

Building the “Front Half” of Your App

(With Grunt)



“Cowboy” Ben Alman
Bocoup

What is Grunt?

What is Grunt?

A task runner.

What is Grunt?

A task runner.

In JavaScript.

Why use Grunt?

Why use Grunt?

Automation.

Why use Grunt?

Automation.

In JavaScript.

But, seriously...

But, seriously...

- Huge community = tons of plugins

But, seriously...

- Huge community = tons of plugins
- Some of the best front-end tooling is written in JavaScript

But, seriously...

- Huge community = tons of plugins
- Some of the best front-end tooling is written in JavaScript
- Node.js works pretty much anywhere

But, seriously...

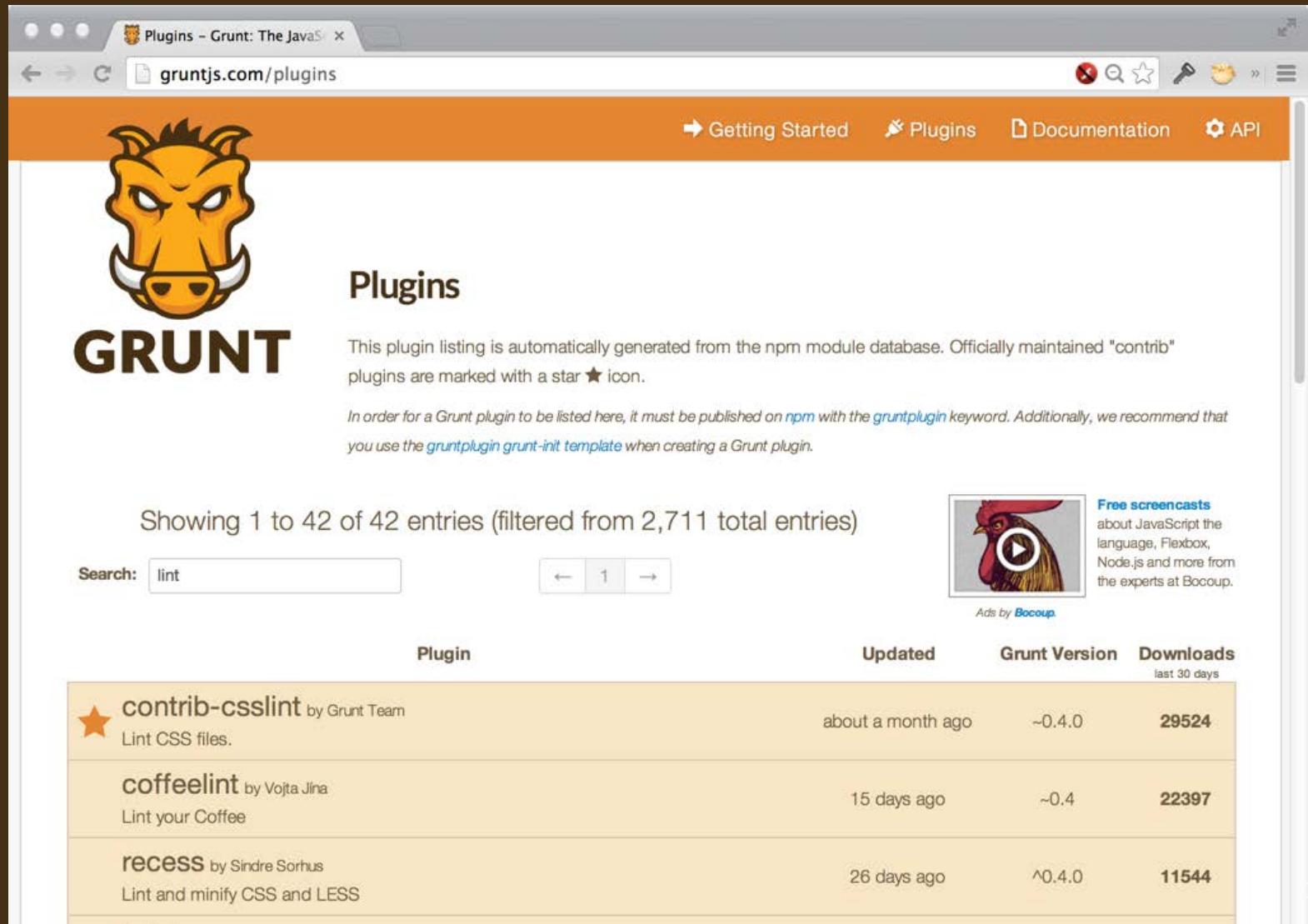
- Huge community = tons of plugins
- Some of the best front-end tooling is written in JavaScript
- Node.js works pretty much anywhere
- Continuous integration

But, seriously...

- Huge community = tons of plugins
- Some of the best front-end tooling is written in JavaScript
- Node.js works pretty much anywhere
- Continuous integration
- IDE support

Huge community = tons of plugins

Huge community = tons of plugins




The screenshot shows the GruntJS website's plugins page. At the top, there's a navigation bar with links for 'Getting Started', 'Plugins', 'Documentation', and 'API'. The main header features the Grunt logo (a stylized orange bull head) and the word 'GRUNT' in large, bold letters. Below the logo, the page title 'Plugins' is displayed. A paragraph explains that the listing is generated from the npm database and that 'contrib' plugins are marked with a star icon. A note specifies that plugins must be published on npm with the 'gruntplugin' keyword and the 'gruntplugin grunt-init' template. A search bar contains the text 'lint', and a pagination control shows '1' of 42 entries. On the right, there's a 'Free screencasts' section with a rooster icon and a play button, advertising content about JavaScript, Flexbox, Node.js, and more from Bocoup. Below this is a table listing plugins with columns for 'Plugin', 'Updated', 'Grunt Version', and 'Downloads (last 30 days)'. The table lists three plugins: 'contrib-csslint' (29524 downloads), 'coffeelint' (22397 downloads), and 'recess' (11544 downloads).

Plugins - Grunt: The JavaScript Task Runner

gruntjs.com/plugins

Getting Started Plugins Documentation API



GRUNT

Plugins


This plugin listing is automatically generated from the npm module database. Officially maintained "contrib" plugins are marked with a star ★ icon.

In order for a Grunt plugin to be listed here, it must be published on [npm](#) with the [gruntplugin](#) keyword. Additionally, we recommend that you use the [gruntplugin grunt-init](#) template when creating a Grunt plugin.

Showing 1 to 42 of 42 entries (filtered from 2,711 total entries)

Search:

← 1 →



Free screencasts
about JavaScript the language, Flexbox, Node.js and more from the experts at Bocoup.

Ads by Bocoup.

Plugin	Updated	Grunt Version	Downloads <small>last 30 days</small>
★ contrib-csslint by Grunt Team Lint CSS files.	about a month ago	~0.4.0	29524
coffeelint by Vojta Jina Lint your Coffee	15 days ago	~0.4	22397
recess by Sindre Sorhus Lint and minify CSS and LESS	26 days ago	^0.4.0	11544

**Some of the best front-end tooling is
written in JavaScript**

Some of the best front-end tooling is written in JavaScript

- JSHint, JSLint, JSCS, Esprima

Some of the best front-end tooling is written in JavaScript

- JSHint, JSLint, JSCS, Esprima
- Stylus, LESS

Some of the best front-end tooling is written in JavaScript

- JSHint, JSLint, JSCS, Esprima
- Stylus, LESS
- CoffeeScript, TypeScript, LiveScript

Some of the best front-end tooling is written in JavaScript

- JSHint, JSLint, JSCS, Esprima
- Stylus, LESS
- CoffeeScript, TypeScript, LiveScript
- jade, handlebars, Lo-dash, Dust

Some of the best front-end tooling is written in JavaScript

- JSHint, JSLint, JSCS, Esprima
- Stylus, LESS
- CoffeeScript, TypeScript, LiveScript
- jade, handlebars, Lo-dash, Dust
- Jasmine, Mocha, QUnit, Nodeunit, Karma

Some of the best front-end tooling is written in JavaScript

- JSHint, JSLint, JSCS, Esprima
- Stylus, LESS
- CoffeeScript, TypeScript, LiveScript
- jade, handlebars, Lo-dash, Dust
- Jasmine, Mocha, QUnit, Nodeunit, Karma
- RequireJS, UglifyJS, webpack, browserify

Node.js works pretty much anywhere

Node.js works pretty much anywhere



Windows Installer (.msi)	32-bit	64-bit
Windows Binary (.exe)	32-bit	64-bit
Mac OS X Installer (.pkg)	Universal	
Mac OS X Binaries (.tar.gz)	32-bit	64-bit
Linux Binaries (.tar.gz)	32-bit	64-bit
SunOS Binaries (.tar.gz)	32-bit	64-bit
Source Code	node-v0.10.26.tar.gz	

Node.js works pretty much anywhere




Windows Installer (.msi)	32-bit	64-bit
Windows Binary (.exe)	32-bit	64-bit
Mac OS X Installer (.pkg)	Universal	
Mac OS X Binaries (.tar.gz)	32-bit	64-bit
Linux Binaries (.tar.gz)	32-bit	64-bit
SunOS Binaries (.tar.gz)	32-bit	64-bit
Source Code	node-v0.10.26.tar.gz	



...and there are Apt / Homebrew / chocolatey packages for Node.js too.

Continuous integration



Continuous integration

- Travis CI  (linux)



Continuous integration

- Travis CI  (linux)
- AppVeyor  (Windows)

Continuous integration

- Travis CI  (linux)
- AppVeyor  (Windows)
- Jenkins (DIY)

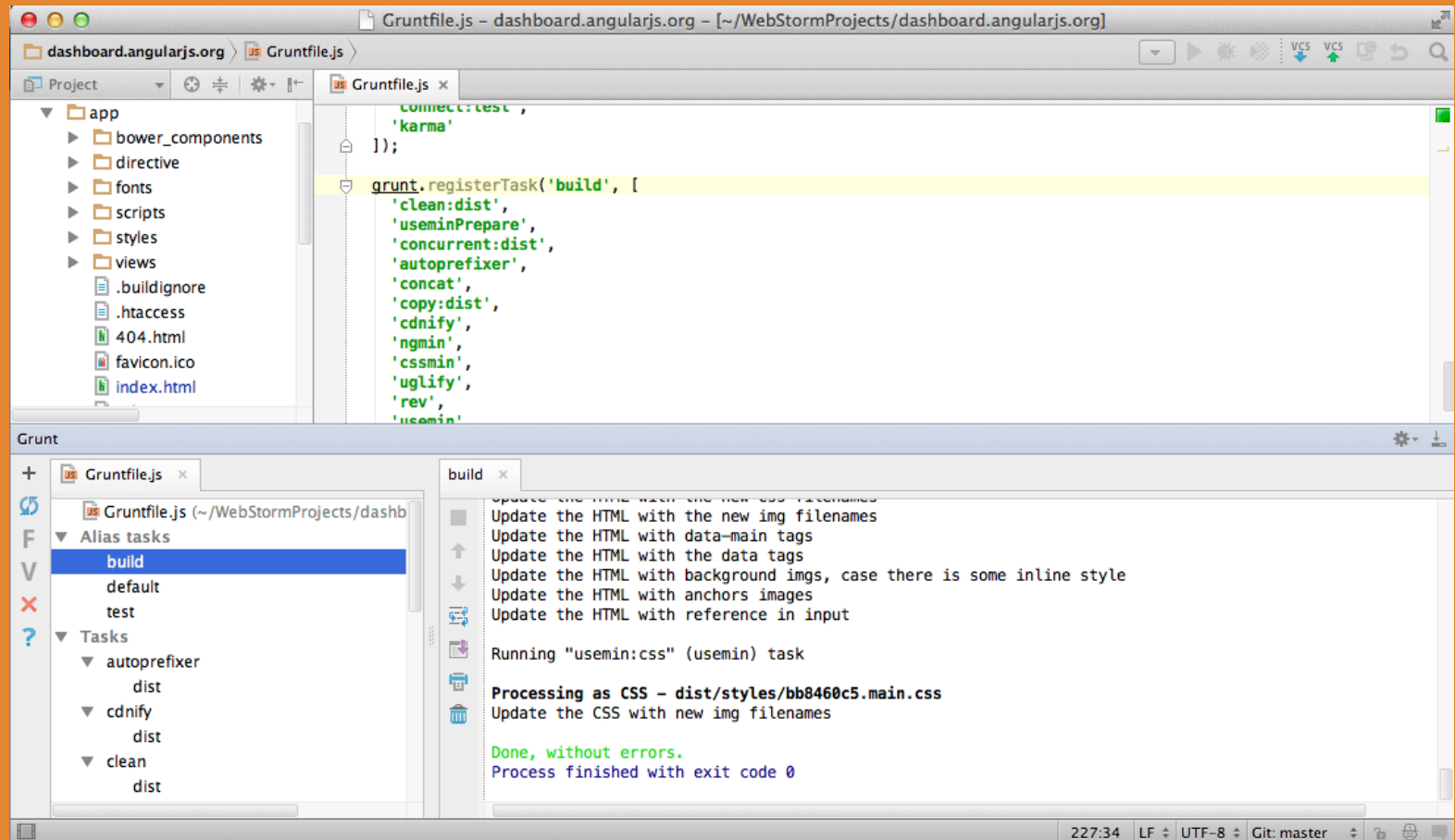
Continuous integration

- Travis CI  (linux)
- AppVeyor  (Windows)
- Jenkins (DIY)
- Or whatever service you want to use, because Grunt uses standard exit codes, just like every other CI tool

IDE support!!!

IDE support!!!

This is pretty new, but WebStorm just added a Grunt console!



What about the “How?”

**You need to know a few
basic things...**

You need to know a few basic things...

- `package.json` (Npm)

You need to know a few basic things...

- package.json (Npm)
- Gruntfile.js (Grunt)

You need to know a few basic things...

- package.json (Npm)
- Gruntfile.js (Grunt)
- The Grunt CLI

package.json (Npm)

package.json (Npm)

This JSON file lists project dependencies, and looks something like this:

```
{
  "name": "my-project-name",
  "version": "0.1.0",
  "devDependencies": {
    "grunt": "^0.4.4",
    "grunt-contrib-jshint": "^0.10.0",
    "grunt-contrib-nodeunit": "^0.3.3",
    "grunt-contrib-uglify": "^0.4.0"
  }
}
```

package.json (Npm)

This JSON file lists project dependencies, and looks something like this:

```
{
  "name": "my-project-name",
  "version": "0.1.0",
  "devDependencies": {
    "grunt": "^0.4.4",
    "grunt-contrib-jshint": "^0.10.0",
    "grunt-contrib-nodeunit": "^0.3.3",
    "grunt-contrib-uglify": "^0.4.0"
  }
}
```

1. Create it with the `npm init` command, a tool like Yeoman, or by hand

package.json (Npm)

This JSON file lists project dependencies, and looks something like this:

```
{
  "name": "my-project-name",
  "version": "0.1.0",
  "devDependencies": {
    "grunt": "^0.4.4",
    "grunt-contrib-jshint": "^0.10.0",
    "grunt-contrib-nodeunit": "^0.3.3",
    "grunt-contrib-uglify": "^0.4.0"
  }
}
```

1. Create it with the `npm init` command, a tool like Yeoman, or by hand
2. Commit it into your project

package.json (Npm)

This JSON file lists project dependencies, and looks something like this:

```
{
  "name": "my-project-name",
  "version": "0.1.0",
  "devDependencies": {
    "grunt": "^0.4.4",
    "grunt-contrib-jshint": "^0.10.0",
    "grunt-contrib-nodeunit": "^0.3.3",
    "grunt-contrib-uglify": "^0.4.0"
  }
}
```

1. Create it with the `npm init` command, a tool like Yeoman, or by hand
2. Commit it into your project
3. Just type `npm install` to install all the dependencies in package.json

package.json (Npm)

This JSON file lists project dependencies, and looks something like this:

```
{
  "name": "my-project-name",
  "version": "0.1.0",
  "devDependencies": {
    "grunt": "^0.4.4",
    "grunt-contrib-jshint": "^0.10.0",
    "grunt-contrib-nodeunit": "^0.3.3",
    "grunt-contrib-uglify": "^0.4.0"
  }
}
```

1. Create it with the `npm init` command, a tool like Yeoman, or by hand
2. Commit it into your project
3. Just type `npm install` to install all the dependencies in package.json
4. Install new dev dependencies with `npm install --save-dev grunt-plugin-name`

Gruntfile.js (Grunt)

Gruntfile.js (Grunt)

This JavaScript file tells Grunt what to do, and looks something like this:

```
module.exports = function(grunt) {

  // Project configuration.
  grunt.initConfig({
    pkg: grunt.file.readJSON('package.json'),
    uglify: {
      options: {
        banner: '/*! ${pkg.name} ${grunt.template.today("yyyy-mm-dd")} */\n'
      },
      build: {
        src: 'src/${pkg.name}.js',
        dest: 'build/${pkg.name}.min.js'
      }
    }
  });

  // Load the plugin that provides the "uglify" task.
  grunt.loadNpmTasks('grunt-contrib-uglify');

  // Default task(s).
  grunt.registerTask('default', ['uglify']);
};
```

The Grunt CLI

The Grunt CLI

- Install this once, globally, per machine with
`npm install -g grunt-cli`

The Grunt CLI

- Install this once, globally, per machine with `npm install -g grunt-cli`
- This puts the `grunt` command into the system path, so you don't have to type `./node_modules/.bin/grunt` or `npm grunt` every time

The Grunt CLI

- Install this once, globally, per machine with `npm install -g grunt-cli`
- This puts the `grunt` command into the system path, so you don't have to type `./node_modules/.bin/grunt` or `npm grunt` every time
- All it really does is finds the project Npm-installed grunt lib, and runs it

Examples

More info...

- gruntjs.com “Getting Started” guide
- [@gruntjs](https://twitter.com/gruntjs) on Twitter
- [#gruntjs](https://irc.freenode.net/#gruntjs) on irc.freenode.net

(Thanks!)

“Cowboy” Ben Alman
Bocoup