



COMPOSER IN DRUPAL 8 MODULES

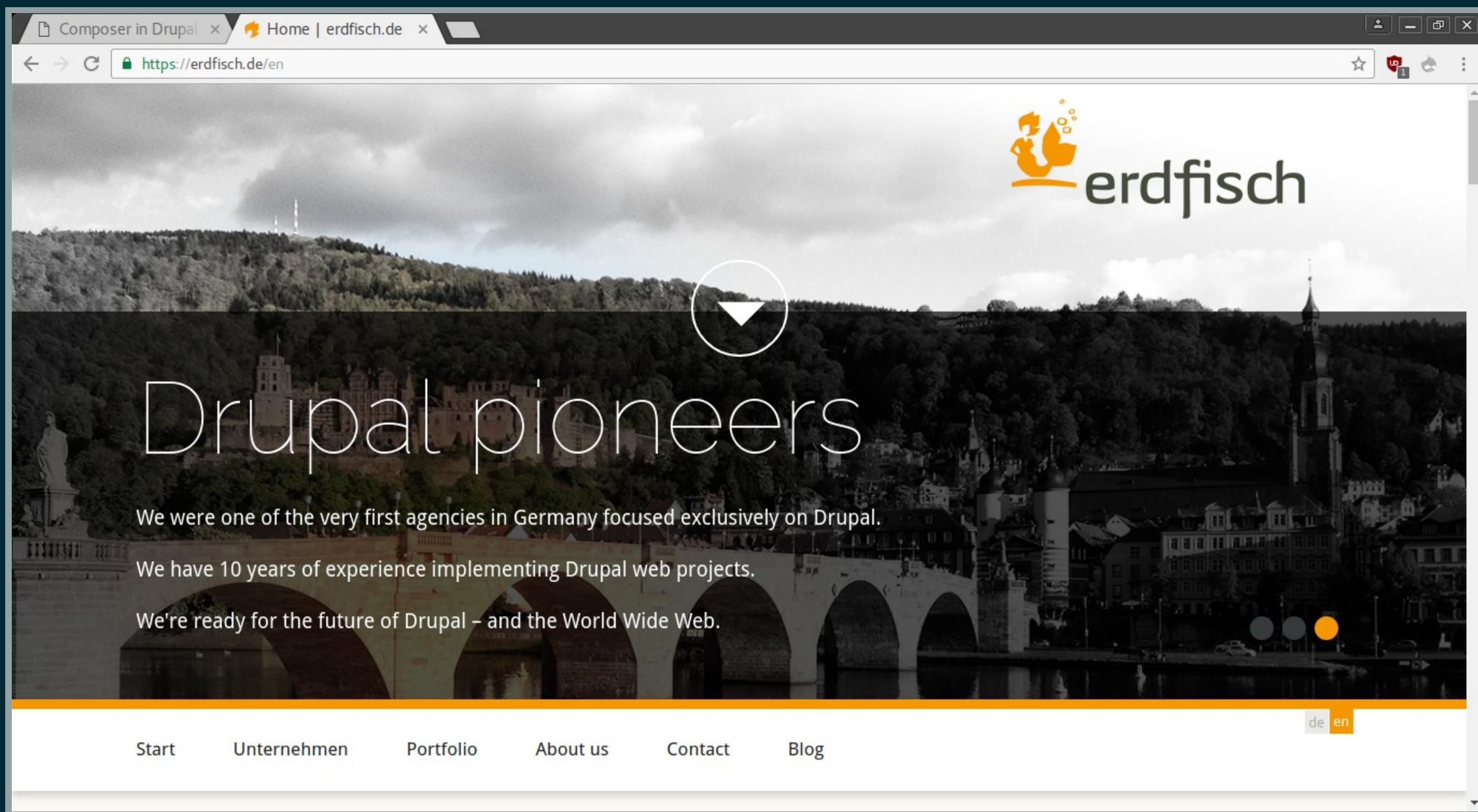


JOIN THE PHP COMMUNITY

Michael Lenahan

#DCMuc16

Drupal Developer at erdfisch in Heidelberg, Germany
michaellenahan on drupal.org and github
@mcaleaa on twitter



#DCMuc16

The slides are here: github.com/michaellenahan/using-composer-drupal-8-modules

A screenshot of a GitHub repository page. The repository name is `michaellenahan / using-composer-drupal-8-modules`. The page shows basic statistics: 6 commits, 1 branch, 0 releases, 1 contributor, and a license of GPL-3.0. The commit history lists four commits: "Michael Lenahan Remove backup talk" (latest), "reveal.js Remove backup talk", "LICENSE Initial commit", and "README.md Update README". Below the commit history, there is a large heading **using-composer-drupal-8-modules** and a note: "These are the slides for a talk given at Drupalcamp Munich on Saturday 3 December 2016: <http://dcmuc16.drupalcamp.de/sessions/using-composer-drupal-8-modules>".

This repository Search Pull requests Issues Gist

michaellenahan / **using-composer-drupal-8-modules** Unwatch 1 Star 0 Fork 0

Code Issues 0 Pull requests 0 Projects 0 Wiki Pulse Graphs Settings

These are the slides for a talk given at Drupalcamp Munich on Saturday 3 December 2016. <http://dcmuc16.drupalcamp.de/sessions/using-composer-drupal-8-modules> — Edit

6 commits 1 branch 0 releases 1 contributor GPL-3.0

Branch: master New pull request Create new file Upload files Find file Clone or download

Michael Lenahan Remove backup talk Latest commit 7c8233a 2 minutes ago

reveal.js Remove backup talk 2 minutes ago

LICENSE Initial commit 31 minutes ago

README.md Update README 3 minutes ago

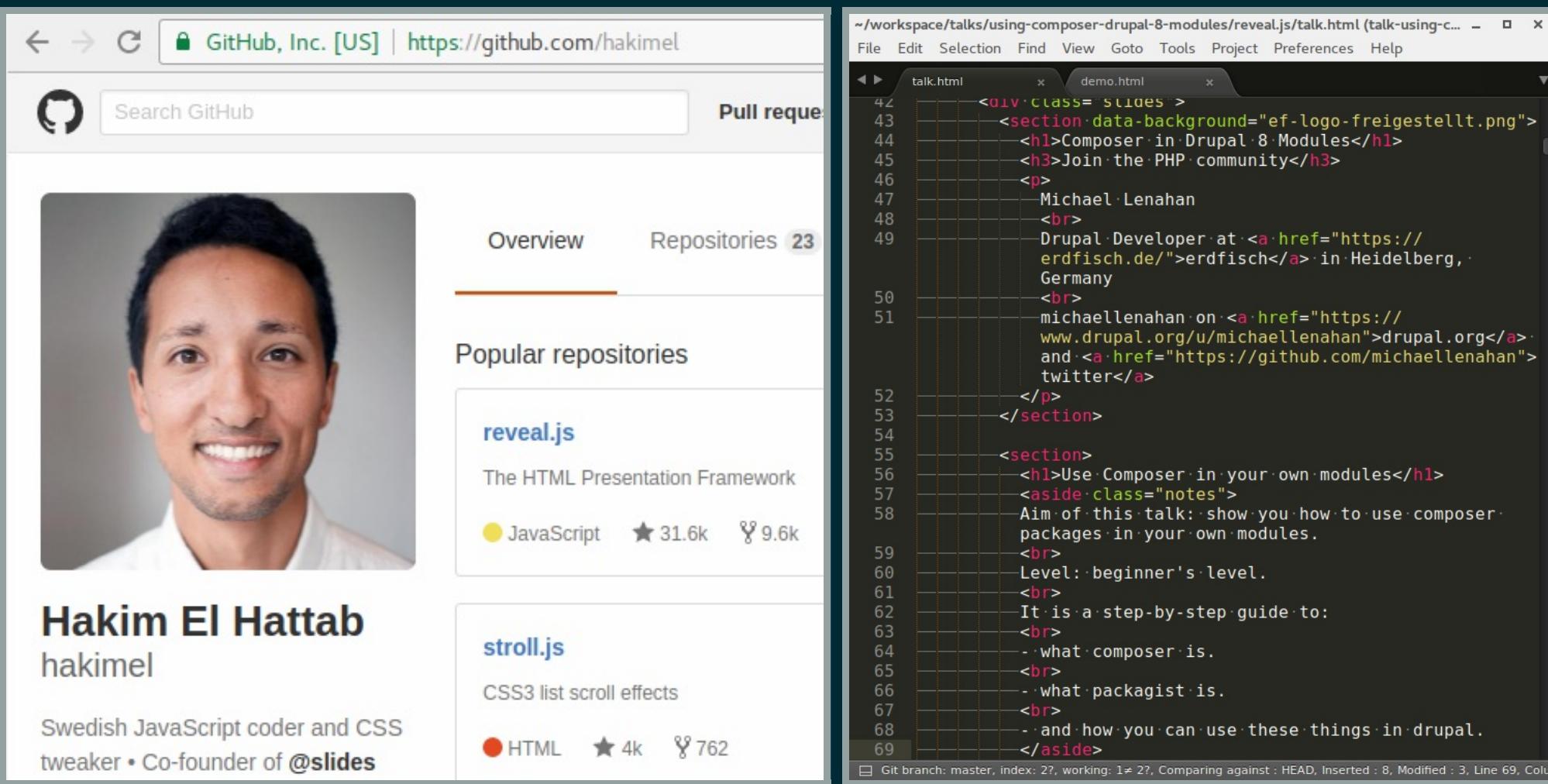
README.md

using-composer-drupal-8-modules

These are the slides for a talk given at Drupalcamp Munich on Saturday 3 December 2016:

#DCMuc16

THESE SLIDES WERE CREATED USING REVEAL.JS



The image shows two side-by-side screenshots. On the left is a screenshot of Hakim El Hattab's GitHub profile page at <https://github.com/hakimel>. It features a large portrait photo of Hakim, his name and handle "Hakim El Hattab hakimel" in bold, and a brief bio stating he is a "Swedish JavaScript coder and CSS tweaker • Co-founder of @slides". Below this are two repository cards: "reveal.js" (The HTML Presentation Framework, 31.6k stars, 9.6k forks) and "stroll.js" (CSS3 list scroll effects, 4k stars, 762 forks). On the right is a screenshot of a code editor showing the source code for a reveal.js presentation. The file "talk.html" contains the slide content, and "demo.html" is shown in the preview pane. The code includes sections for "Composer in Drupal 8 Modules" and "Use Composer in your own modules", along with notes about packagist and Drupal.

```
<div><h1>Composer in Drupal 8 Modules</h1>
<h3>Join the PHP community</h3>
<p>Michael Lenahan<br>
Drupal Developer at <a href="https://erdfisch.de/">erdfisch</a> in Heidelberg, Germany<br>
michaellenahan on <a href="https://www.drupal.org/u/michaellenahan">drupal.org</a> and <a href="https://github.com/michaellenahan">twitter</a></p>
</div>
<section>
<h1>Use Composer in your own modules</h1>
<aside class="notes">
Aim of this talk: show you how to use composer packages in your own modules.
<br>
Level: beginner's level.
<br>
It is a step-by-step guide to:
<br>
- what composer is.
<br>
- what packagist is.
<br>
- and how you can use these things in drupal.
</aside>
```

FAMILY

#DCMuc16



#DCMuc16

A vibrant mural depicting a scene from the children's book "The Mouse and the TV". On the left, several mice are gathered around a television set. One mouse in a white shirt and blue pants is holding a remote control and looking at the screen. Another mouse in a pink dress is sitting next to them. In the background, there are shelves filled with books. The scene is set against a warm orange and yellow gradient.

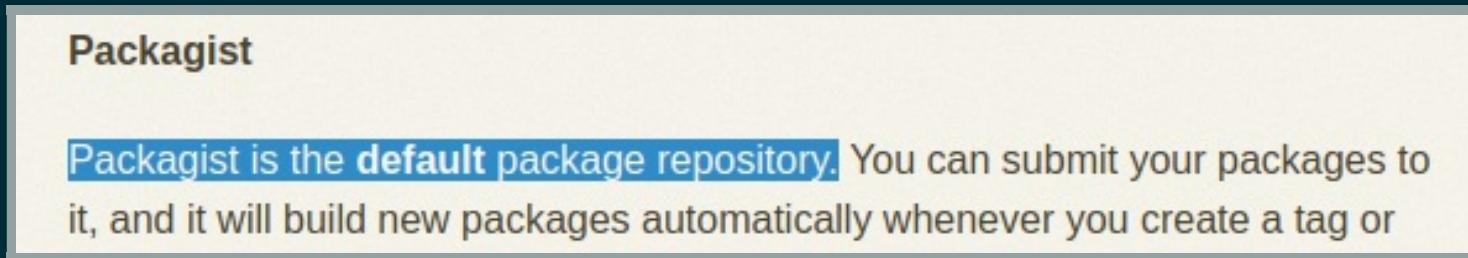
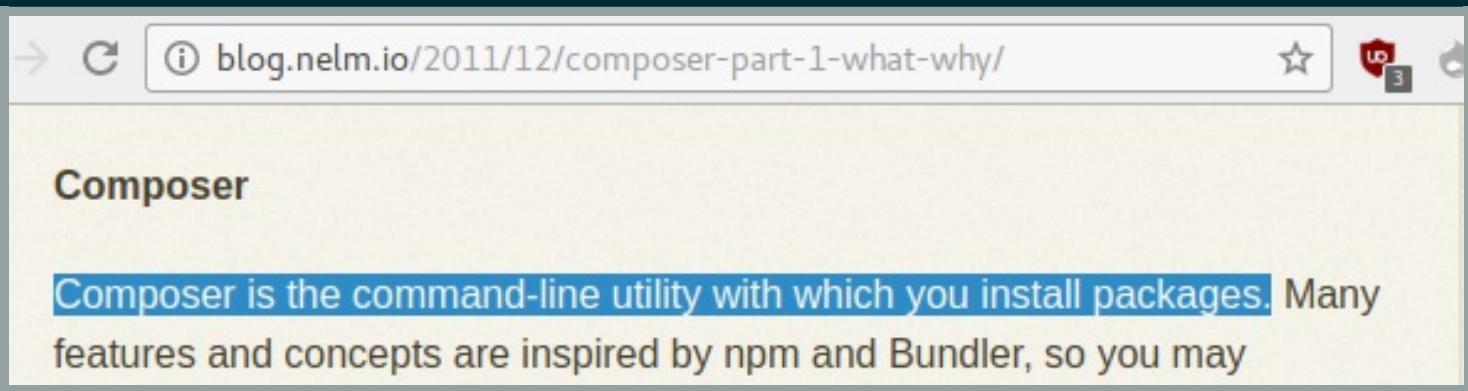
I want to break free-ee

#DCMuc16

USE COMPOSER IN YOUR OWN MODULES

JORDI BOGGIANO, DECEMBER 2011

Composer was created by Nils Adermann and Jordi Boggiano



WHAT IS COMPOSER?

It's a command-line tool

```
michael@t430s: ~
michael@t430s: ~ 150x38

→ ~ composer



Composer version 1.2.2 2016-11-03 17:43:15

Usage:
  command [options] [arguments]

Options:
  -h, --help          Display this help message
  -q, --quiet         Do not output any message
  -V, --version       Display this application version
  --ansi              Force ANSI output
  --no-ansi           Disable ANSI output
  -n, --no-interaction Do not ask any interactive question
  --profile           Display timing and memory usage information
  --no-plugins        Whether to disable plugins.
  -d, --working-dir=WORKING-DIR If specified, use the given directory as working directory.
  -v|vv|vvv, --verbose Increase the verbosity of messages: 1 for normal output, 2 for more verbose output and 3 for debug

Available commands:
  about      Short information about Composer
  archive    Create an archive of this composer package
  browse     Opens the package's repository URL or homepage in your browser.
  clear-cache Clears composer's internal package cache.
  clearcache Clears composer's internal package cache.
  config     Set config options
  create-project Create new project from a package into given directory.
  depends    Shows which packages cause the given package to be installed
  diagnose   Diagnoses the system to identify common errors.
  dump-autoload Dumps the autoloader
  dumpautoload Dumps the autoloader
```

WHAT IS COMPOSER?

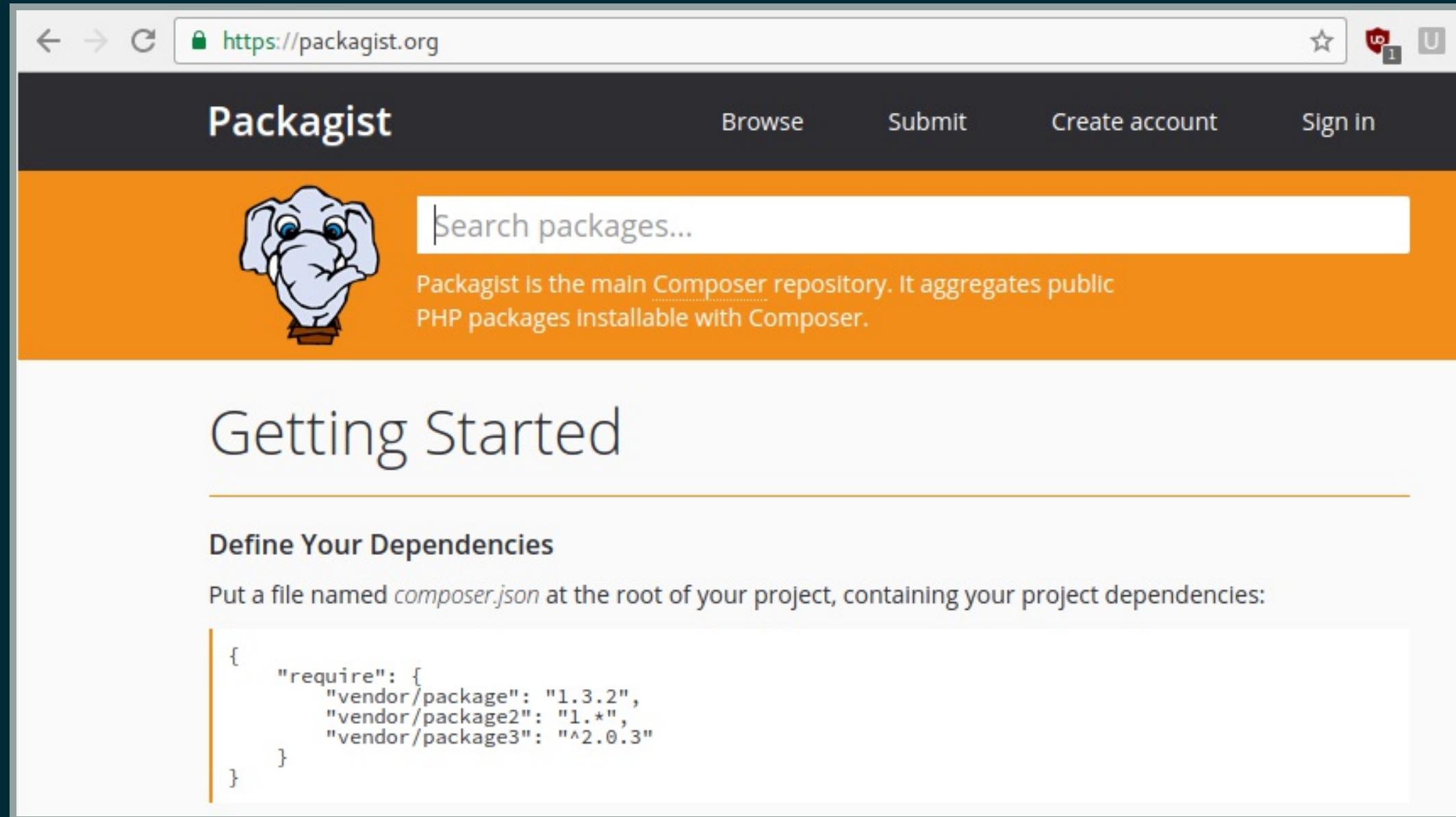
Composer is the command-line utility with which you install packages

```
composer require "drupal/devel"
```

```
drush dl devel
```

WHAT IS PACKAGIST?

Packagist is the *default* package repository

A screenshot of the Packagist.org homepage. The page has a dark header with the title "Packagist" and navigation links for "Browse", "Submit", "Create account", and "Sign in". Below the header is an orange banner featuring a cartoon elephant icon and the text: "Search packages...
Packagist is the main [Composer](#) repository. It aggregates public PHP packages installable with Composer." The main content area is titled "Getting Started" and includes a section on "Define Your Dependencies" with a code snippet for a composer.json file.

Getting Started

Define Your Dependencies

Put a file named `composer.json` at the root of your project, containing your project dependencies:

```
{  
    "require": {  
        "vendor/package": "1.3.2",  
        "vendor/package2": "1.*",  
        "vendor/package3": "^2.0.3"  
    }  
}
```

WHAT IS PACKAGIST?

Packagist is the *default* package repository

The screenshot shows a web browser window with multiple tabs open. The active tab is 'https://www.drupal.org/project/project_module'. The page itself is the 'Download & Extend' section of the Drupal website, specifically the module search page. The URL in the address bar is 'https://www.drupal.org/project/project_module'. The page features the Drupal logo and navigation links for 'Download & Extend', 'Community', 'Documentation', 'Support', 'Jobs', 'Marketplace', and 'About'. A search bar and a user profile icon are also present. The main content area displays a message: '36,030 Modules match your search'. Below this are several filter dropdowns: 'Maintenance status' (set to 'Any'), 'Development status' (set to 'Any'), 'Module categories' (set to 'Any'), 'Core compatibility' (set to 'Any'), and 'Status' (set to 'Full projects'). There is also a 'Search Modules' input field and a 'Sort by' dropdown set to 'Most installed'. A green 'Search' button is located at the bottom of this section. To the right, there are two sidebar boxes: 'New Modules' listing 'Typelt', 'key slapping slippard', 'Text to HTML List', and 'Purecloud' with a 'More new modules' link; and 'Most installed' listing 'Chaos tool suite (ctools)', 'Views', 'Token', and 'Pathauto' with a 'More most installed' link. At the very bottom of the page, a note reads: 'Extend and customize Drupal functionality with contributed modules. If a module doesn't quite do what you want it to do, if you find a bug or have a suggestion, then [join forces](#) and help the module maintainer. Or, share your own by [starting a new module](#)'.

#DCMuc16

"KINDA LIKE" = "KIND OF LIKE"

Composer is "kinda like" drush

Packagist is "kinda like" drupal.org

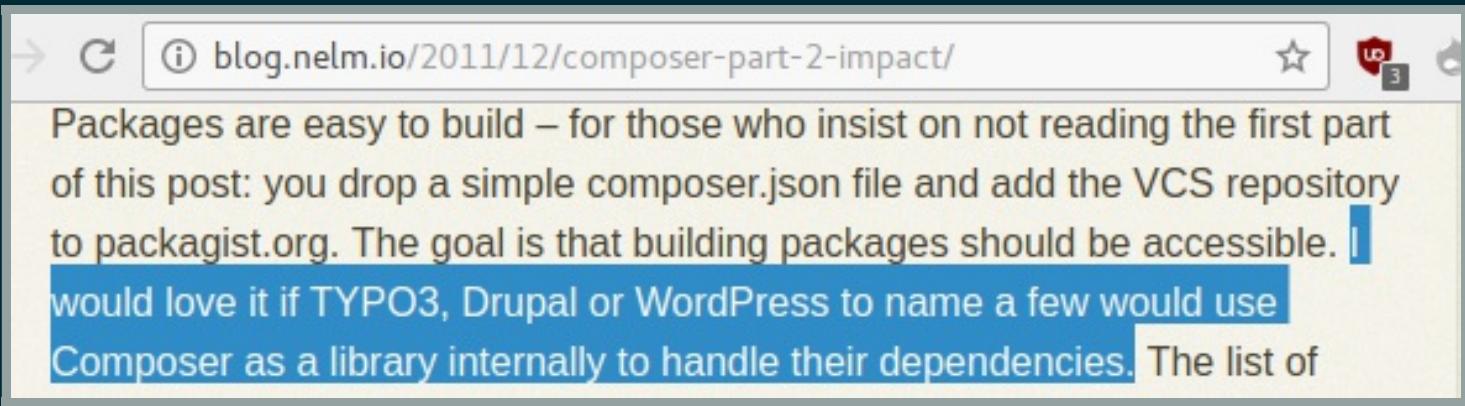
Composer Packages are "kinda like" drupal modules or
themes

DRUPAL SPECIFIC

Drupal-specific code means we don't share with other
frameworks

JORDI BOGGIANO, DECEMBER 2011

Composer was created by Nils Adermann and Jordi Boggiano



EXPLORING PACKAGIST

... and how to find good packages

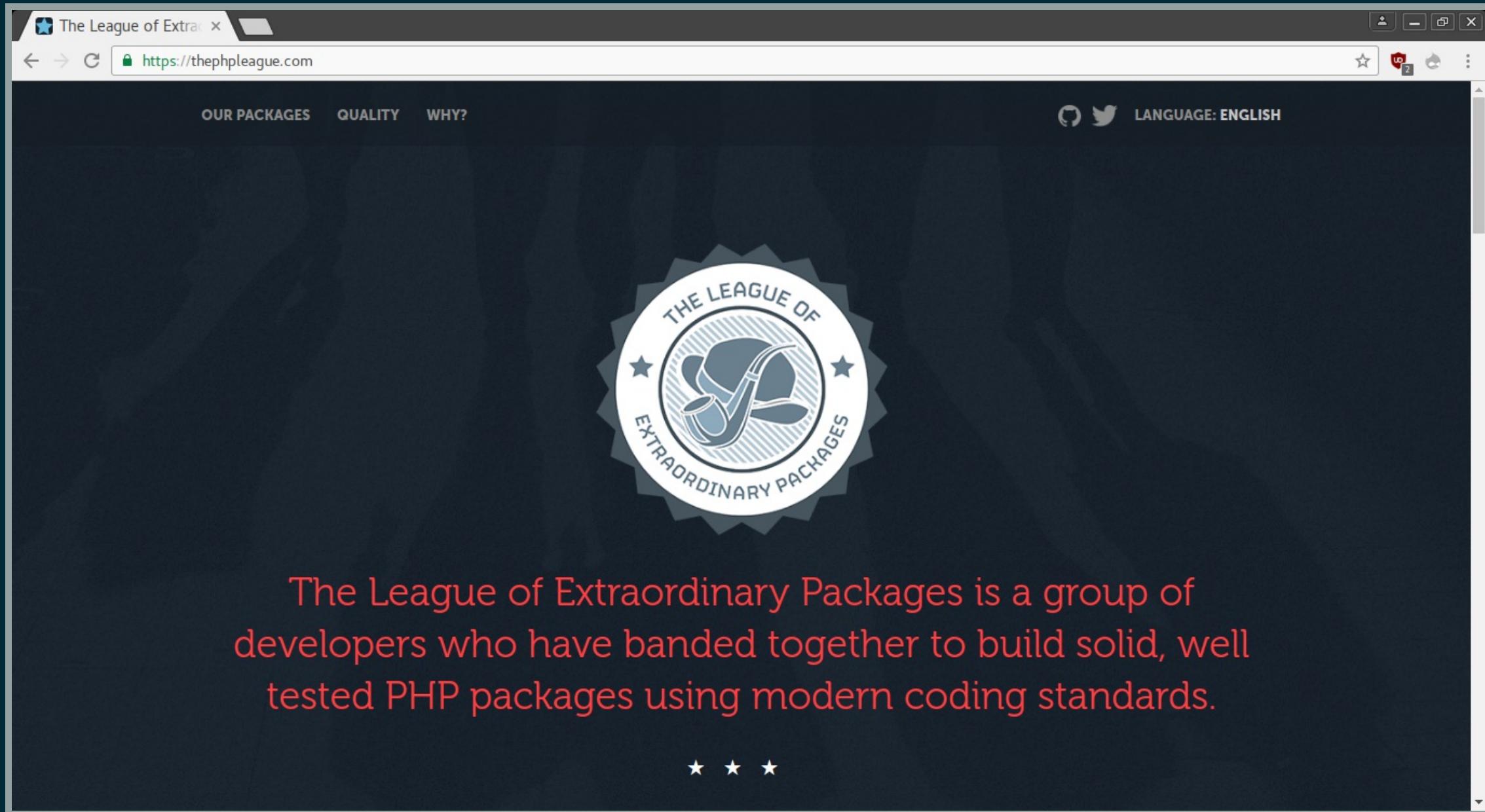
#DCMuc16

A screenshot of a web browser showing the Packagist search results for the package 'zendframework/ldap'. The browser has multiple tabs open, including 'Composer in Drupal', 'Packagist', 'reveal.js - The HTM...', and 'hakimel/reveal.js: Th...'. The main content area shows the Packagist logo and navigation links for 'Browse', 'Submit', 'Create account', and 'Sign in'. A search bar contains the query 'zendframework/ldap'. Below the search bar, a message reads 'Packagist is the main Composer repository. It aggregates public PHP packages installable with Composer.' A large orange button with a cartoon elephant icon and the text 'zendframework/ldap' is visible. The search results list several packages by Zend Framework:

Package	Type	Downloads	Stars
zendframework/zend-ldap	PHP	240 822	15
zendframework/zendframework	PHP	2 930 630	5 532
zendframework/zendxml	PHP	2 295 071	10
zendframework/zendframework1	PHP	1 562 496	343
zendframework/zenddiagnostics	PHP	1 036 226	100

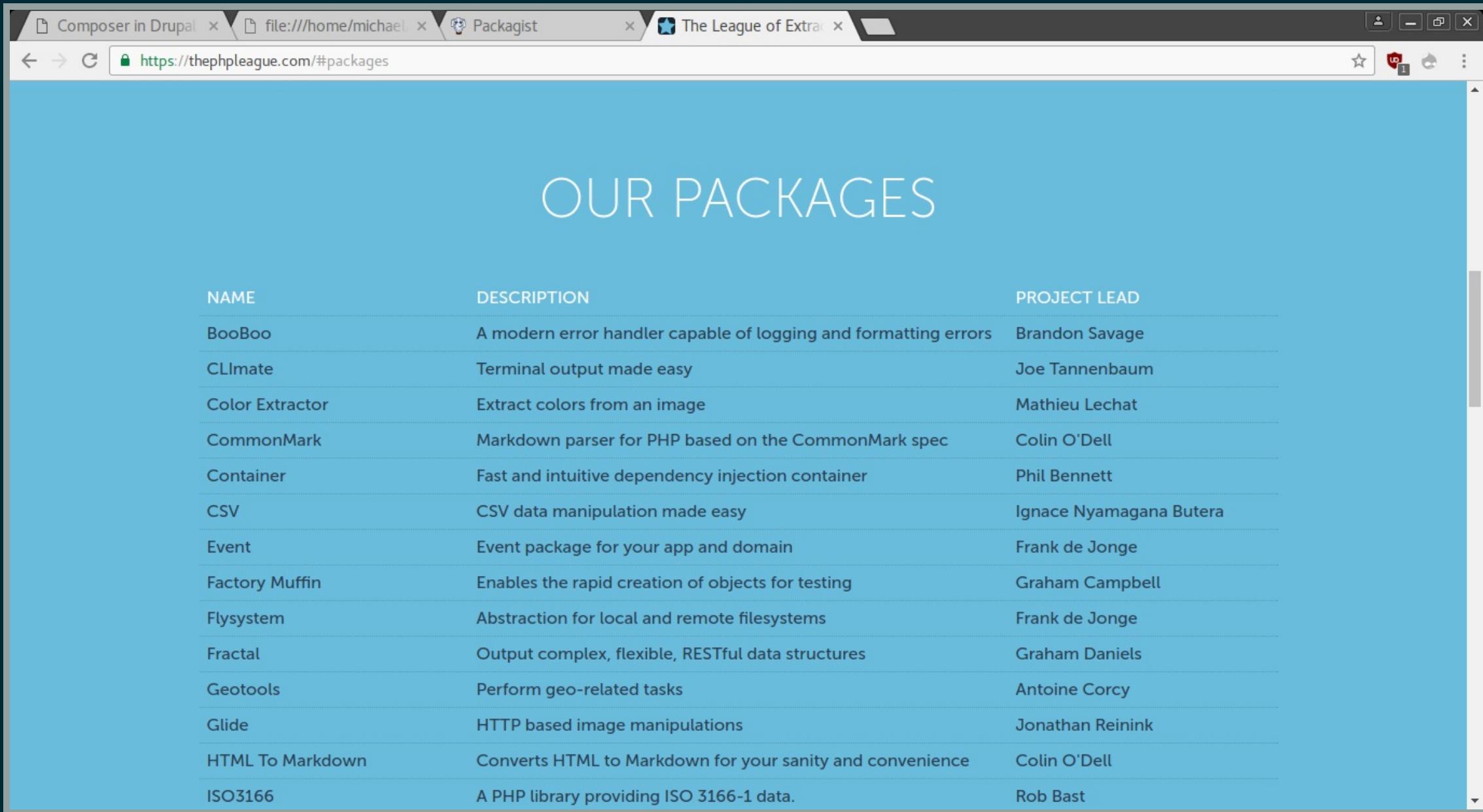
#DCMuc16

THEPHPLEAGUE.COM



#DCMuc16

THEPHPLEAGUE.COM

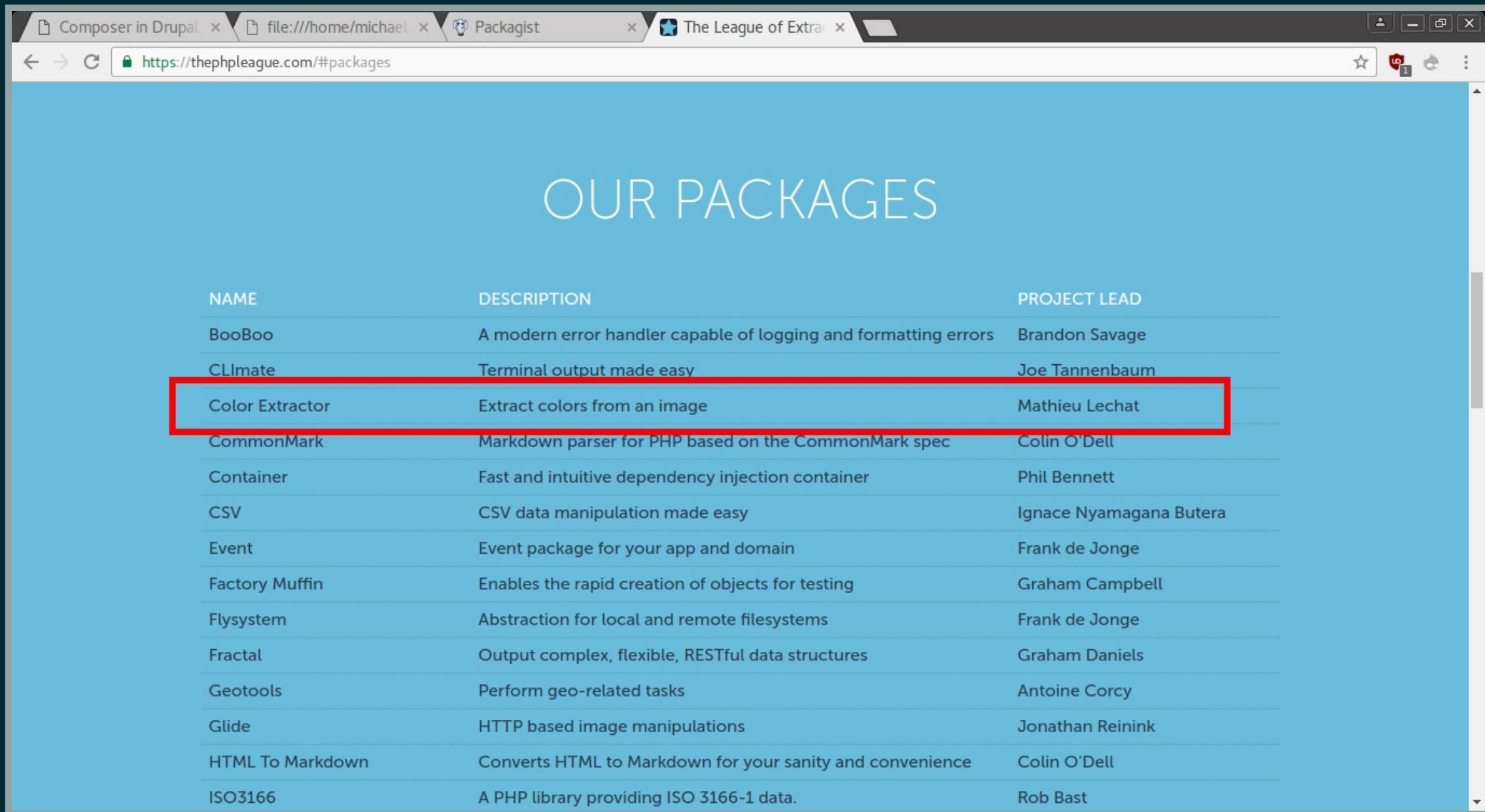


The screenshot shows a web browser window with the URL <https://thephpleague.com/#packages> in the address bar. The page title is "OUR PACKAGES". Below the title is a table listing 15 packages, each with a name, description, and project lead.

NAME	DESCRIPTION	PROJECT LEAD
BooBoo	A modern error handler capable of logging and formatting errors	Brandon Savage
CLimate	Terminal output made easy	Joe Tannenbaum
Color Extractor	Extract colors from an image	Mathieu Lechat
CommonMark	Markdown parser for PHP based on the CommonMark spec	Colin O'Dell
Container	Fast and intuitive dependency injection container	Phil Bennett
CSV	CSV data manipulation made easy	Ignace Nyamagana Butera
Event	Event package for your app and domain	Frank de Jonge
Factory Muffin	Enables the rapid creation of objects for testing	Graham Campbell
Flysystem	Abstraction for local and remote filesystems	Frank de Jonge
Fractal	Output complex, flexible, RESTful data structures	Graham Daniels
Geotools	Perform geo-related tasks	Antoine Corcy
Glide	HTTP based image manipulations	Jonathan Reinink
HTML To Markdown	Converts HTML to Markdown for your sanity and convenience	Colin O'Dell
ISO3166	A PHP library providing ISO 3166-1 data.	Rob Bast

#DCMuc16

THEPHPLEAGUE.COM



OUR PACKAGES

NAME	DESCRIPTION	PROJECT LEAD
BooBoo	A modern error handler capable of logging and formatting errors	Brandon Savage
CLimate	Terminal output made easy	Joe Tannenbaum
Color Extractor	Extract colors from an image	Mathieu Lechat
CommonMark	Markdown parser for PHP based on the CommonMark spec	Colin O'Dell
Container	Fast and intuitive dependency injection container	Phil Bennett
CSV	CSV data manipulation made easy	Ignace Nyamagana Butera
Event	Event package for your app and domain	Frank de Jonge
Factory Muffin	Enables the rapid creation of objects for testing	Graham Campbell
Flysystem	Abstraction for local and remote filesystems	Frank de Jonge
Fractal	Output complex, flexible, RESTful data structures	Graham Daniels
Geotools	Perform geo-related tasks	Antoine Corcy
Glide	HTTP based image manipulations	Jonathan Reinink
HTML To Markdown	Converts HTML to Markdown for your sanity and convenience	Colin O'Dell
ISO3166	A PHP library providing ISO 3166-1 data.	Rob Bast

#DCMuc16

~/workspace/composer-color-extractor/index.php (composer-color-extractor) - Sublime Text

```

File Edit Selection Find View Goto Tools Project Preferences Help
index.php x
1 <?php
2
3 require 'vendor/autoload.php';
4
5 use League\ColorExtractor\Color;
6 use League\ColorExtractor\ColorExtractor;
7 use League\ColorExtractor\Palette;
8
9 // Some images from Creative Commons on Flickr.
10 // https://www.flickr.com/search/?text=&license=2C3C4C5C6C9&sort=interestingness-desc
11 $images = [
12 ];
13
14 $number_of_colors_to_extract = 8;
15
16 foreach ($images as $image) {
17
18     $palette = Palette::fromFilename($image['file']);
19
20     // an extractor is built from a palette
21     $extractor = new ColorExtractor($palette);
22
23     // it defines an extract method which return the most "representative" colors
24     $colors = $extractor->extract($number_of_colors_to_extract);
25
26     // Creative commons attribution.
27     // See: https://wiki.creativecommons.org/wiki/best_practices_for_attribution#This_is_an_ideal_attribution
28     $attribution = "<title>$title<br>by $author<br>is licensed under $license";
29
30     echo "<table><tr><td>";
31     echo "<img src='".$image['file']."' />";
32     echo "<p>$attribution</p>";
33     echo "</td><td>";
34
35     foreach ($colors as $key => $color) {
36         $hex_code = Color::fromIntToHex($color);
37         echo "<h1 style='color: $hex_code'>$key: $hex_code</h1>";
38     }
39
40     echo "</td></tr></table>";
41
42 }
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64

```

composer-color-ext x www.flickr.com/pho x www.flickr.com/pho x

composer-color-extractor.local/

Osaka, Japan - Sunset from Umeda City
by Pedro Szekely
is licensed under CC BY-SA 2.0

Autumn Colours, Landgoed Voorstonden, Brummen, Netherlands - 1032
by Tom Jutte
is licensed under CC BY-NC-ND 2.0

0: #63210B
1: #9E471C
2: #F1AF37
3: #BE6B3F
4: #F9882C
5: #34170F
6: #FFDE48
7: #BC8E50
0: #7DADDE
1: #3017EC
2: #548AD2
3: #B25900
4: #6C7518
5: #710C00
6: #EF9A3F
7: #CEB034

#DCMuc16

THIS IS PLAIN PHP ...

The screenshot illustrates a workflow for extracting colors from images using PHP. On the left, a Sublime Text editor window shows the source code for `index.php`. The code uses the `League\ColorExtractor` library to find the most representative colors from Flickr images. It includes attribution logic for Creative Commons images. On the right, two browser windows show the results. The top window displays a sunset over a city skyline with color hex codes listed on the right. The bottom window shows autumn foliage reflected in water, also with color hex codes listed.

Code Snippet (index.php):

```
1 <?php
2 require 'vendor/autoload.php';
3
4 use League\ColorExtractor\Color;
5 use League\ColorExtractor\ColorExtractor;
6 use League\ColorExtractor\Palette;
7
8 // Some images from Creative Commons on Flickr.
9 // https://www.flickr.com/search/?text=&license=2C3C4C5C6C9&sort=interestingness-desc
10
11 $images = [
12 ];
13
14 $number_of_colors_to_extract = 8;
15
16 foreach ($images as $image) {
17
18     $palette = Palette::fromFilename($image['file']);
19
20     // an extractor is built from a palette
21     $extractor = new ColorExtractor($palette);
22
23     // it defines an extract method which return the most "representative" colors
24     $colors = $extractor->extract($number_of_colors_to_extract);
25
26     // Creative commons attribution.
27     // See: https://wiki.creativecommons.org/wiki/
28     // best practices for attribution#This is an ideal attribution
29     $attribution = "$title<br>by $author<br>is licensed under $license";
30
31     echo "<table><tr><td>";
32     echo "<img src='".$image['file']."' />";
33     echo "<p>$attribution</p>";
34     echo "</td></tr></table>";
35
36 }
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
```

Result 1 (Sunset):

Osaka, Japan - Sunset from Umeda City
by Pedro Szekely
is licensed under CC BY-SA 2.0

0: #63210B
1: #9E471C
2: #F1AF37
3: #BE6B3F
4: #F9882C
5: #34170F
6: #FFDE48
7: #BC8E50

Result 2 (Autumn Foliage):

Autumn Colours, Landgoed Voorstonden, Brummen, Netherlands - 1032
by Tom Jutte
is licensed under CC BY-NC-ND 2.0

0: #7DADDE
1: #3017EC
2: #548AD2
3: #B25900
4: #6C7518
5: #710C00
6: #EF9A3F
7: #CEB034

THIS IS *NOT* DRUPAL ...

```
~/workspace/composer-color-extractor/index.php (composer-color-extractor) - Sublime Text
File Edit Selection Find View Goto Tools Project Preferences Help
index.php x
1 <?php
2 require 'vendor/autoload.php';
3
4 use League\ColorExtractor\Color;
5 use League\ColorExtractor\ColorExtractor;
6 use League\ColorExtractor\Palette;
7
8 // Some images from Creative Commons on Flickr.
9 // https://www.flickr.com/search/?text=&license=2C3C4C5C6C9&sort=interestingness-desc
10
11 $images = [
12 ];
13
14 $number_of_colors_to_extract = 8;
15
16 foreach ($images as $image) {
17
18     $palette = Palette::fromFilename($image['file']);
19
20     // an extractor is built from a palette
21     $extractor = new ColorExtractor($palette);
22
23     // it defines an extract method which return the most "representative" colors
24     $colors = $extractor->extract($number_of_colors_to_extract);
25
26     // Creative commons attribution.
27     // See: https://wiki.creativecommons.org/wiki/
28     // best practices for attribution#This is an ideal attribution
29     $attribution = "$title<br>by $author<br>is licensed under $license";
30
31     echo "<table><tr><td>";
32     echo "<img src='".$image['file']."' />";
33     echo "<p>$attribution</p>";
34     echo "</td></tr></table>";
35
36 }
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
```

The screenshot shows a web browser window with two images and their color palettes. The top image is a sunset over a city skyline with orange and red hues, and the bottom image is a landscape with autumn foliage reflected in water. To the right of each image is a vertical list of color hex codes.

Sunset Image (Top):

- 0: #63210B
- 1: #9E471C
- 2: #F1AF37
- 3: #BE6B3F
- 4: #F9882C
- 5: #34170F
- 6: #FFDE48
- 7: #BC8E50

Autumn Landscape Image (Bottom):

- 0: #7DADDE
- 1: #3017EC
- 2: #548AD2
- 3: #B25900
- 4: #6C7518
- 5: #710C00
- 6: #EF9A3F
- 7: #CEB034

Osaka, Japan - Sunset from Umeda City
by Pedro Szekely
is licensed under CC BY-SA 2.0

Autumn Colours, Landgoed Voorstonden, Brummen, Netherlands - 1032
by Tom Jutte
is licensed under CC BY-NC-ND 2.0

BECAUSE WE ARE ON A DRUPAL HOLIDAY

The image shows a developer's environment with two windows. On the left is a Sublime Text editor window titled "index.php" containing PHP code for extracting colors from images. On the right are two browser windows showing Flickr images with their corresponding color palettes.

Code (Sublime Text):

```
1 <?php
2 require 'vendor/autoload.php';
3
4 use League\ColorExtractor\Color;
5 use League\ColorExtractor\ColorExtractor;
6 use League\ColorExtractor\Palette;
7
8 // Some images from Creative Commons on Flickr.
9 // https://www.flickr.com/search/?text=&license=2%2C3%2C4%2C5%2C6%2C9&sort=interestingness-desc
10
11 $images = [
12 ];
13
14 $number_of_colors_to_extract = 8;
15
16 foreach ($images as $image) {
17
18     $palette = Palette::fromFilename($image['file']);
19
20     // an extractor is built from a palette
21     $extractor = new ColorExtractor($palette);
22
23     // it defines an extract method which return the most "representative" colors
24     $colors = $extractor->extract($number_of_colors_to_extract);
25
26     // Creative commons attribution.
27     // See: https://wiki.creativecommons.org/wiki/
28     // best practices for attribution#This is an ideal attribution
29     $attribution = "$title<br>by $author<br>is licensed under $license";
30
31     echo "<table><tr><td>";
32     echo "<img src='".$image['file']."' />";
33     echo "<p>$attribution</p>";
34     echo "</td></tr></table>";
35 }
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64 }
```

Browser Window 1 (Top): A sunset over a city skyline with mountains in the background. The color palette on the right lists the top 8 colors extracted from the image.

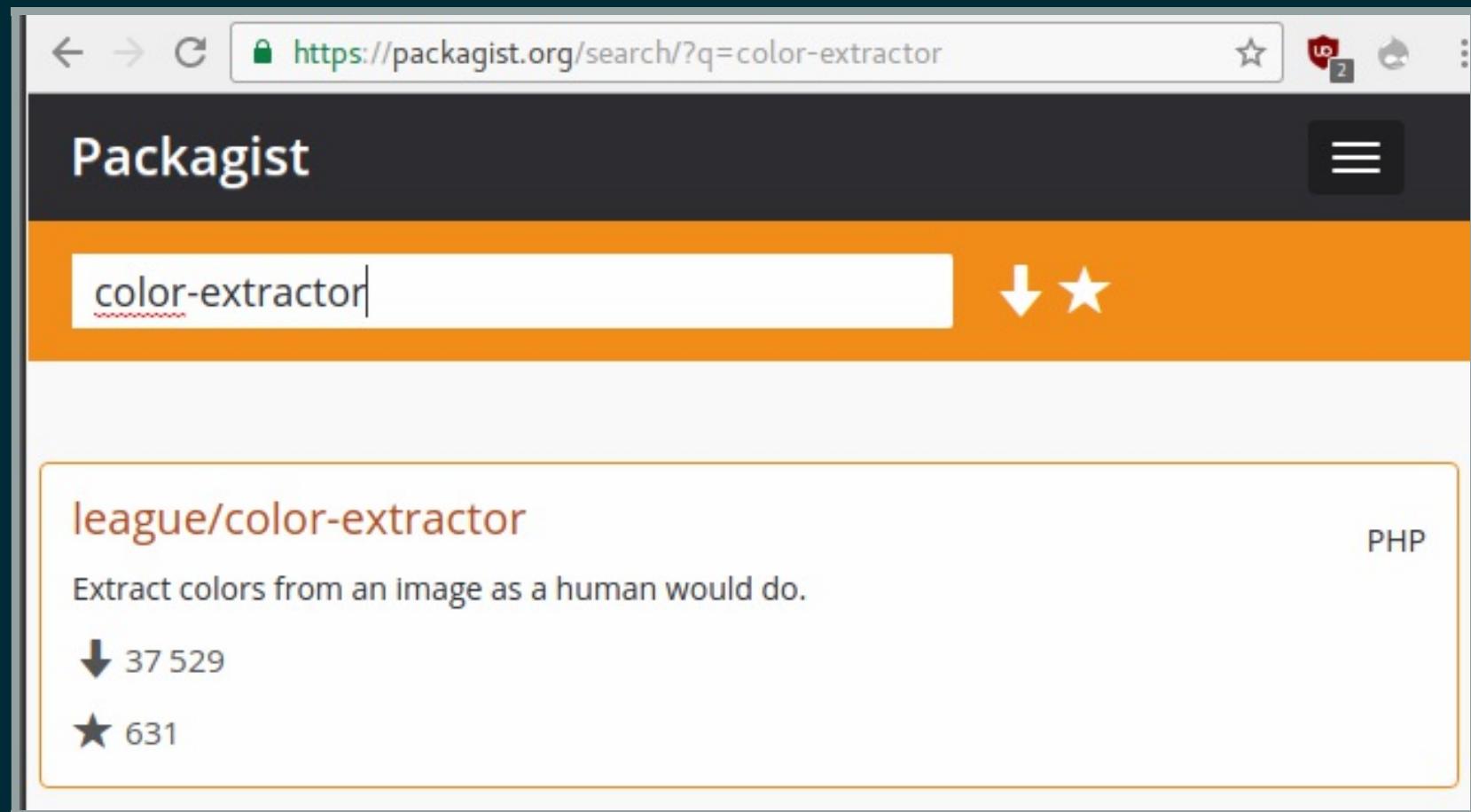
Color Index	Hex Code
0	#63210B
1	#9E471C
2	#F1AF37
3	#BE6B3F
4	#F9882C
5	#34170F
6	#FFDE48
7	#BC8E50

Browser Window 2 (Bottom): A river flowing through a forest with autumn foliage. The color palette on the right lists the top 8 colors extracted from the image.

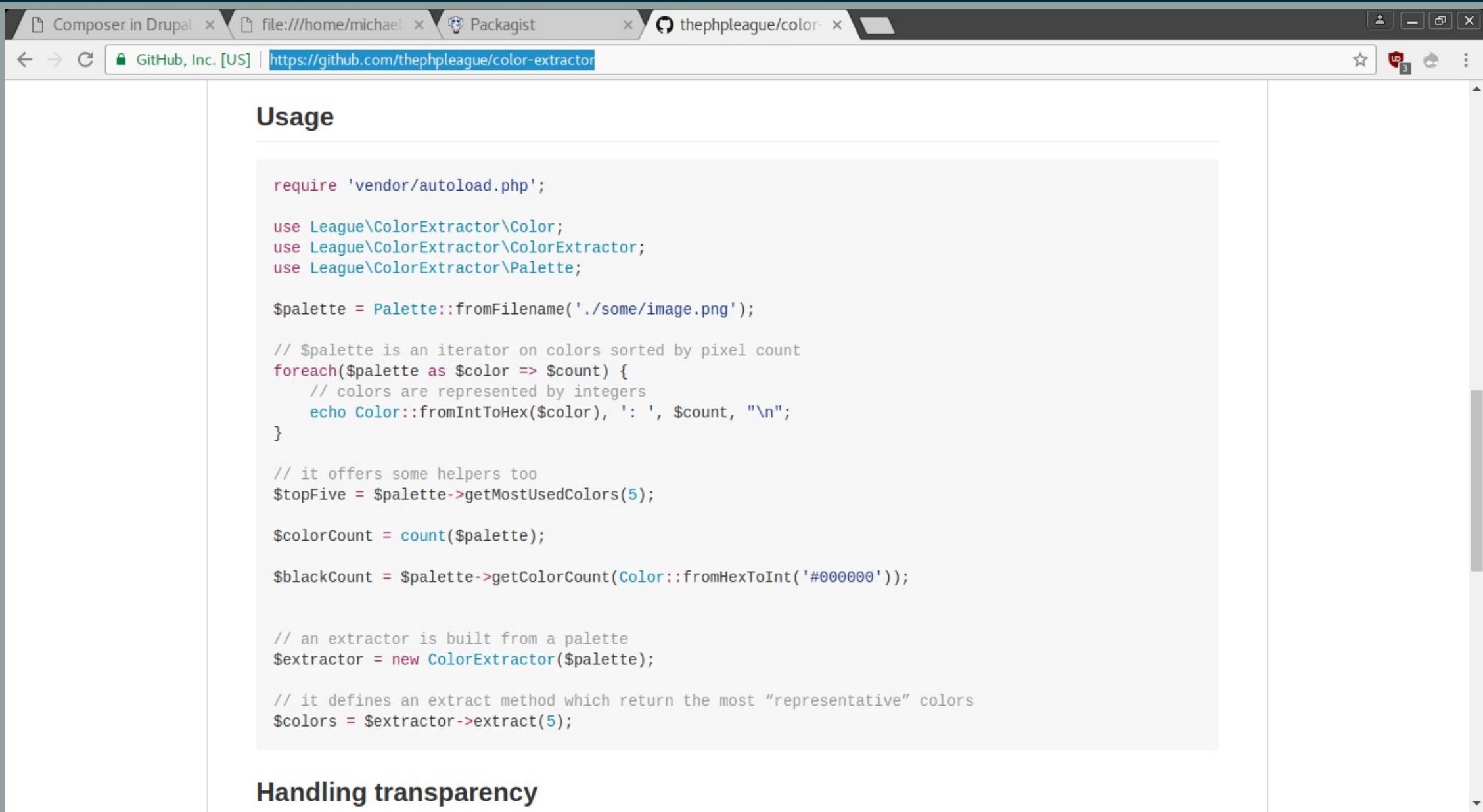
Color Index	Hex Code
0	#7DADDE
1	#3017EC
2	#548AD2
3	#B25900
4	#6C7518
5	#710C00
6	#EF9A3F
7	#CEB034

#DCMuc16

COLOR EXTRACTOR EXAMPLE



COLOR-EXTRACTOR ON GITHUB



The screenshot shows a web browser window with the GitHub URL <https://github.com/thephpleague/color-extractor>. The page content is titled "Usage" and contains the following PHP code example:

```
require 'vendor/autoload.php';

use League\ColorExtractor\Color;
use League\ColorExtractor\ColorExtractor;
use League\ColorExtractor\Palette;

$palette = Palette::fromFilename('./some/image.png');

// $palette is an iterator on colors sorted by pixel count
foreach($palette as $color => $count) {
    // colors are represented by integers
    echo Color::fromIntToHex($color), ': ', $count, "\n";
}

// it offers some helpers too
$topFive = $palette->getMostUsedColors(5);

$colorCount = count($palette);

$blackCount = $palette->getColorCount(Color::fromHexToInt('#000000'));

// an extractor is built from a palette
$extractor = new ColorExtractor($palette);

// it defines an extract method which return the most "representative" colors
$colors = $extractor->extract(5);
```

Below the code, there is a section titled "Handling transparency".

THREE STEPS TO PHP HEAVEN

Get the package

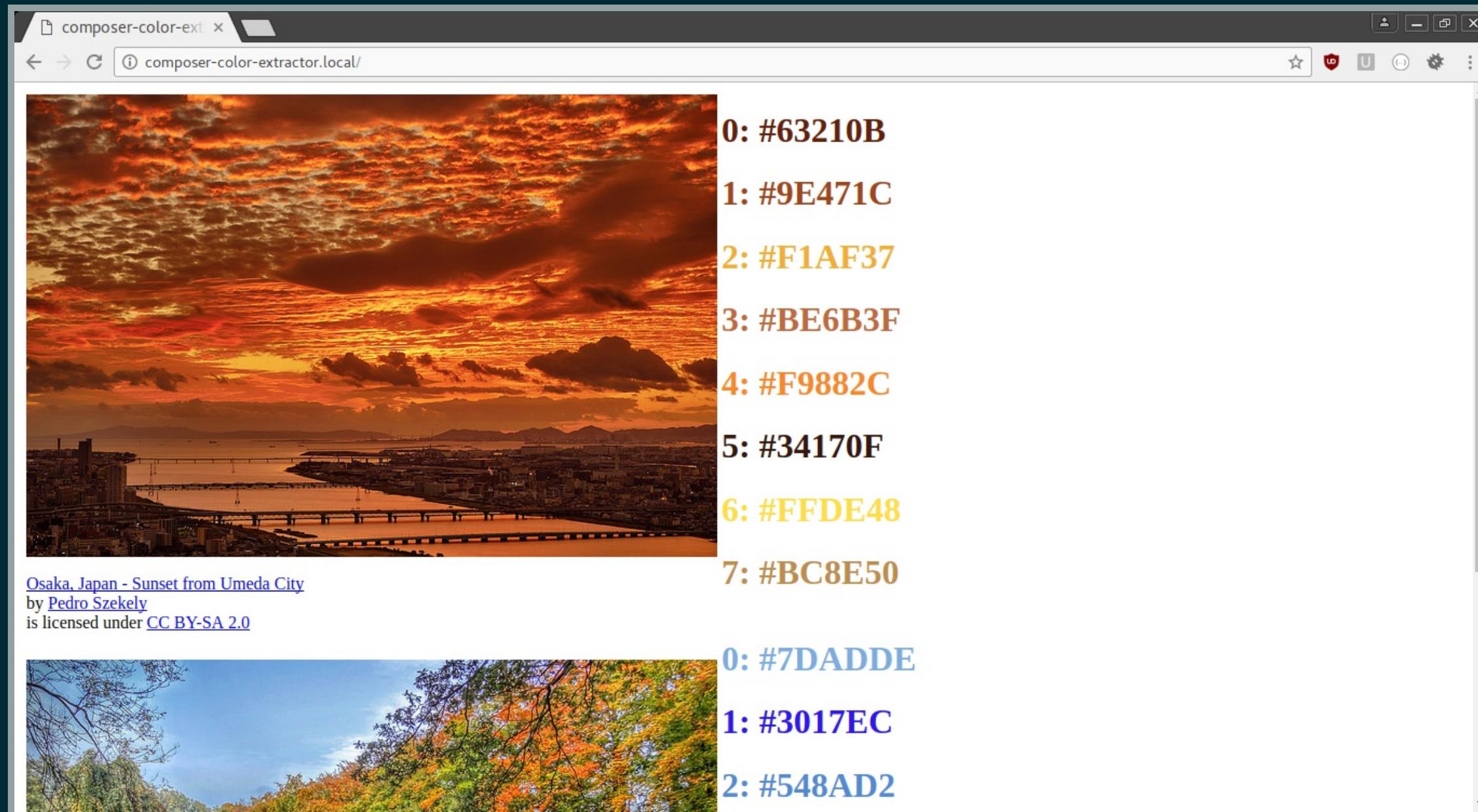
```
composer require vendorname/packagename
```

Get the example code on github put it in index.php

```
php -S localhost:8000
```

Play around with it and have fun

<https://github.com/michaellenahan/composer-color-extractor>



COLOR EXTRACTOR IN PHP

Go into any directory you want and run this command:

```
composer require "league/color-extractor"
```

COLOR EXTRACTOR IN PHP

So, what have you got now, in the root of your project?

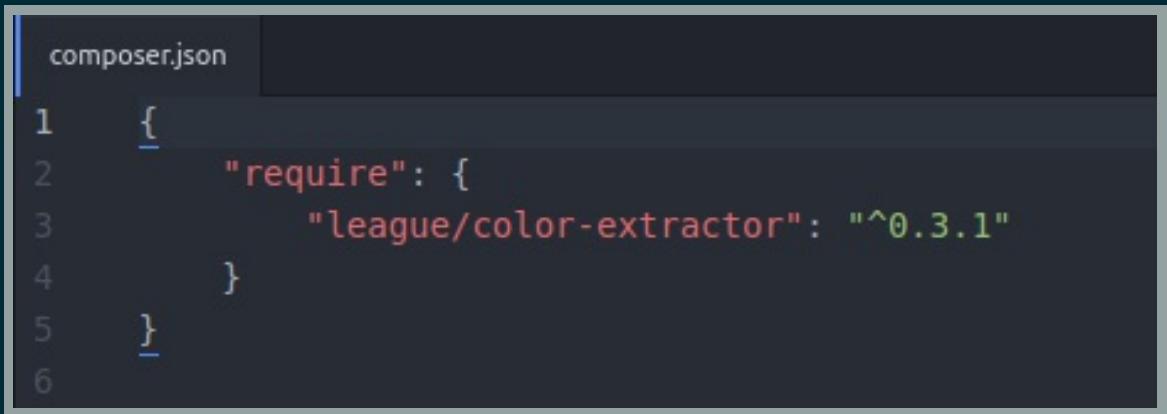
composer.json file

composer.lock file

vendor directory

COLOR EXTRACTOR IN PHP

composer.json file



A screenshot of a code editor showing a file named "composer.json". The file contains the following JSON code:

```
1  {
2      "require": {
3          "league/color-extractor": "^0.3.1"
4      }
5  }
```

smile and take a moment to admire the beautiful simplicity of
what you just witnessed

COLOR EXTRACTOR IN PHP

You did this

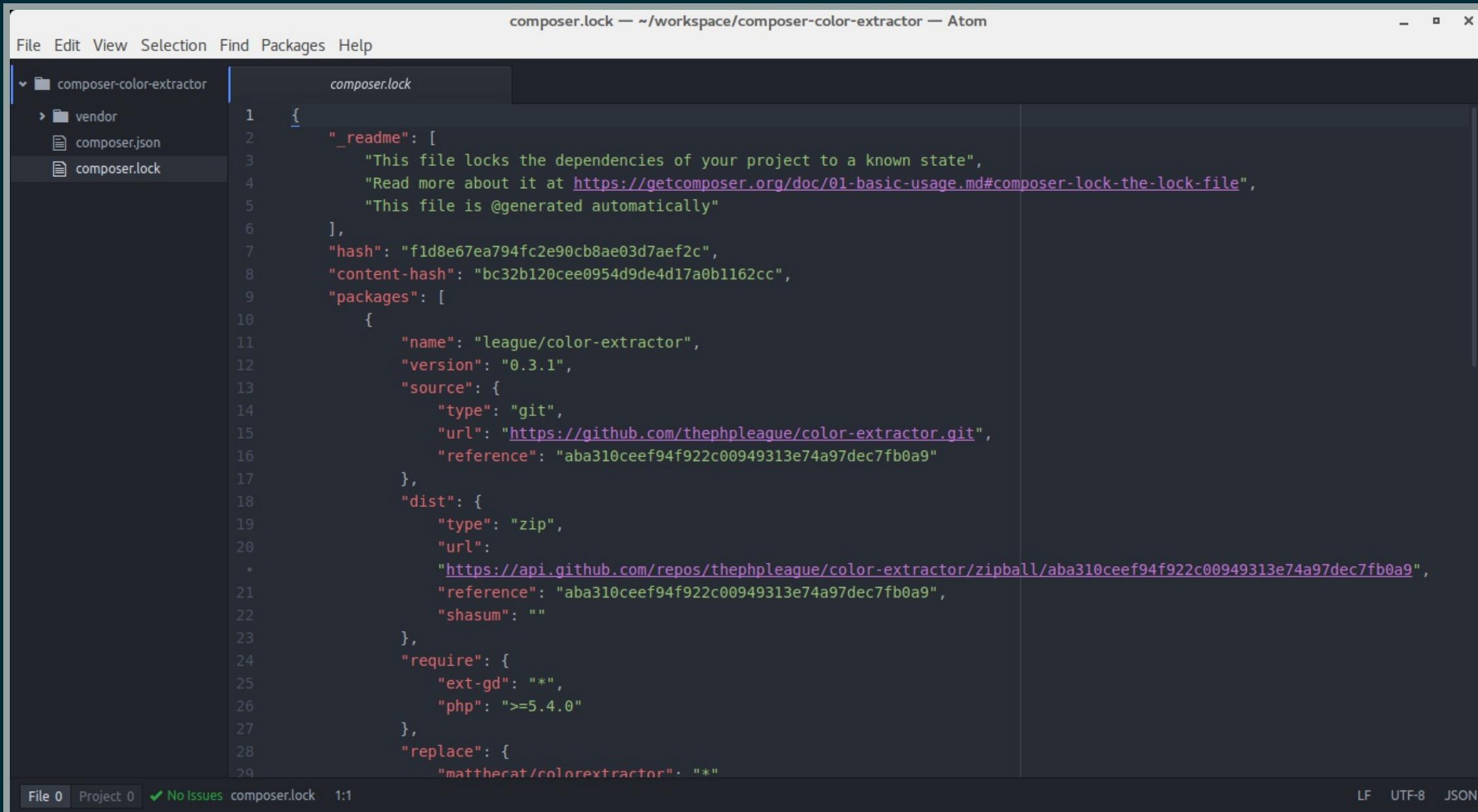
```
composer require "league/color-extractor"
```

You got this



```
composer.json
1  {
2      "require": {
3          "league/color-extractor": "^0.3.1"
4      }
5  }
6
```

COMPOSER LOCK FILE



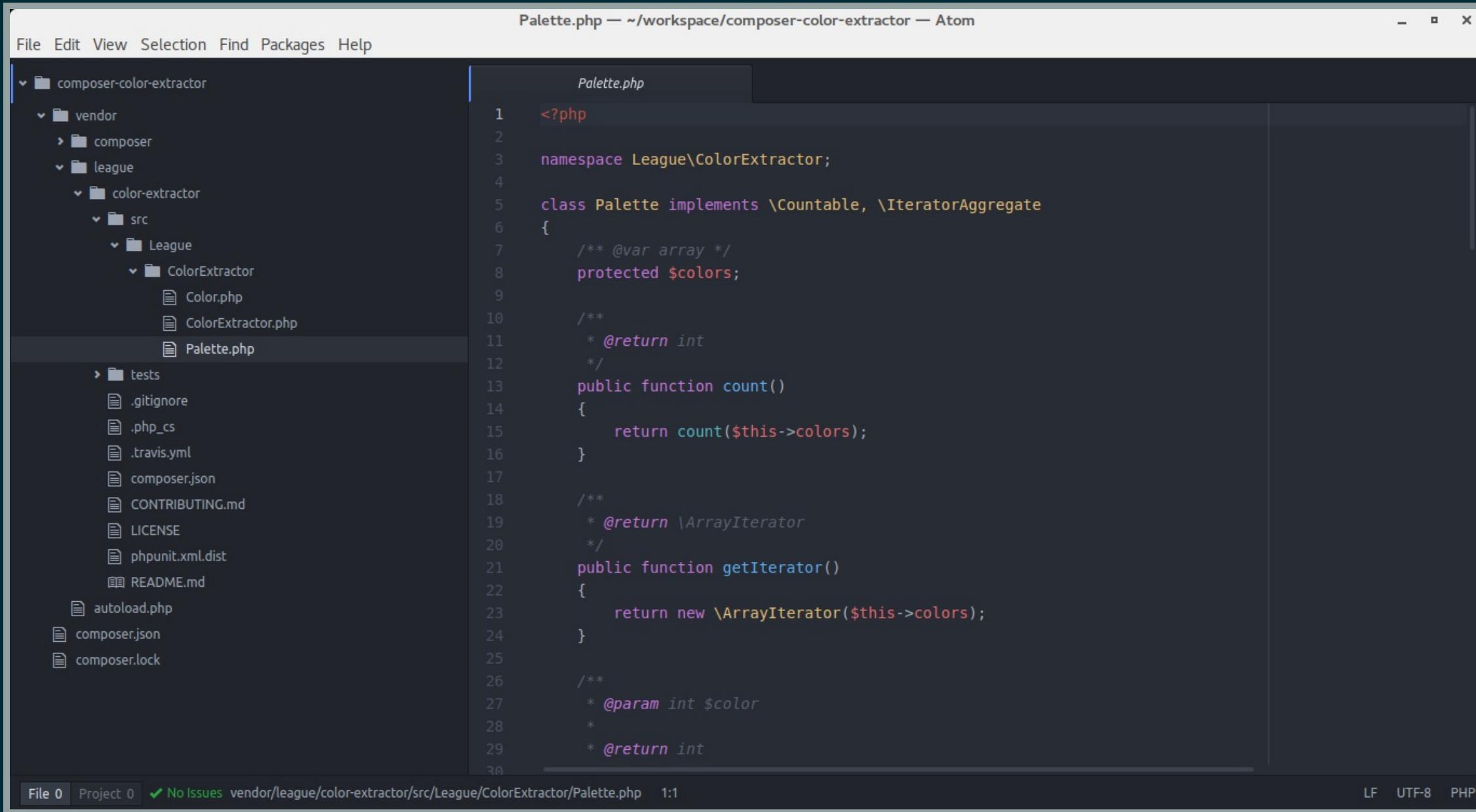
The screenshot shows the Atom code editor interface with a dark theme. The title bar reads "composer.lock — ~/workspace/composer-color-extractor — Atom". The menu bar includes File, Edit, View, Selection, Find, Packages, and Help. The left sidebar shows a project structure with "composer-color-extractor" expanded, containing "vendor", "composer.json", and "composer.lock". The main editor area displays the contents of the "composer.lock" file:

```
1 {  
2     "_readme": [  
3         "This file locks the dependencies of your project to a known state",  
4         "Read more about it at https://getcomposer.org/doc/01-basic-usage.md#composer-lock-the-lock-file",  
5         "This file is @generated automatically"  
6     ],  
7     "hash": "f1d8e67ea794fc2e90cb8ae03d7aef2c",  
8     "content-hash": "bc32b120cee0954d9de4d17a0b1162cc",  
9     "packages": [  
10         {  
11             "name": "league/color-extractor",  
12             "version": "0.3.1",  
13             "source": {  
14                 "type": "git",  
15                 "url": "https://github.com/thepleague/color-extractor.git",  
16                 "reference": "aba310ceef94f922c00949313e74a97dec7fb0a9"  
17             },  
18             "dist": {  
19                 "type": "zip",  
20                 "url": "https://api.github.com/repos/thepleague/color-extractor/zipball/aba310ceef94f922c00949313e74a97dec7fb0a9",  
21                 "reference": "aba310ceef94f922c00949313e74a97dec7fb0a9",  
22                 "shasum": ""  
23             },  
24             "require": {  
25                 "ext-gd": "*",  
26                 "php": ">=5.4.0"  
27             },  
28             "replace": {  
29                 "matthewcat/colorextractor": "*"  
30             }  
31         }  
32     }  
33 }
```

The status bar at the bottom shows "File 0 Project 0 ✓ No Issues composer.lock 1:1" and "LF UTF-8 JSON".

#DCMuc16

YOUR VENDOR DIRECTORY HAS THE CODE



The screenshot shows the Atom code editor interface with the following details:

- Title Bar:** Palette.php — ~/workspace/composer-color-extractor — Atom
- Menu Bar:** File Edit View Selection Find Packages Help
- Left Panel (File Tree):** Shows the project structure:
 - composer-color-extractor
 - vendor
 - composer
 - league
 - color-extractor
 - src
 - League
 - ColorExtractor
 - Color.php
 - ColorExtractor.php
 - Palette.php
 - tests
 - .gitignore
 - .php_cs
 - .travis.yml
 - composer.json
 - CONTRIBUTING.md
 - LICENSE
 - phpunit.xml.dist
 - README.md
 - autoload.php
 - composer.json
 - composer.lock
 - Right Panel (Code Editor):** Displays the content of the file `Palette.php`. The code implements the `\Countable` and `\IteratorAggregate` interfaces. It contains methods for counting colors and returning an iterator.
 - Bottom Status Bar:** File 0 Project 0 ✓ No Issues vendor/league/color-extractor/src/League/ColorExtractor/Palette.php 1:1 LF UTF-8 PHP

<https://github.com/michaellenahan/composer-color-extractor>

The screenshot displays a Sublime Text editor window on the left containing the PHP source code for the `index.php` file of the `composer-color-extractor` project. The code implements a color extraction process using the League's Color Extractor library to analyze images from Creative Commons on Flickr.

On the right, there are two browser windows showing the results of the color extraction:

- Top Window:** A sunset over a city skyline with orange and red hues. The extracted colors are listed on the right:
 - 0: #63210B
 - 1: #9E471C
 - 2: #F1AF37
 - 3: #BE6B3F
 - 4: #F9882C
 - 5: #34170F
 - 6: #FFDE48
 - 7: #BC8E50
- Bottom Window:** A landscape with autumn foliage reflected in water. The extracted colors are listed on the right:
 - 0: #7DADDE
 - 1: #3017EC
 - 2: #548AD2
 - 3: #B25900
 - 4: #6C7518
 - 5: #710C00
 - 6: #EF9A3F
 - 7: #CEB034

Code Snippet (index.php):

```
1 <?php
2
3 require 'vendor/autoload.php';
4
5 use League\ColorExtractor\Color;
6 use League\ColorExtractor\ColorExtractor;
7 use League\ColorExtractor\Palette;
8
9 // Some images from Creative Commons on Flickr.
10 // https://www.flickr.com/search/?text=&license=2%2C3%2C4%2C5%2C6%2C9&sort=interestingness-desc
11 $images = [
12 ];
13
14 $number_of_colors_to_extract = 8;
15
16 foreach ($images as $image) {
17     $palette = Palette::fromFilename($image['file']);
18
19     // an extractor is built from a palette
20     $extractor = new ColorExtractor($palette);
21
22     // it defines an extract method which return the most "representative" colors
23     $colors = $extractor->extract($number_of_colors_to_extract);
24
25     // Creative commons attribution.
26     // See: https://wiki.creativecommons.org/wiki/best_practices_for_attribution#This_is_an_ideal_attribution
27     // $attribution = "$title<br>by $author<br>is licensed under $license";
28
29     echo "<table><tr><td>";
30     echo "<img src='".$image['file']."' />";
31     echo "<p>$attribution</p>";
32     echo "</td><td>";
33
34     foreach ($colors as $key => $color) {
35         $hex_code = Color::fromIntToHex($color);
36         echo "<h1 style='color: $hex_code'$key: $hex_code</h1>";
37     }
38
39     echo "</td></tr></table>";
40
41 }
42
43 }
```

#DCMuc16

SO, WHAT ABOUT
DRUPAL?

The screenshot shows a web browser window with multiple tabs open at the top. The active tab is titled "Using Composer with Drupal" and has the URL <https://www.drupal.org/docs/develop/using-composer/using-composer-with-drupal>. The page content is as follows:

The page header includes the Drupal logo, navigation links for "Download & Extend", "Community", "Documentation", "Support", "Jobs", "Marketplace", and "About", a search bar with placeholder text "Search documentation...", and a user profile icon.

The main title "Documentation" is displayed prominently. Below it, a breadcrumb navigation shows "Develop" and "Using Composer".

Using Composer with Drupal

Last updated on November 13, 2016 - 14:21

Background

Composer is a dependency manager for PHP. Drupal core uses Composer to manage core dependencies like Symfony components and Guzzle.

There are many benefits to using Composer. In short, it allows us to systematically manage a sprawling list of dependencies (and their subsidiary dependencies). It assists with locating, downloading, validating, and loading said packages, all while ensuring that exactly the right versions for each package is used. It can be a pain.

A green "Edit" button with a pencil icon is located on the right side of the main content area.

The right sidebar, titled "Using Composer", contains a list of related topics:

- Using Composer in a Drupal project
- Composer in relation to Drush Make
- Using Composer to manage Drupal site dependencies
- Managing dependencies for a contributed project
- Managing dependencies for a custom project
- Troubleshooting Composer
- Using packages.drupal.org

#DCMuc16

github.com/drupal-composer/drupal-project

The screenshot shows the GitHub repository page for `drupal-composer/drupal-project`. The repository has 127 commits, 3 branches, 0 releases, and 26 contributors. The latest commit was 14 days ago. The repository is licensed under GPL-2.0.

File / Commit	Description	Date
<code>drush</code>	#192: Update README with a new link (#195)	2 months ago
<code>scripts/composer</code>	Fix	3 months ago
<code>.gitignore</code>	Ignore SimpleTest multi-site environment (#208)	28 days ago
<code>.travis.yml</code>	Disable DI unit test	2 months ago
<code>LICENSE</code>	Update LICENSE	a year ago
<code>README.md</code>	Minor README improvement: embed only valid json (#217)	14 days ago
<code>composer.json</code>	Update composer/installers and drupal-composer/drupal-scaffold	28 days ago
<code>phpunit.xml.dist</code>	Init D8 Template	2 years ago
<code>README.md</code>		

#DCMuc16

DRUPAL COMPOSER TEMPLATE

```
composer create-project drupal-composer/drupal-  
project:8.x-dev YOUR-DIRECTORY-NAME --stability dev --  
no-interaction
```

COLOR EXTRACTOR IN DRUPAL 8

Install Drupal 8 using the composer template

```
composer require "league/color-extractor:^0.3"
```

```
diff --git a/composer.json b/composer.json
index 727e031..832370e 100644
--- a/composer.json
+++ b/composer.json
@@ -5,7 +5,8 @@
     "license": "GPL-2.0+",
     "require": {
         "composer/installers": "^1.0.21",
-        "wikimedia/composer-merge-plugin": "~1.3"
+        "wikimedia/composer-merge-plugin": "~1.3",
+        "league/color-extractor": "^0.3"
     },
     "replace": {
         "drupal/core": "~8.2"
     }
 }
```

COLOR EXTRACTOR IN DRUPAL 8

Your drupalroot/vendor directory has the code

```
→ d8-color-extractor git:(master) ✘ cd vendor ←
→ vendor git:(master) ✘ cd league/color-extractor
→ color-extractor git:(master) ✘ ls
composer.json    LICENSE          README.md  tests
CONTRIBUTING.md  phpunit.xml.dist  src
→ color-extractor git:(master) ✘ cd src/League/ColorExtractor
→ ColorExtractor git:(master) ✘ ls
ColorExtractor.php  Color.php  Palette.php
→ ColorExtractor git:(master) ✘
```

COLOR EXTRACTOR IN DRUPAL 8

... and here are the classes you will use in your custom module

```
→ d8-color-extractor git:(master) ✘ cd vendor
→ vendor git:(master) ✘ cd league/color-extractor
→ color-extractor git:(master) ✘ ls
composer.json    LICENSE          README.md  tests
CONTRIBUTING.md  phpunit.xml.dist  src
→ color-extractor git:(master) ✘ cd src/League/ColorExtractor
→ ColorExtractor git:(master) ✘ ls
ColorExtractor.php  Color.php  Palette.php ←
→ ColorExtractor git:(master) ✘
```

CREATE THE CUSTOM MODULE

... using drupal console

```
drupal generate:module
```

```
drupal generate:module  
drupal generate:module 74x38  
→ d8-template-color-extractor git:(master) drupal generate:module  
  
// Welcome to the Drupal module generator  
  
Enter the new module name:  
> color_extractor  
  
Enter the module machine name [color_extractor]:  
>  
  
Enter the module Path [/modules/custom]:  
>  
  
Enter module description [My Awesome Module]:  
> See: https://github.com/michaellenahan/using-composer-drupal-8-modules  
  
Enter package name [Custom]:  
>  
  
Enter Drupal Core version [8.x]:  
>  
  
Do you want to generate a .module file (yes/no) [yes]:  
>  
  
Define module as feature (yes/no) [no]:  
>  
  
Do you want to add a composer.json file to your module (yes/no) [yes]:  
>
```

#DCMuc16

We'll use the D8 color_field module to store the colors

[d8-template-color-extractor.local/node/1](#)

Autumn Colours, Landgoed Voorstonden, Brummen, Netherlands - 1032

Submitted by admin on Sun, 12/04/2016 - 09:50



© www.tomjutte.nl

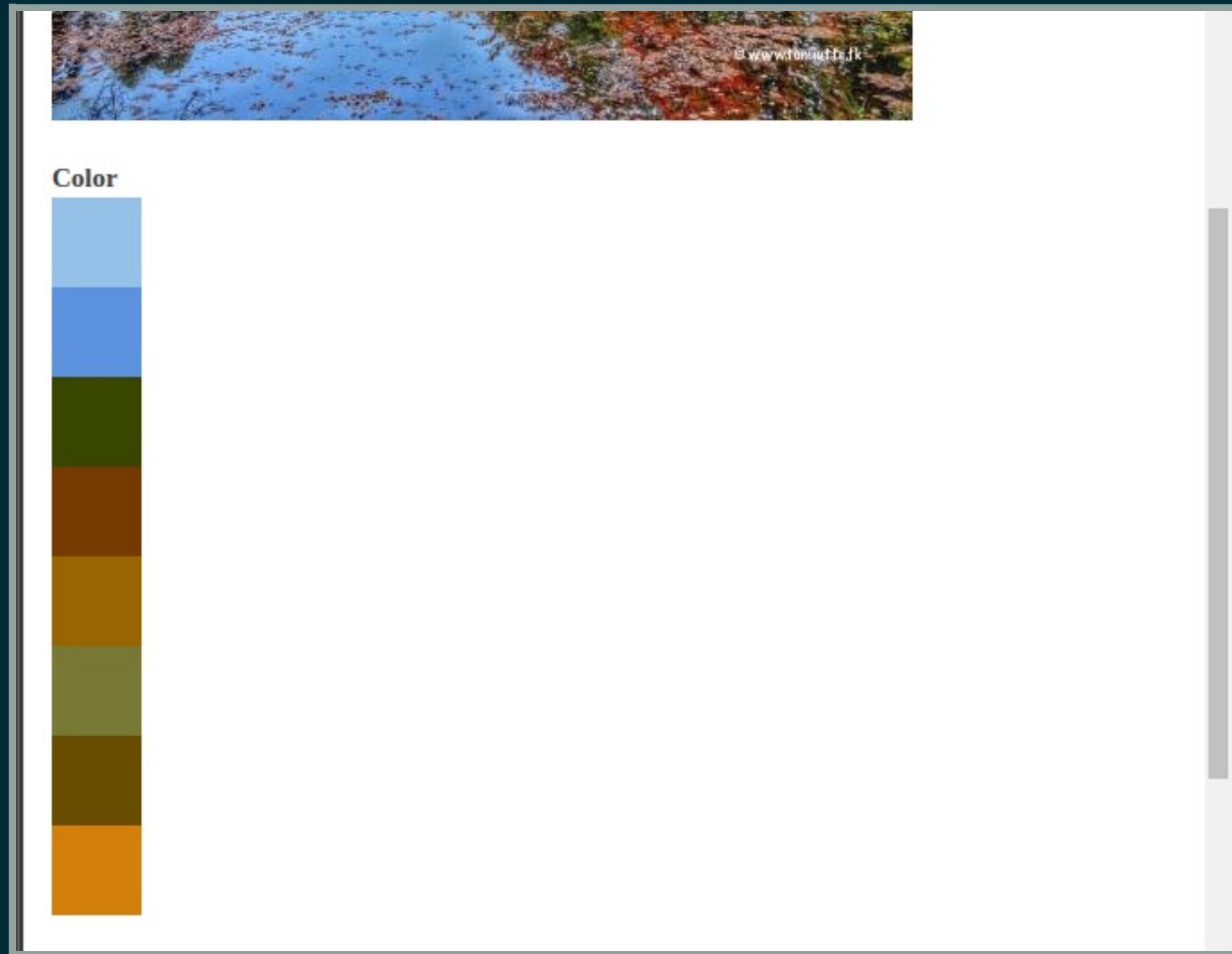
Autumn Colours,
Landgoed
Voorstonden,
Brummen,
Netherlands - 1032"
by Tom Jutte is
licensed under CC
BY ND 2.0

Color

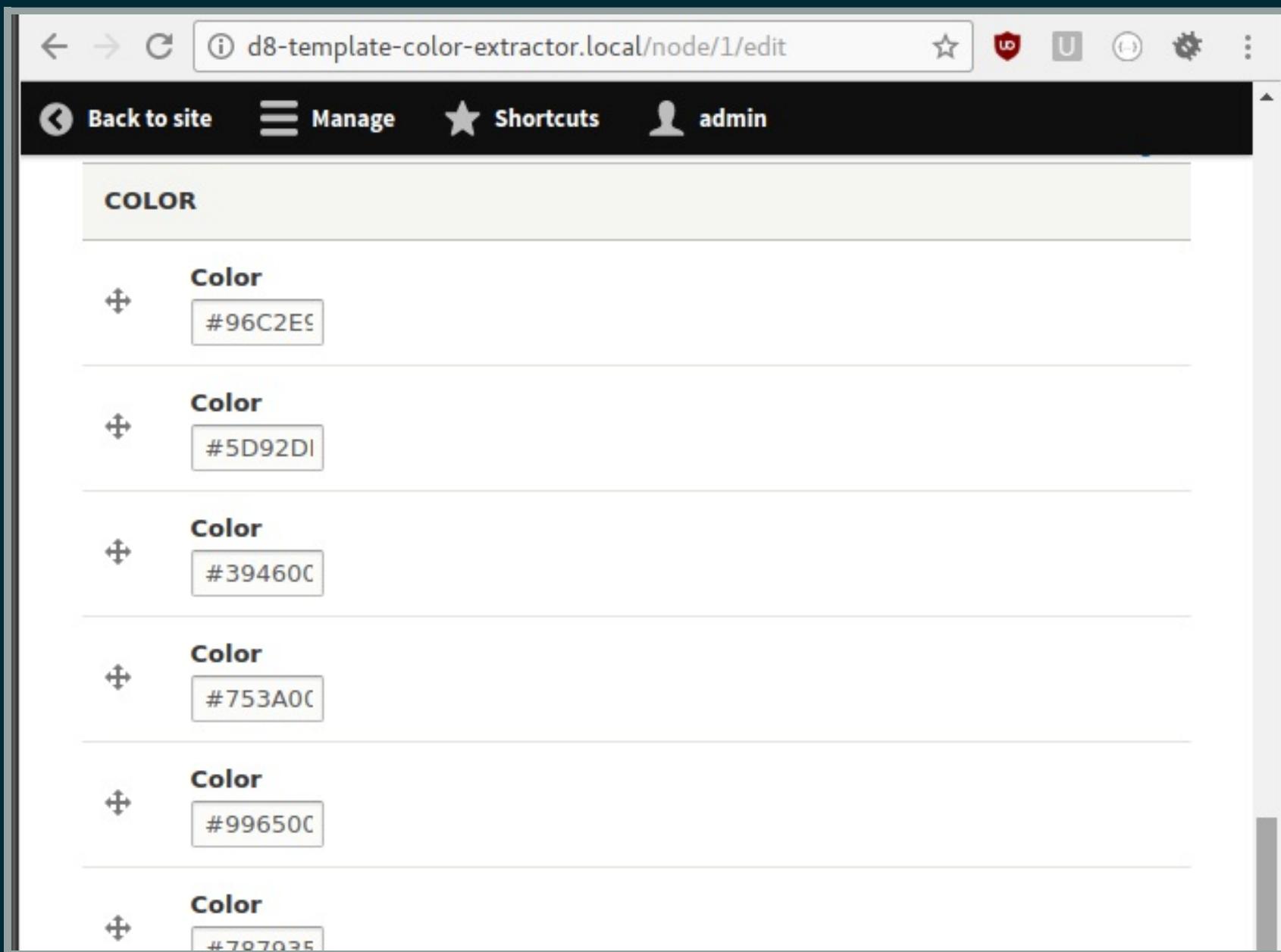


#DCMuc16

We'll use the D8 color_field module to store the colors



We'll use the D8 color_field module to store the colors



DOWNLOAD THE COLOR_FIELD MODULE

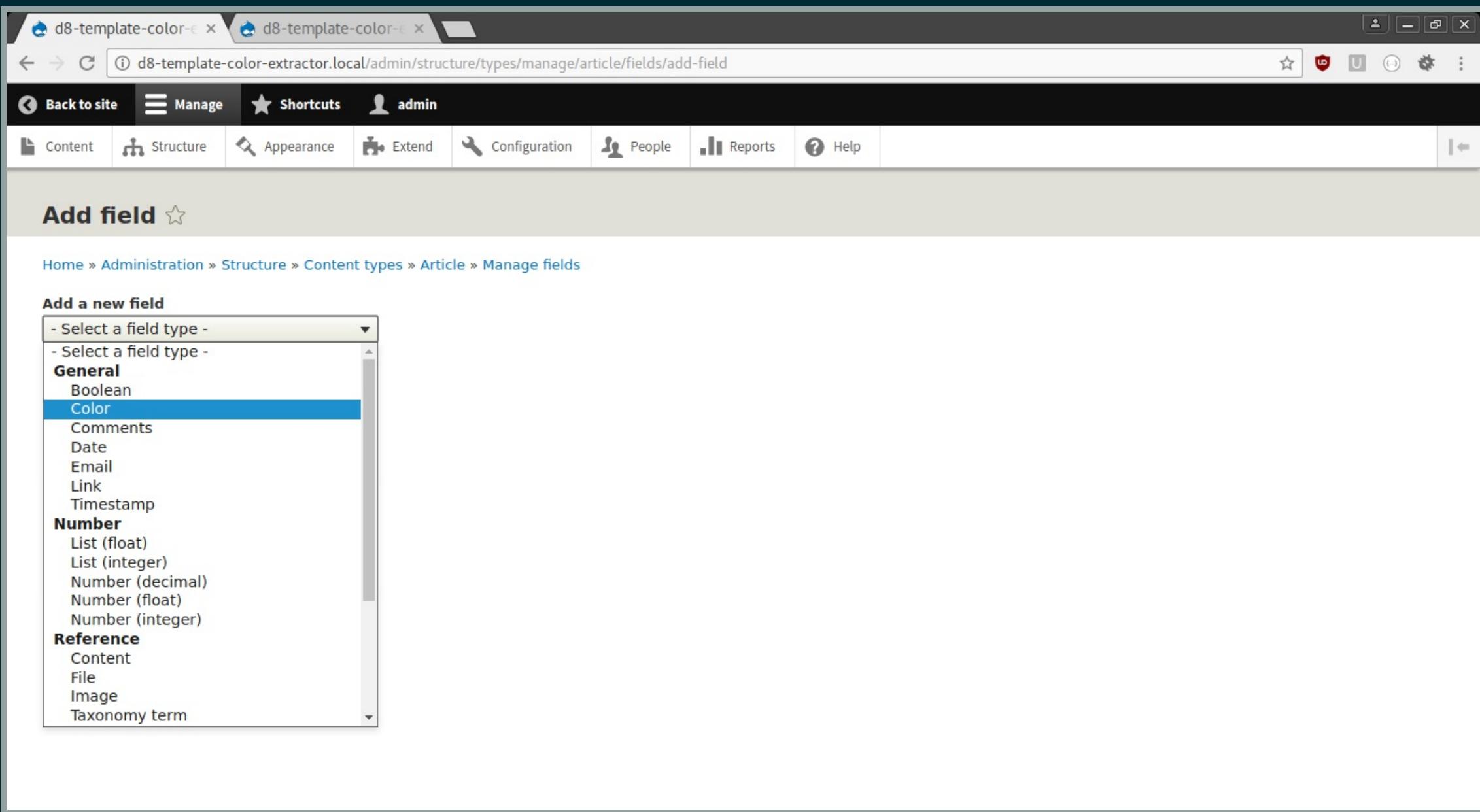
... using composer require ...

```
composer require drupal/color_field "2.x-dev"
```

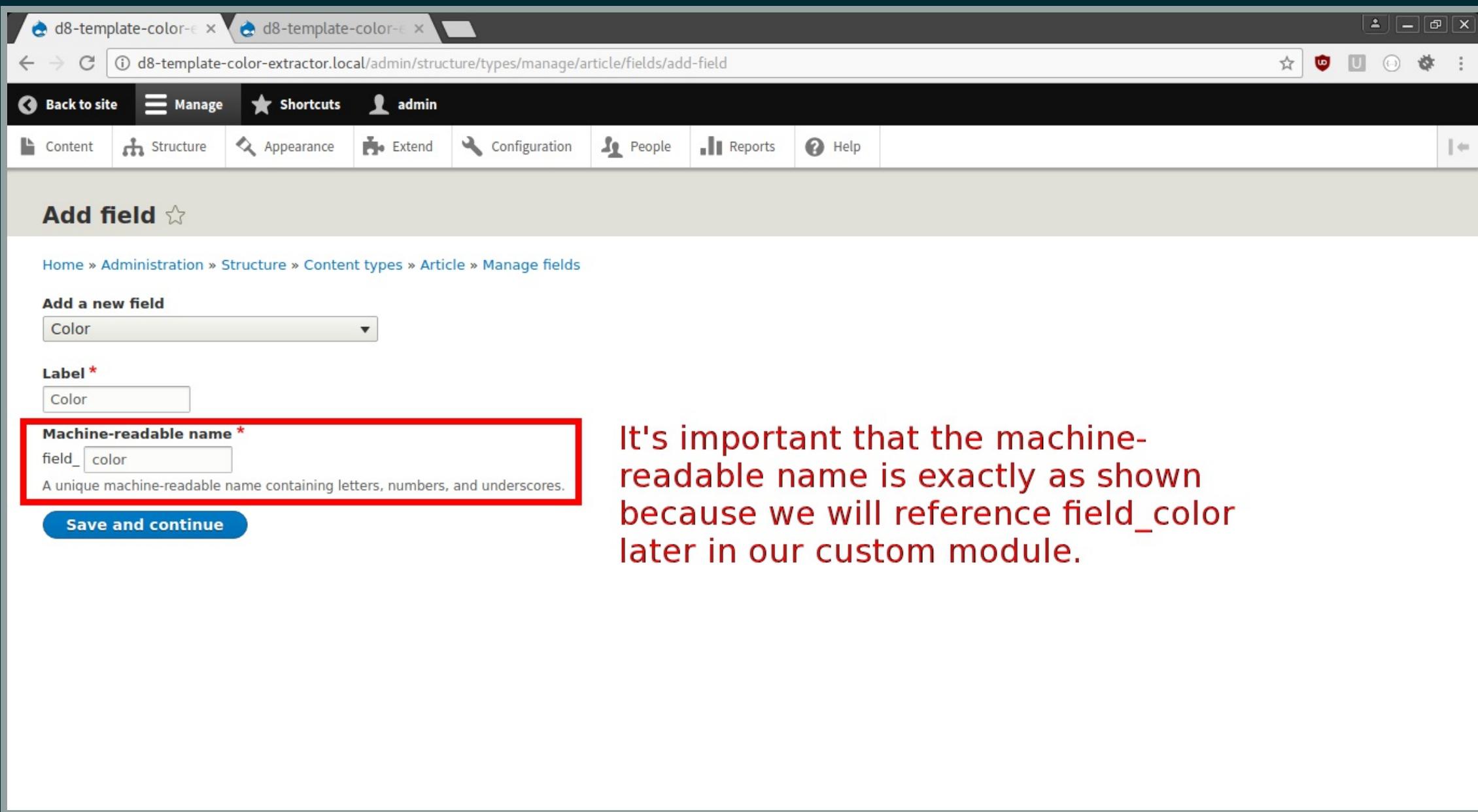
The screenshot shows the 'Manage fields' page for the 'Article' content type in a Drupal 8 environment. The URL is `d8-color-extractor.local/admin/structure/types/manage/article/fields`. The page has a dark header with 'Back to site', 'Manage', 'Shortcuts', and 'admin'. Below the header is a navigation bar with links for 'Content', 'Structure', 'Appearance', 'Extend', 'Configuration', 'People', 'Reports', and 'Help'. The main title is 'Manage fields ☆'. Below the title are four tabs: 'Edit' (selected), 'Manage fields', 'Manage form display', and 'Manage display'. The breadcrumb trail is 'Home > Administration > Structure > Content types > Article'. A blue button labeled '+ Add field' is visible. The main content area is a table with columns: 'LABEL', 'MACHINE NAME', 'FIELD TYPE', and 'OPERATIONS'. The table rows are:

LABEL	MACHINE NAME	FIELD TYPE	OPERATIONS
Body	body	Text (formatted, long, with summary)	Edit ▾
Comments	comment	Comments	Edit ▾
Image	field_image	Image	Edit ▾
Tags	field_tags	Entity reference	Edit ▾

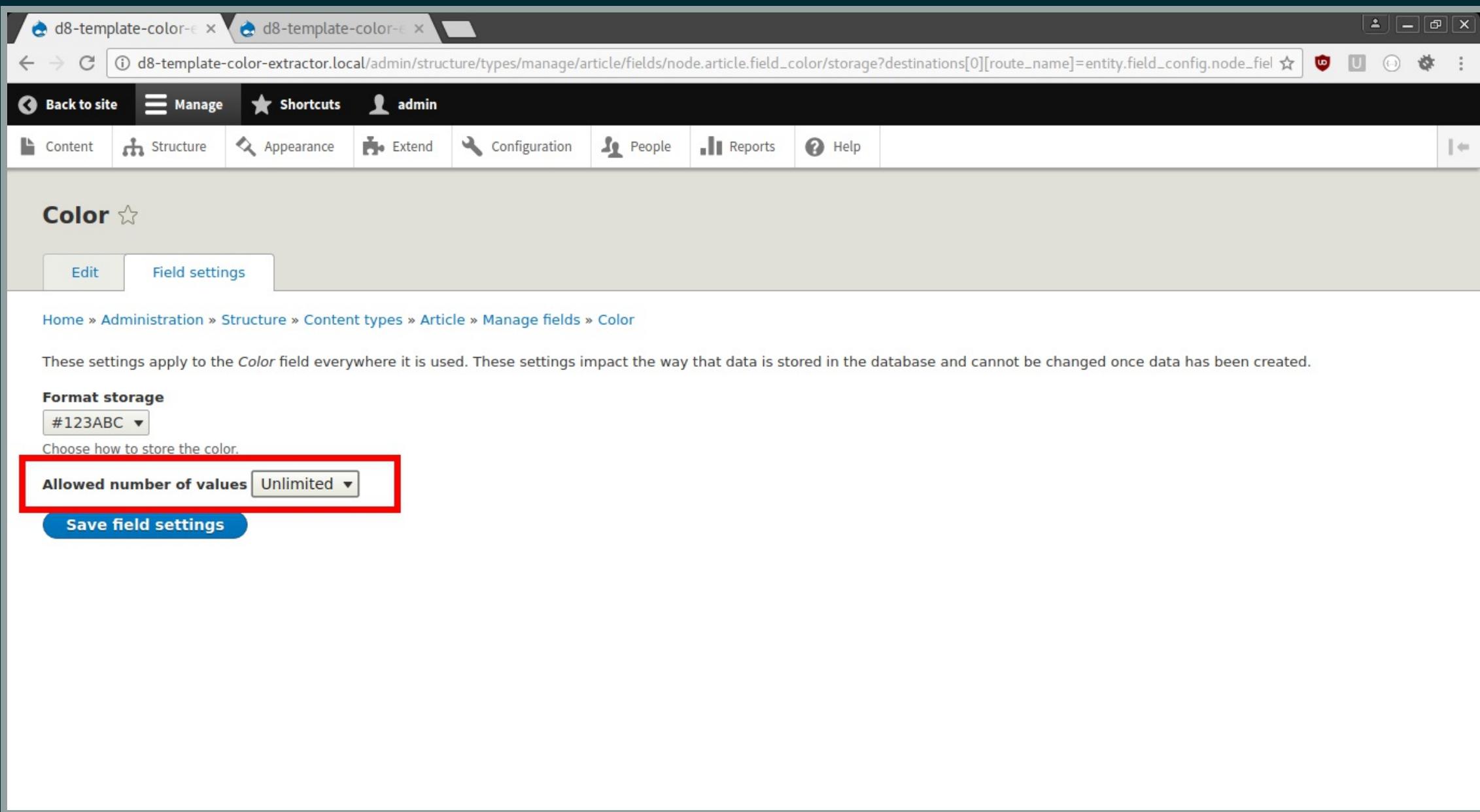
#DCMuc16



#DCMuc16



It's important that the machine-readable name is exactly as shown because we will reference `field_color` later in our custom module.



#DCMuc16

The screenshot shows the Drupal 8 Admin UI for managing form display settings. The page title is "d8-template-color-extractor.local/admin/structure/types/manage/article/form-display". The top navigation bar includes "Home", "Manage", "Shortcuts", and "admin". Below the navigation is a horizontal menu with links for "Content", "Structure", "Appearance", "Extend", "Configuration", "People", "Reports", and "Help".

The main content area displays several field settings:

- "Promoted to front page": Type is "Single on/off checkbox", "Use field label: Yes".
- "Sticky at top of lists": Type is "Single on/off checkbox", "Use field label: Yes".
- "Comments": Type is "Comment".
- "URL alias": Type is "URL alias".
- "Color": Type is "Color grid". This field is highlighted with a red box.

Below the fields, there is a section titled "Disabled" with the note "No field is hidden." At the bottom left is a blue "Save" button. To the right of the "Color" field, there is a technical summary:
Cell width: 10
Cell height: 10
Cell margin: 1
Box width: 115
Box height: 20
Columns: 16

#DCMuc16

Create a node with an image

The screenshot shows the Drupal 8 node creation interface. The top navigation bar includes tabs for Back to site, Manage, Shortcuts, and admin. Below the navigation is a toolbar with links for Content, Structure, Appearance, Extend, Configuration, People, Reports, and Help.

Title *: Autumn Colours, Landgoed Voorstonden, Brummen, Netherlands - 1032

Body (Edit summary): The body content is a block of HTML code describing a photo from Flickr. It includes a link to the photo and information about its license under CC BY ND 2.0.

Text format: Basic HTML

Tags: A text input field for comma-separated tags.

Image: A file upload input field showing "No file chosen". Below it, instructions state "One file only.", "2 MB limit.", and "Allowed types: png gif jpg jpeg."

Last saved: Not saved yet

Author: admin

Create new revision

Revision log message: A text area for describing changes, with the placeholder "Briefly describe the changes you have made."

MENU SETTINGS

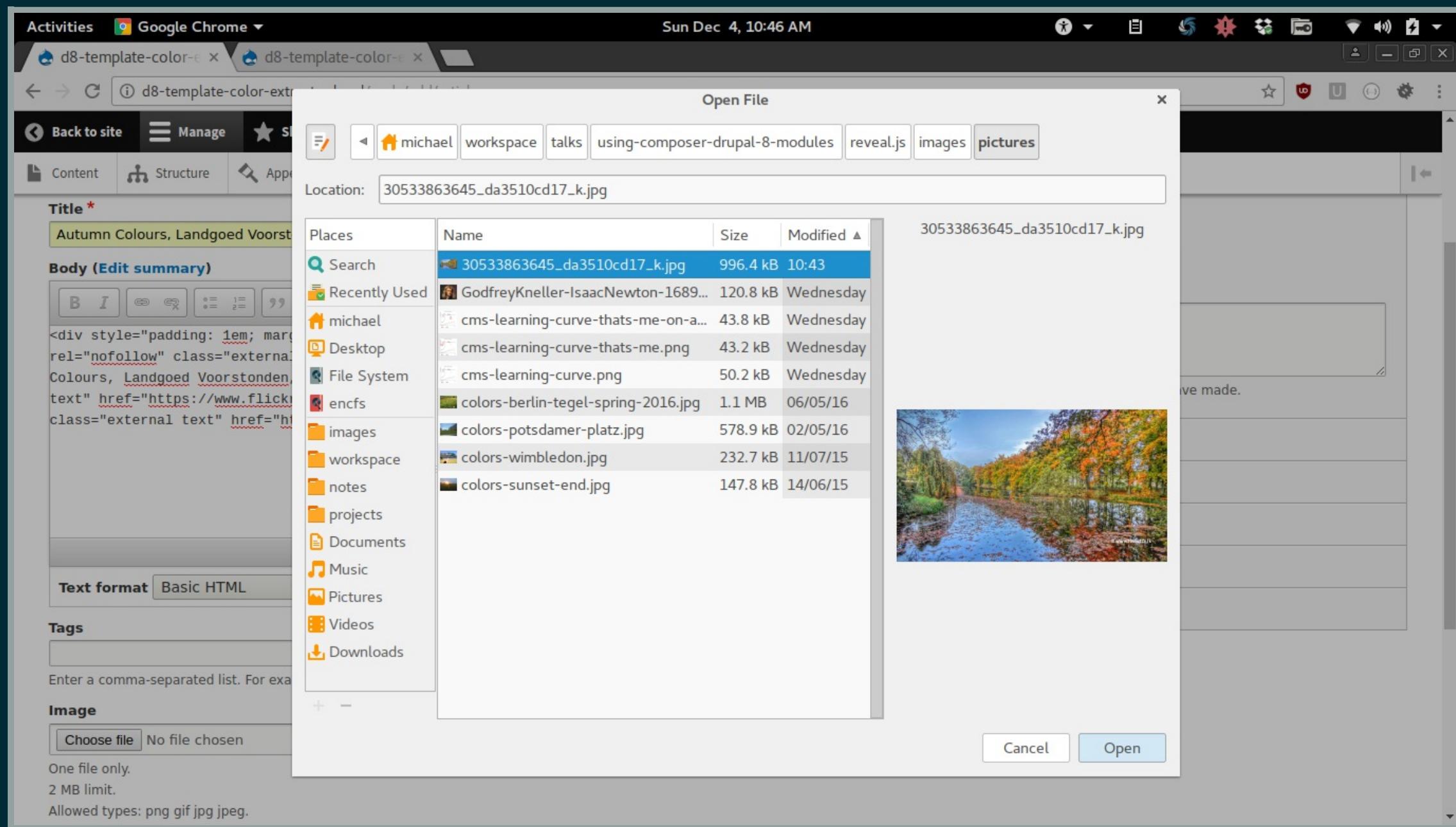
COMMENT SETTINGS *

URL PATH SETTINGS

AUTHORING INFORMATION

PROMOTION OPTIONS

Create a node with an image



#DCMuc16

When we save the node we expect the color fields to be populated ... because of the form submit callback in the code

Image



Alternative text *

Autumn Colours, Landgoed Voorstonden, Brummen, Netherlands - 1032 by ^
This text will be used by screen readers, search engines, or when the image cannot be loaded.

 30533863645_da3510cd17_k.jpg (973.09 KB) Remove

[Show row weights](#)

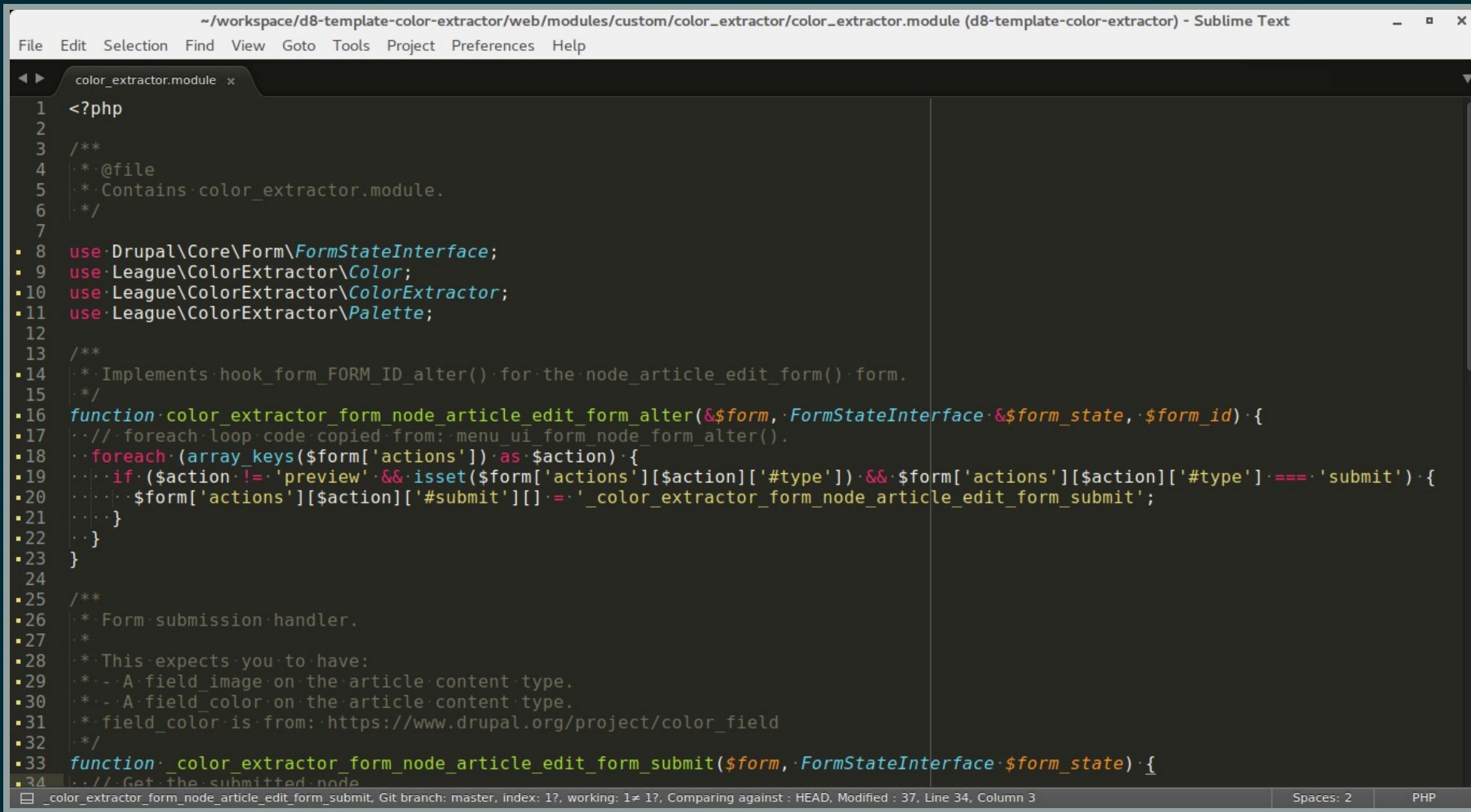
COLOR

 **Color**

Add another item

Save and publish Preview

Here's the code

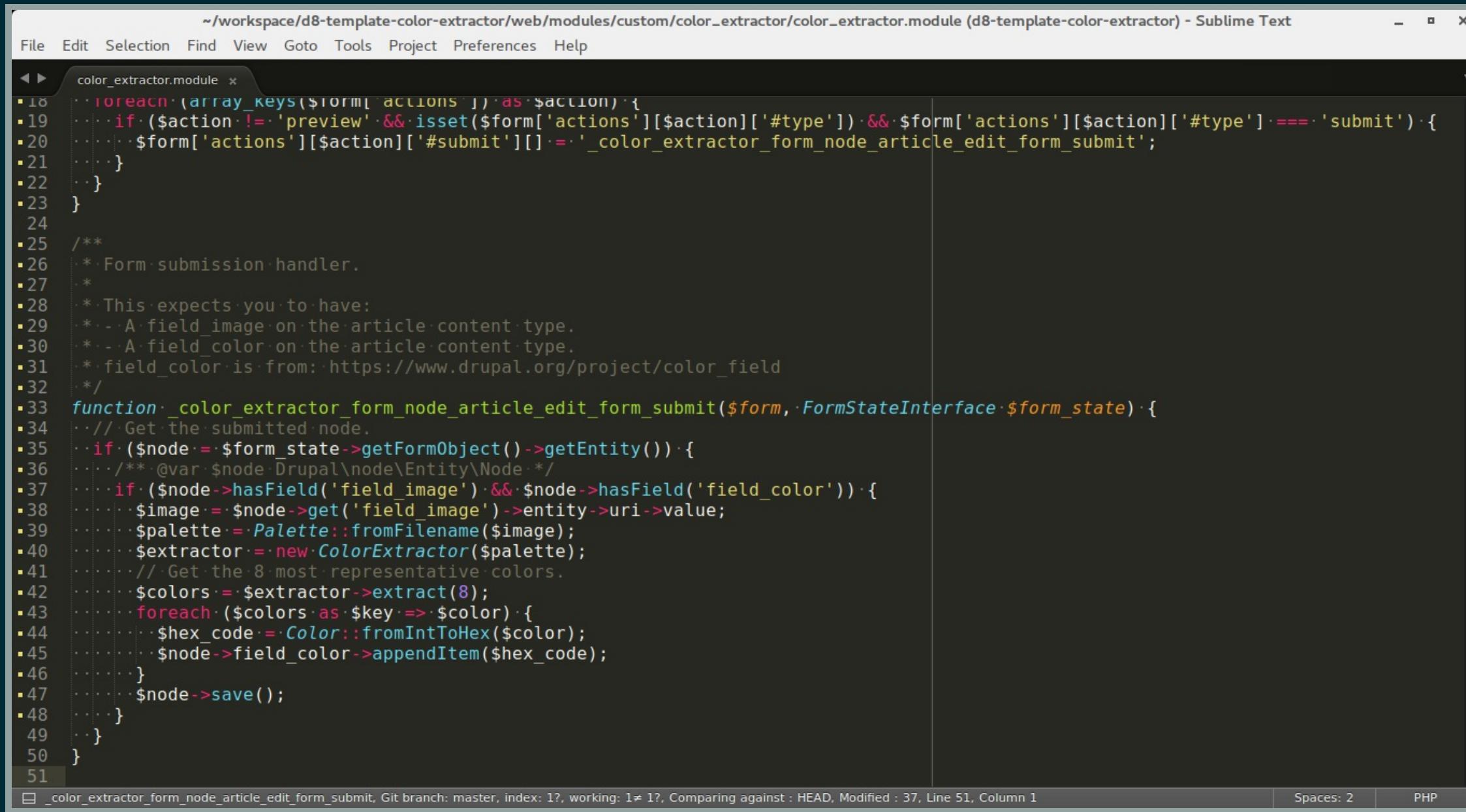


A screenshot of a Sublime Text editor window displaying PHP code for a Drupal module named 'color_extractor.module'. The code implements two functions: 'color_extractor_form_node_article_edit_form_alter' and '_color_extractor_form_node_article_edit_form_submit'. The code uses namespaces like 'Drupal\Core\Form\FormStateInterface' and 'League\ColorExtractor\Color'. It includes comments explaining the purpose of each function and the submission handler.

```
~/workspace/d8-template-color-extractor/web/modules/custom/color_extractor/color_extractor.module (d8-template-color-extractor) - Sublime Text
File Edit Selection Find View Goto Tools Project Preferences Help
color_extractor.module *
1 <?php
2
3 /**
4 * @file
5 * Contains color_extractor.module.
6 */
7
8 use Drupal\Core\Form\FormStateInterface;
9 use League\ColorExtractor\Color;
10 use League\ColorExtractor\ColorExtractor;
11 use League\ColorExtractor\Palette;
12
13 /**
14 * Implements hook_form_FORM_ID_alter() for the node_article_edit_form() form.
15 */
16 function color_extractor_form_node_article_edit_form_alter(&$form, $FormStateInterface, $form_id) {
17   // ... foreach loop code copied from menu_ui_form_node_form_alter().
18   foreach (array_keys($form['actions']) as $action) {
19     if ($action != 'preview' && !isset($form['actions'][$action]['#type']) && $form['actions'][$action]['#type'] == 'submit') {
20       $form['actions'][$action]['#submit'][] = '_color_extractor_form_node_article_edit_form_submit';
21     }
22   }
23 }
24
25 /**
26 * Form submission handler.
27 *
28 * This expects you to have:
29 * - A field_image on the article content type.
30 * - A field_color on the article content type.
31 * field_color is from: https://www.drupal.org/project/color_field
32 */
33 function _color_extractor_form_node_article_edit_form_submit($form, $FormStateInterface, $form_state) {
34   // ... Get the submitted node
```

_color_extractor_form_node_article_edit_form_submit, Git branch: master, index: 1?, working: 1≠ 1?, Comparing against : HEAD, Modified : 37, Line 34, Column 3 | Spaces: 2 | PHP

Here's the code



The screenshot shows a Sublime Text editor window displaying a PHP file named `color_extractor.module`. The file contains code for a Drupal module, specifically a form submission handler for article nodes. The code uses the `ColorExtractor` class to extract colors from the node's image field and save them to the node's color field. The code is well-commented, explaining the logic for handling form submissions and extracting colors.

```
~/workspace/d8-template-color-extractor/web/modules/custom/color_extractor/color_extractor.module (d8-template-color-extractor) - Sublime Text
File Edit Selection Find View Goto Tools Project Preferences Help
color_extractor.module *
18     foreach(array_keys($form['actions']) as $action) {
19         if($action != 'preview' && !isset($form['actions'][$action]['#type']) && $form['actions'][$action]['#type'] === 'submit') {
20             $form['actions'][$action]['#submit'][] = '_color_extractor_form_node_article_edit_form_submit';
21         }
22     }
23 }
24 /**
25 * Form submission handler.
26 */
27 /**
28 * This expects you to have:
29 * - A field_image on the article content type.
30 * - A field_color on the article content type.
31 * field_color is from: https://www.drupal.org/project/color_field
32 */
33 function _color_extractor_form_node_article_edit_form_submit($form, FormStateInterface $form_state) {
34     // Get the submitted node.
35     if($node = $form_state->getFormObject()->getEntity()) {
36         /** @var $node \Drupal\node\Entity\Node */
37         if($node->hasField('field_image') && $node->hasField('field_color')) {
38             $image = $node->get('field_image')->entity->uri->value;
39             $palette = Palette::fromFilename($image);
40             $extractor = new ColorExtractor($palette);
41             // Get the 8 most representative colors.
42             $colors = $extractor->extract(8);
43             foreach($colors as $key => $color) {
44                 $hex_code = Color::fromIntToHex($color);
45                 $node->field_color->appendItem($hex_code);
46             }
47             $node->save();
48         }
49     }
50 }
51
```

_color_extractor_form_node_article_edit_form_submit, Git branch: master, index: 1?, working: 1≠ 1?, Comparing against : HEAD, Modified : 37, Line 51, Column 1 | Spaces: 2 | PHP

NEXT STEPS

Make this code generic and contribute it on drupal.org

Want to help?

Get in touch!

michaellenahan on [drupal.org](#) and [github](#)

[@mcaleaa](#) on twitter

**OUR FAMILY IS GROWING
AND IT'S GETTING BETTER
GO TO DIFFERENT CONFERENCES!
GO TO YOUR LOCAL PHP MEETUPS!**

BUT STAY WITH US, TOO:

DAVOS, 16-19 FEBRUARY 2017

ICELAND, 24-26 FEBRUARY 2017

FRANKFURT, 22-23 APRIL 2017

HEIDELBERG, 14-17 JULY 2017

DRUPALCON VIENNA, 25-29 SEPTEMBER 2017

SCHWERIN, 21-22 OCTOBER 2017