

Laravel Elixir for any project

Tom Castleman



Introduction: me



Tom Castleman

Freelance Web Developer

- github.com/tomcastleman
- twitter.com/tomcastleman
- tomcastleman.me



Frontend Tooling: so much choice!



Frontend Tooling: Issues

- So much choice!
- Can quickly become complex
- Rolling your own can be a steep learning curve
- Am I following best practices and making the most of new features?



Vue Components

```
<template>
  <div class="hero">
    {{ message }}
  </div>
</template>
```

```
<script>
export default {
  data () {
    return {
      message: 'Hello World!'
    };
  }
};
</script>
```

```
<style>
.hero {
  background: #eee;
  padding: 1em;
}
</style>
```



Laravel Elixir

- Not elixir [<http://elixir-lang.org/>] !!
- Comes with Laravel PHP framework
- Laravel Elixir provides a clean, fluent API for defining some basic Gulp tasks
- Works out the box with little or no configuration
- we shall focus on soon to be released v6



Just use gulp, duh?!

```
var gulp = require('gulp'),
    sass = require('gulp-ruby-sass'),
    autoprefixer = require('gulp-autoprefixer'),
    rename = require('gulp-rename'),
    notify = require('gulp-notify'),
    live reload = require('gulp-livereload'),
    lr = require('tiny-lr'),
    server = lr();

gulp.task('sass', function() {
  return gulp.src('resources/assets/sass/app.scss')
    .pipe(sass({
      style: 'compressed',
      source map: true
    }))
    .pipe(autoprefixer('last 2 version', 'i.e. 9', 'iOS 6'))
    .pipe(gulp.dest('public/css'))
    .pipe(rename({suffix: '.min'}))
    .pipe(livereload(server))
    .pipe(notify({
      title: "Karani",
      message: "Styles task complete."
    }));
});
```

<https://blog.tighten.co/setting-up-your-first-vuejs-site-using-laravel-elixir-and-vueify>



In Laravel Elixir...

```
var elixir = require('laravel-elixir');  
  
elixir(function(mix) {  
    mix.sass('app.css');  
});
```

<https://blog.tighten.co/setting-up-your-first-vuejs-site-using-laravel-elixir-and-vueify>



Laravel Elixir features: javascript

Module Bundling

- **web pack**
- **browserify**
- **roll up**

Javascript Compilation

- **babel**
- **buble**



Laravel Elixir features: Vue

Vue component compilation

- **vue-loader**
 - Vue.js component loader for Webpack.
 - hot reloading via the webpack-dev-server (not yet supported natively by Elixir)
- **vueify**
 - Browserify transform for Vue.js components, with scoped CSS and component hot-reloading.



Laravel Elixir features: Styles

Style compilation

- **Sass**
- **Less**
- **Stylus**



Style Tasks

Method	Description
<code>mix.sass()</code>	Compile sass, including minification and auto-prefixing
<code>mix.less()</code>	Compile less, including minification and auto-prefixing
<code>mix.stylus()</code>	Compile stylus, including minification and auto-prefixing. Supports any postStylus plugins
<code>mix.styles()</code>	Concatenate and minify css files



Script Tasks

Method	Description
<code>mix.webpack()</code>	Transpile ES2015 into single entry point. Minifies. Source Maps. Supports custom config.
<code>mix.browserify()</code>	Transpile ES2015 into single entry point. Minify. Source Maps. Supports custom options
<code>mix.rollup()</code>	Next-Gen module bundler for your ES2015 modules! Replaces <code>mix.babel()</code> in pre Elixir v6
<code>mix.coffee()</code>	Compile and minify
<code>mix.scripts()</code>	Concatenate and minify
<code>mix.scriptsIn()</code>	Concatenate and minify



File Tasks

Method

Description

`mix.copy()`

Copy files and directories

`mix.combine()`

Concatenate files

`mix.version()`

Create new hashed file and add to manifest



Testing Tasks

Method	Description
<code>mix.browserSync()</code>	Fully working Browsersync config out-the-box including auto refresh of changed files.
<code>mix.phpUnit()</code>	Trigger your entire PHPUnit test suite on file changes
<code>mix.phpSpec()</code>	Trigger your entire PHPUnit test suite on file changes
<code>mix.tdd()</code>	Wrapper for phpUnit and phpSpec



Live Code Examples!



Receiving data from backend

- JSON encoded string through props
- AJAX using vue-resource plugin



Questions?



Feedback



<https://joind.in/talk/32dc8>

