is demo-1",
    "cover": "demo-1.png"
  },
  "demo-2.md": {
    "title": "Demo-2",
    "tags": [
      "demo",
      "test"
    ],
    "date": "2016-07-01",
    "desc": "This is demo-2",
    "cover": "demo-2.png"
  }
}
```

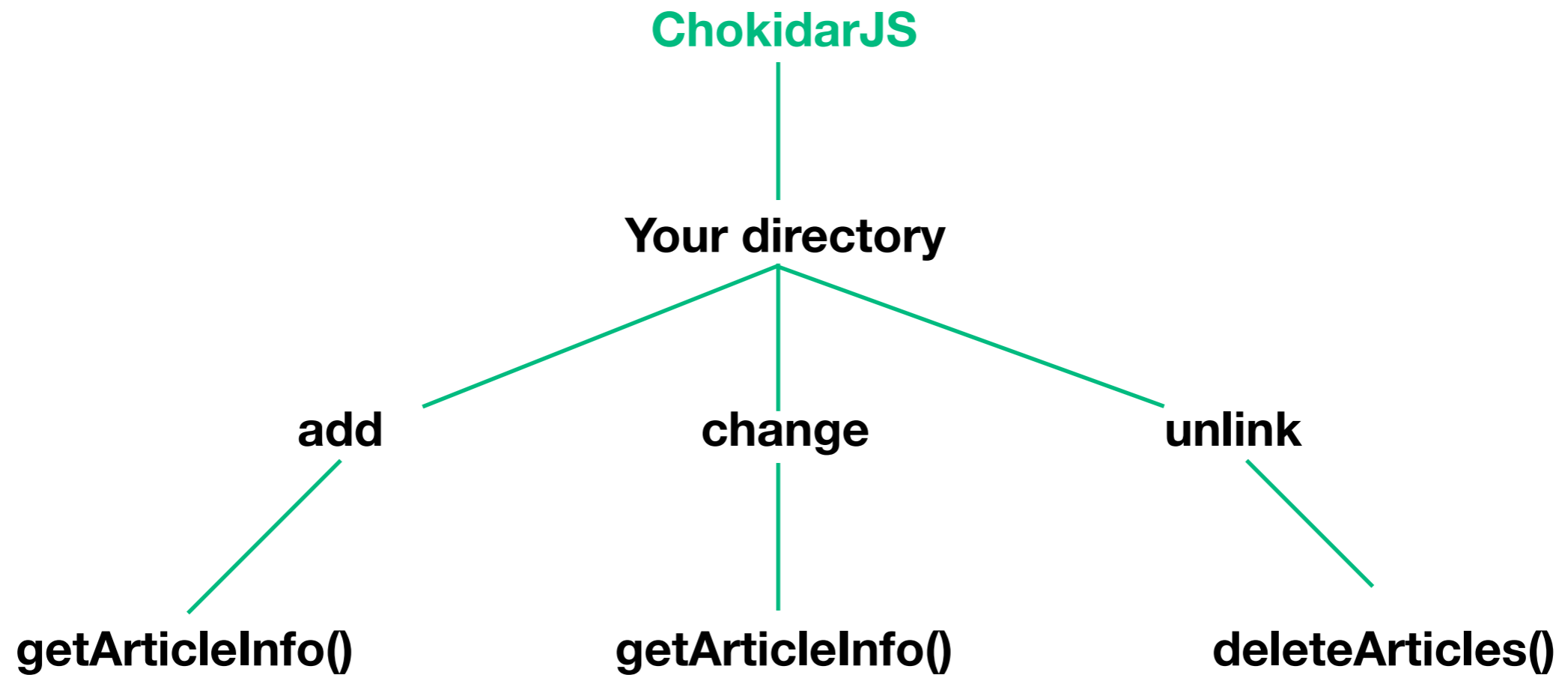
在markdown中使用JSON格式填写文章信息，统一提取到articles.json内，供后续使用。

以后只需要通过CSS把这段JSON隐藏即可。

关键正则表达式：

```
^{\(.*?\)\}/
```

# Watching file changes



# Directory structure



# Routes with Dynamic Import

○ => import ○

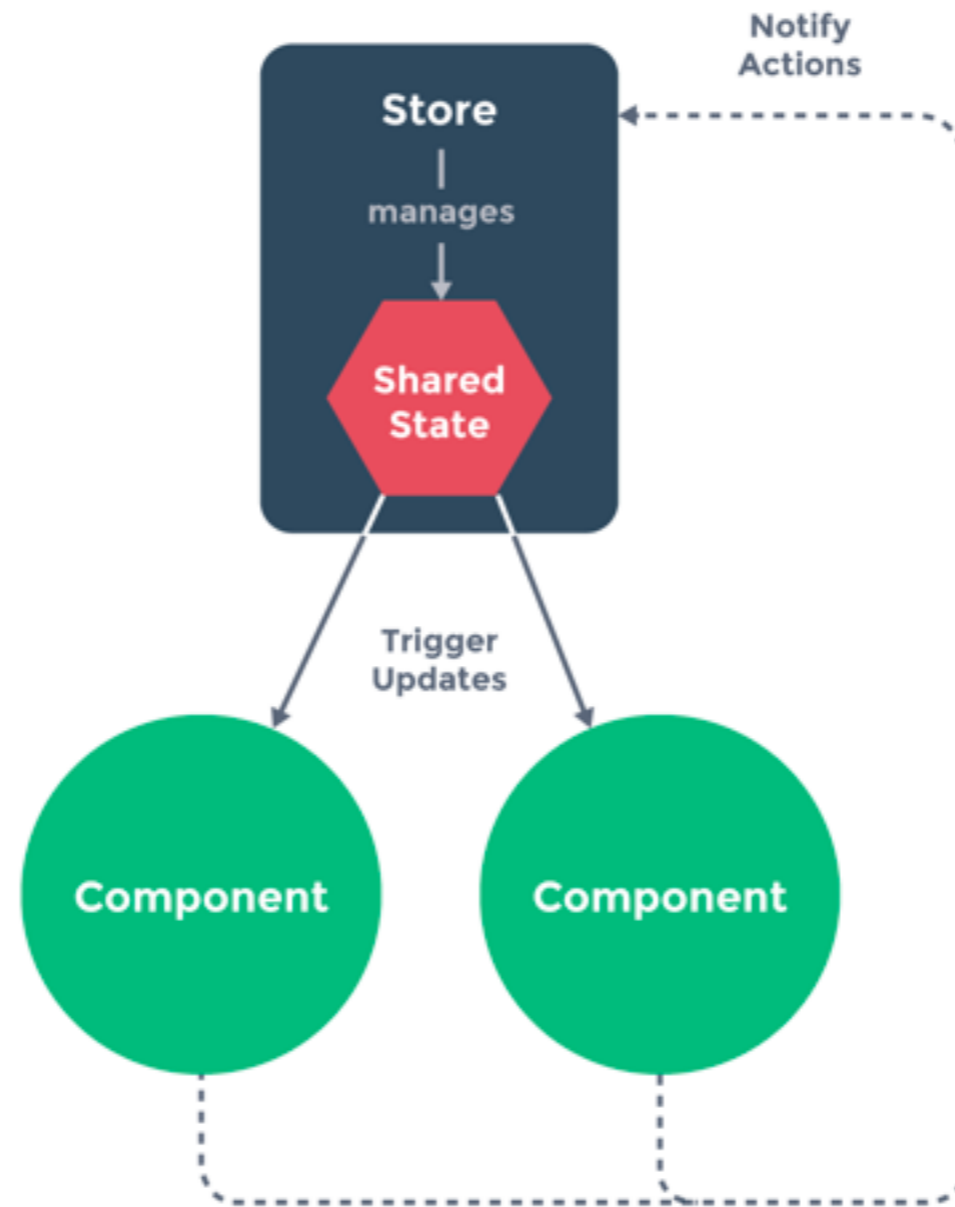
防止打包文件过大

支持代码分割

语法清晰简单



# State management



# Deploying

使用&&符号继发执行npm script命令

```
"npm run build && git add . && git commit -m 'update' && git push"
```

# GitHub pages setting

## GitHub Pages

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

**Source**  
GitHub Pages is currently disabled. Select a source below to enable GitHub Pages for this repository. [Learn more.](#)

None ▾ Save

Select source ×

- master branch  
Use the master branch for GitHub Pages.
- master branch /docs folder**  
Use only the /docs folder for GitHub Pages.
- ✓ None  
Disable GitHub Pages.

theme using the master branch. [Learn more.](#)

## GitHub Pages

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

Your site is ready to be published at <https://jrainlau.github.io/vonut/>.

**Source**  
Your GitHub Pages site is currently being built from the /docs folder in the master branch. [Learn more.](#)

master branch /docs folder ▾ Save

# What's more...

- 添加时间轴
- 添加评论系统
- 解耦工程与博客文章，参考hexo的做法
- 添加换肤功能
- 服务端渲染
- PWA
- ...

**“Thanks for watching”**

*@ Jrain*