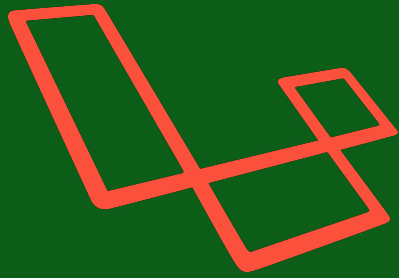


Laravel 5 & Yii 2

A look into two popular PHP frameworks



sourcetoad
DEVELOPMENT STUDIO





Who am I?

- Connor Tumbleson
- <https://github.com/iBotPeaches>
- Developer (SourceToad)

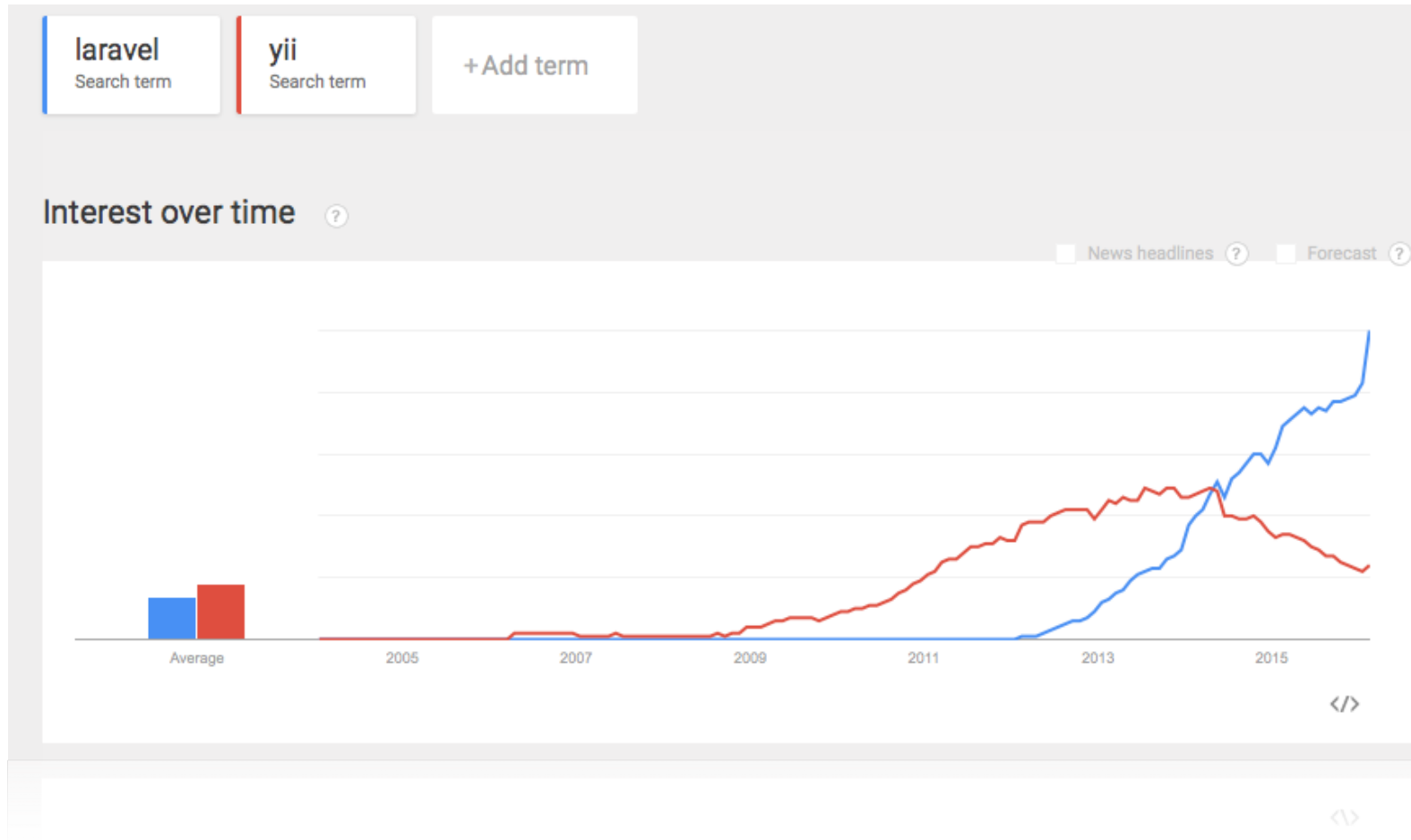
- Built Applications in
 - CodeIgniter
 - Laravel 4.2
 - Laravel 5
 - Laravel 5.1
 - Yii 2

Plan

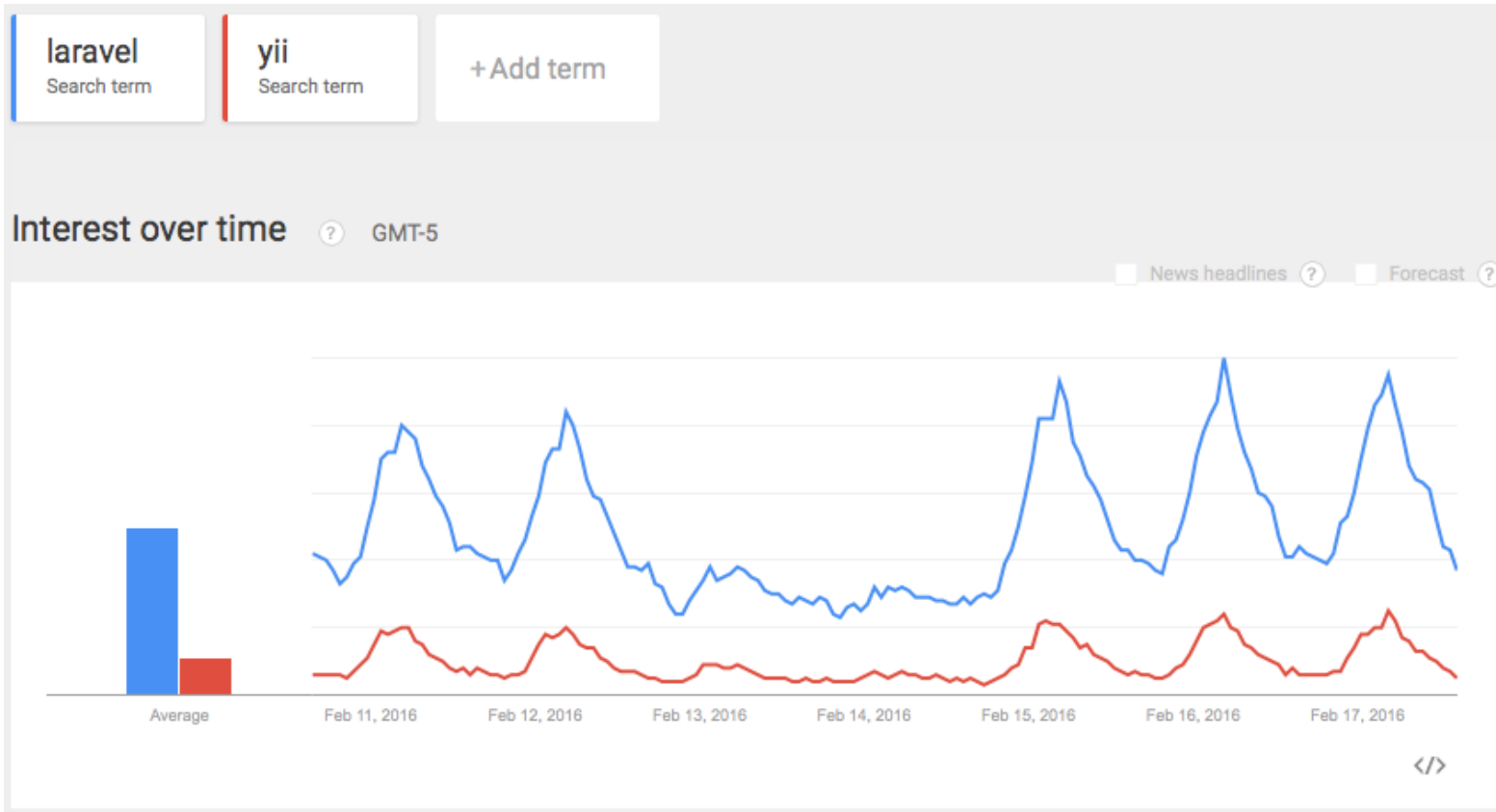
- Framework Popularity
- History
- Setup & Configuration
- Environments
- Controllers & Routes
- Active Record
- Views & Widgets
- Modules & Packages
- Code Generation
- Community
- Versioning



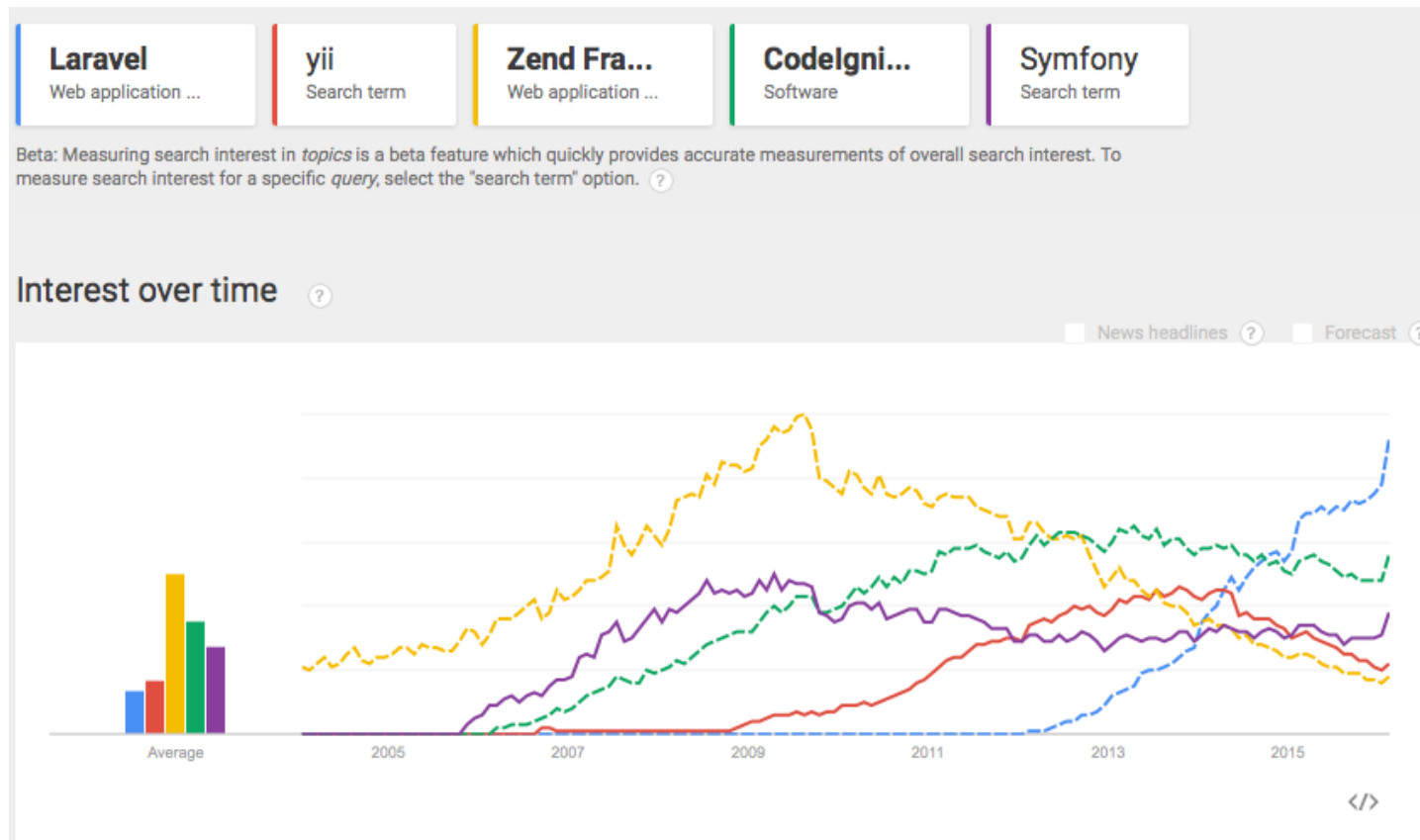
Framework Popularity - History



Framework Popularity - 7 day



Framework Popularity - Multiple Frameworks



Laravel - The PHP Framework for Web Artisans

- Started in June 2011
- 1.0 Released August 2011
- Laravel 4 included Composer in May 2013

- Started by: Taylor Otwell
- Previously: CodeIgniter (used, not team)
- Last Release: v5.2.20 (2/19/2016)
 - as of 2/22/2016

Yii - A high-performance PHP framework best for developing Web 2.0 applications

- Started in October 2006
- 1.0 Release December 2008
- Yii2 included Composer in October 2014

- Started by: Qiang Xue
- Previously: Prado (dev)
- Last Release: v2.0.7 (2/14/2016)

Yii is pronounced as Yee, and is an acronym for "Yes It Is!" This is often the accurate, and most concise response to inquires from those new to Yii:

Is it fast? ...

Is it secure? ...

Is it professional? ...

Is it right for my next project? ... Yes, it is!

Setup

Both install via 1-2 lines with composer

composer is PHP's answer to package managers.

- matching Ruby's bundler & Node's npm

```
composer create-project --prefer-dist laravel/laravel blog
```

```
composer global require "fxp/composer-asset-plugin:~1.1.1"
```

```
composer create-project --prefer-dist yiisoft/yii2-app-basic basic
```

Configuration

- **Laravel**

- Highly commented and separated files
- Priority to dotenv (environment separation)
- Lots of files (12 on stock)

- **Yii**

- Chained loading of configuration files
- Multi depth arrays (5+)
- Mixed logic
- Small amount of configuration files

app.php

```

1 <?php
2
3 return [
4
5     /*
6     |-----|
7     | Application Environment
8     |-----|
9
10    | This value determines the "environment" your application is currently
11    | running in. This may determine how you prefer to configure various
12    | services your application utilizes. Set this in your ".env" file.
13    |
14    */
15
16    'env' => env('APP_ENV', 'production'),
17
18    /*
19    |-----|
20    | Application Debug Mode
21    |-----|
22
23    | When your application is in debug mode, detailed error messages with
24    | stack traces will be shown on every error that occurs within your
25    | application. If disabled, a simple generic error page is shown.
26    |
27    */
28
29    'debug' => env('APP_DEBUG', false),
30
31    /*
32    |-----|
33    | Application URL
34    |-----|
35
36    | This URL is used by the console to properly generate URLs when using
37    | the Artisan command line tool. You should set this to the root of
38    | your application so that it is used when running Artisan tasks.
39    |
40    */
41
42    'url' => 'http://localhost',
43
44    /*
45    |-----|
46    | Application Timezone

```

web.php

```

1 <?php
2
3 $params = require(__DIR__ . '/params.php');
4
5 $config = [
6     'id' => 'basic',
7     'basePath' => dirname(__DIR__),
8     'bootstrap' => ['log'],
9     'components' => [
10         'request' => [
11             // !!! Insert a secret key in the following (if it is empty) - this is required by cook
12             'cookieValidationKey' => '1CKe4whYJ3CAMWzVA0rp-e9q-VQWdSqK',
13         ],
14         'cache' => [
15             'class' => 'yii\caching\FileCache',
16         ],
17         'user' => [
18             'identityClass' => 'app\models\User',
19             'enableAutoLogin' => true,
20         ],
21         'errorHandler' => [
22             'errorAction' => 'site/error',
23         ],
24         'mailer' => [
25             'class' => 'yii\swiftmailer\Mailer',
26             // send all mails to a file by default. You have to set
27             // 'useFileTransport' to false and configure a transport
28             // for the mailer to send real emails.
29             'useFileTransport' => true,
30         ],
31         'log' => [
32             'traceLevel' => YII_DEBUG ? 3 : 0,
33             'targets' => [
34                 [
35                     'class' => 'yii\log\FileTarget',
36                     'levels' => ['error', 'warning'],
37                 ],
38             ],
39         ],
40         'db' => require(__DIR__ . '/db.php'),
41     ],
42     /*
43     |-----|
44     | Application Timezone

```

Environments

- **Laravel**

- Loads environment variables into `$_ENV` and `$_SERVER`
- `.env` file out of repository

- **Yii**

- Logic in configuration files, in addition to bootstrap files
- `YII_ENV_DEV` for enabling debug features
- Can get complicated with multiple environments

- composer.json
- composer.lock
- gulpfile.js
- package.json
- phpunit.xml
- readme.md
- server.php
- yii2
 - assets
 - commands
 - config
 - console.php
 - db.php
 - params.php
 - web.php
 - controllers
 - mail
 - models
 - runtime
 - tests
 - vendor
 - views
 - web
 - assets
 - css
 - favicon.ico
 - index-test.php
 - index.php
 - robots.txt
 - .bowerrc
 - .gitignore
 - composer.json
 - composer.lock
 - LICENSE.md

```
index.php
21
22 require __DIR__.'../bootstrap/autoload.php';
23
24 /*
25 |-----
26 | Turn On The Lights
27 |-----
28 |
29 | We need to illuminate PHP development, so let us turn on the lights
30 | This bootstraps the framework and gets it ready for use, then it
31 | will load up this application so that we can run it and send
32 | the responses back to the browser and delight our users.
33 |
34 */
35
36 $app = require_once __DIR__.'../bootstrap/app.php';
37
38 /*
39 |-----
40 | Run The Application
41 |-----
42 |
43 | Once we have the application, we can handle the incoming request
44 | through the kernel, and send the associated response back to
45 | the client's browser allowing them to enjoy the creative
46 | and wonderful application we have prepared for them.
47 |
48 */
49
50 $kernel = $app->make(Illuminate\Contracts\Http\Kernel::class);
51
52 $response = $kernel->handle(
53     $request = Illuminate\Http\Request::capture()
54 );
55
56 $response->send();
57
```

```
index.php
1 <?php
2
3 // comment out the following two lines when deployed to production
4 defined('YII_DEBUG') or define('YII_DEBUG', true);
5 defined('YII_ENV') or define('YII_ENV', 'dev');
6
7 require(__DIR__ . '/../vendor/autoload.php');
8 require(__DIR__ . '/../vendor/yiisoft/yii2/Yii.php');
9
10 $config = require(__DIR__ . '/../config/web.php');
11
12 (new yii\web\Application($config))->run();
13
```

Controllers & Routes

- **Laravel**

- Resource Controllers, Implicit Controllers
- Route Grouping, Name spacing
- Model binding

- **Yii**

- Controller mapping, 3rd party modules
- Actions - errors, views, etc
- Enforced moduleId/controllerId/actionId structure

MeetupController.php

```
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6
7 use App\Http\Requests;
8 use App\Http\Controllers\Controller;
9
10 class MeetupController extends Controller
11 {
12     public function getIndex()
13     {
14         return 'We are here.';
15     }
16
17     public function getPresenter($name = "")
18     {
19         return [
20             'presenter' => $name,
21             'time' => time()
22         ];
23     }
24
25     public function anyView() // reponds to any http verb
26     {
27         return view('meetup');
28     }
29
30     public function postLogout() // POST /meetup/logout
31     {
32         return 'Logout';
33     }
34 }
35
```

SiteController.php

```
1 <?php
2
3 namespace app\controllers;
4
5 use Yii;
6 use yii\filters\AccessControl;
7 use yii\web\Controller;
8 use yii\filters\VerbFilter;
9
10 class SiteController extends Controller
11 {
12     public function behaviors()
13     {
14         return [
15             'verbs' => [
16                 'class' => VerbFilter::className(),
17                 'actions' => [
18                     'logout' => ['post'],
19                 ],
20             ],
21         ];
22     }
23
24     public function actionIndex() // GET /site/index
25     {
26         return $this->render('index');
27     }
28
29     public function actionLogout() // POST /site/logout
30     {
31         Yii::$app->user->logout();
32
33         return $this->goHome();
34     }
35 }
```


Active Record

- **Both**

- Relations, events (Creating, Saving, Deleting)
- Validation

- **Laravel**

- JSON Castable
- Built in Soft Delete

- **Yii**

- Scenarios

Views

- **Both**

- Layouts, Partial

- **Yii**

- Blocks
- Meta Injection
- View Events

- **Laravel**

- Blade Template Engine
- Sections

main.php

```
1 <?php
2
3 /* @var $this \yii\web\View */
4 /* @var $content string */
5
6 use yii\helpers\Html;
7 use yii\bootstrap\Nav;
8 use yii\bootstrap\NavBar;
9 use yii\widgets\Breadcrumbs;
10 use app\assets\AppAsset;
11
12 AppAsset::register($this);
13 ?>
14 <?php $this->beginPage() ?>
15 <!DOCTYPE html>
16 <html lang="<?= Yii::$app->language ?>">
17 <head>
18     <meta charset="<?= Yii::$app->charset ?>">
19     <meta name="viewport" content="width=device-width, initial-scale=1">
20     <?= Html::csrfMetaTags() ?>
21     <title><?= Html::encode($this->title) ?></title>
22     <?php $this->head() ?>
23 </head>
24 <body>
25 <?php $this->beginBody() ?>
26 <div class="wrap">
27     <div class="container">
28         <?= $content ?>
29     </div>
30 </div>
31 <footer class="footer">
32     <div class="container">
33         <p class="pull-right"><?= Yii::powered() ?></p>
34     </div>
35 </footer>
36 <?php $this->endBody() ?>
37 </body>
38 </html>
39 <?php $this->endPage() ?>
40
```

app.blade.php

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4     <meta charset="utf-8">
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6     <meta name="csrf-token" content="{{ csrf_token() }}">
7     <title>{{ $title or 'Meetup' }}</title>
8     <!--[if lte IE 8]><script src="https://oss.maxcdn.com/html5shiv/3.7.2/html5shiv.min.js"></script>
9 </head>
10 <body>
11     @yield('content')
12 </body>
13 </html>
```

Widgets

- **Yii**

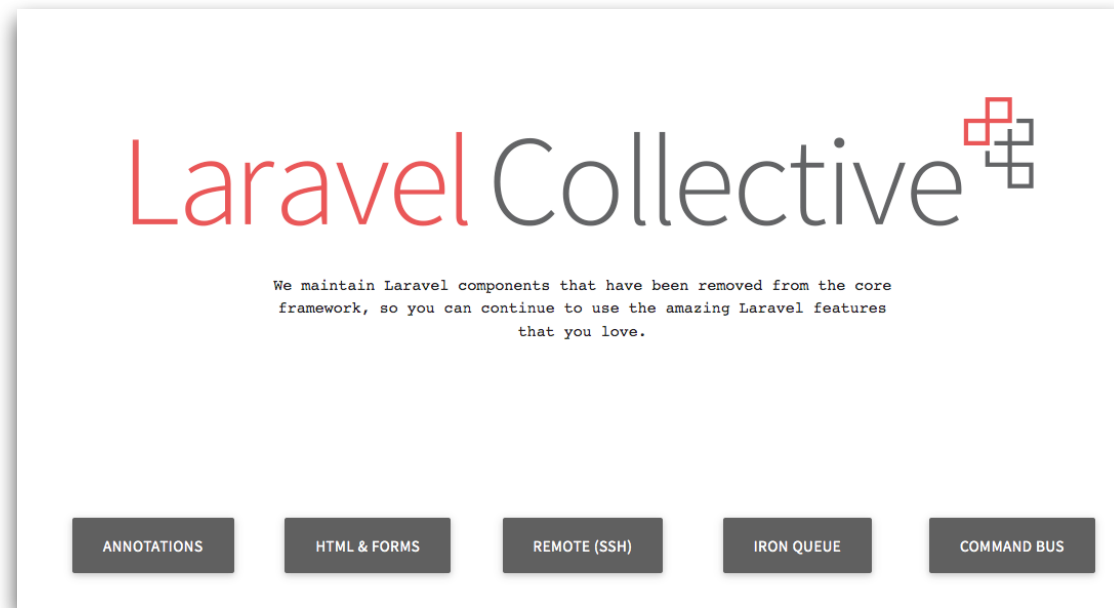
- DetailedView, ListView, GridView
- PJAX
- ActiveForm, Field
- Bootstrap/Regular
- Very powerful

- **Laravel**

- Moved to LaravelCollective (out of core)
- ActiveForm, Field

Widgets - Laravel

- Laravel Collective
 - Contains old functionality
 - ActiveForm
 - 3rd party
- Alternatives?
 - View Composers/Creators
 - Extend Blade



Widgets - Yii

- Very powerful widgets
- Listing data
- Sorting/Searching data
- Displaying data
- Tons of options
- Not all documented
- Asset Bundle
 - Inserting js/css as needed

```
<?= GridView::widget([
    'dataProvider' => $dataProvider,
    'columns' => [
        'id',
        'description',
        [
            'class' => 'yii\grid\ActionColumn',
            'controller' => 'items',
            'template' => '{update} {delete}'
        ],
    ],
]); ?>
```

</div>

Modules & Packages

- **Yii**

- Advanced / Basic template
- AuthClient
- Twig
- Debug
- mongoDB / elastic search

- **Laravel**

- Socialite
- Cashier
- Envoy
- Spark
- Lumen

Power Packed

Laravel is amazing out of the box, but there's more to explore! Let **Cashier** make subscription billing painless, or use **Socialite** to authenticate with Facebook, Twitter, and more.

yii2-queue

Yii 2.0 Queue Extension

Updated 19 days ago

yii2-elasticsearch

Yii 2 Elasticsearch extension.

Updated 20 days ago

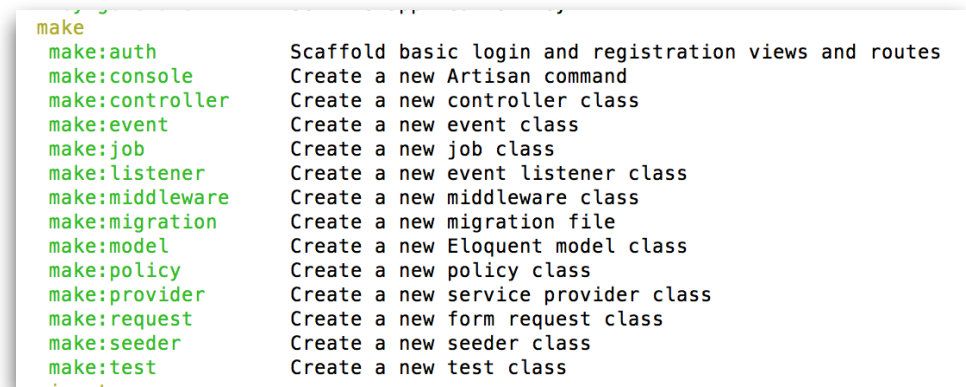
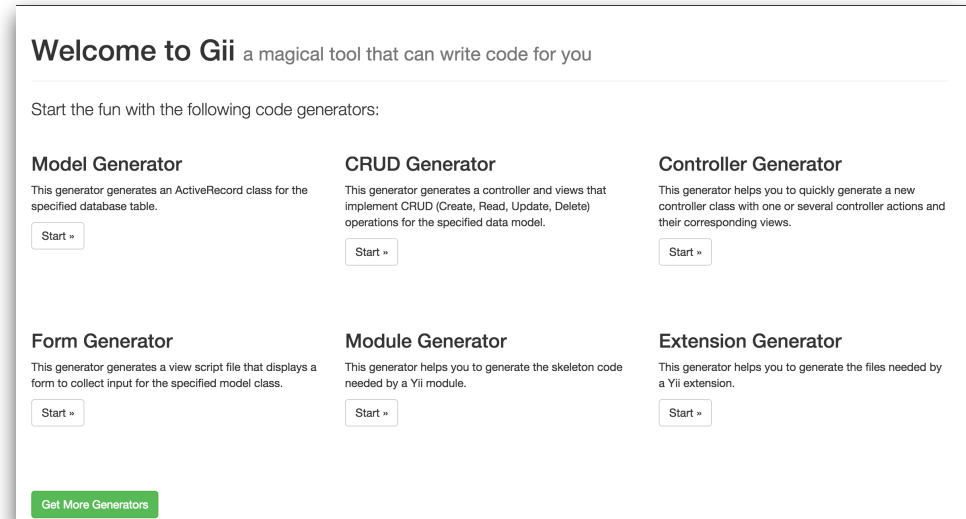
yii2-twig

Yii 2 Twig extension.

Updated 24 days ago

Boilerplate & Code Generation

- **Yii**
 - Gii
 - CRUD, Models, Controller
 - Extensions, Forms, Modules
 - Enforce pattern
 - GUI and CLI
- **Laravel**
 - *artisan make*
 - most stubs for file placement
 - No GUI



Code Generation - Yii

<http://meetup.yii/>

Quick little demo using Gii (`meetup_admin / meetup`)

Welcome to Gii a magical tool that can write code for you

Start the fun with the following code generators:

Community - Laravel

- Laracasts
- Jeffrey Way
- Graham Campbell
- Homestead - Vagrant
- Forge - Provision
- Envoyer - Deployments

LaraJobs

It's Kinda Like Netflix for Developers.

Learn practical, modern web development, through expert screencasts by **Jeffrey Way**.

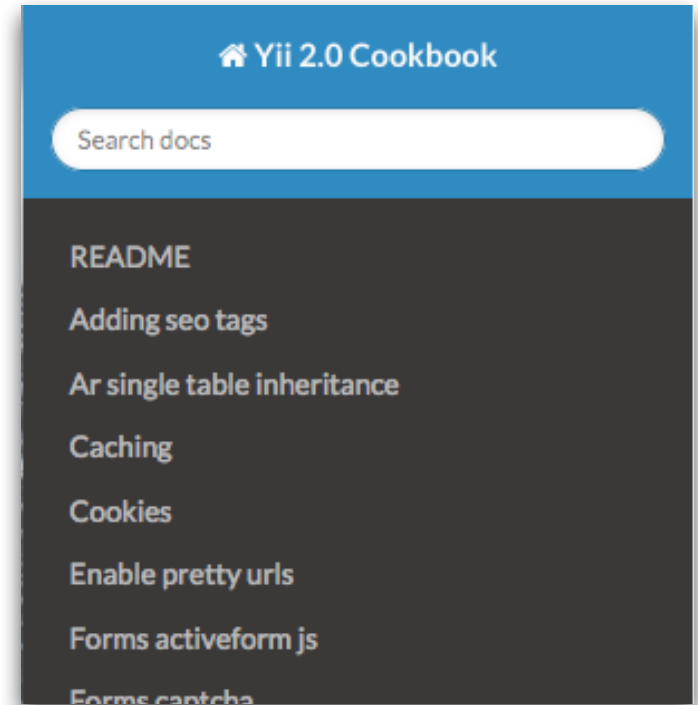
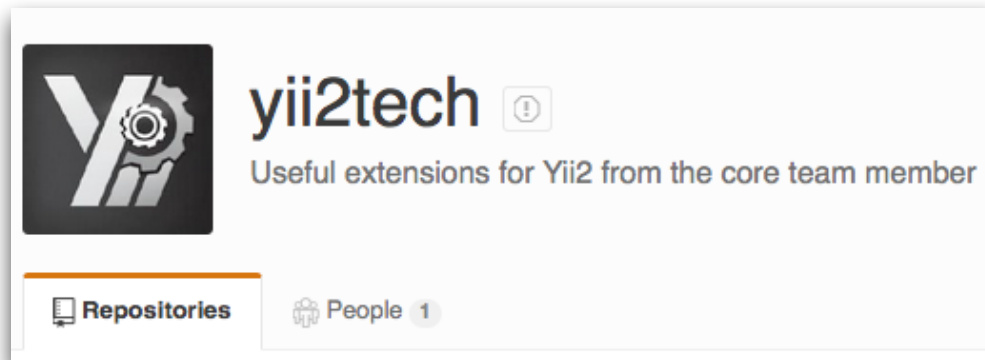
Video tutorials are often boring. These aren't.

Zero Downtime PHP Deployment

Deployments you've only dreamed about. Zero downtime. Zero fuss.

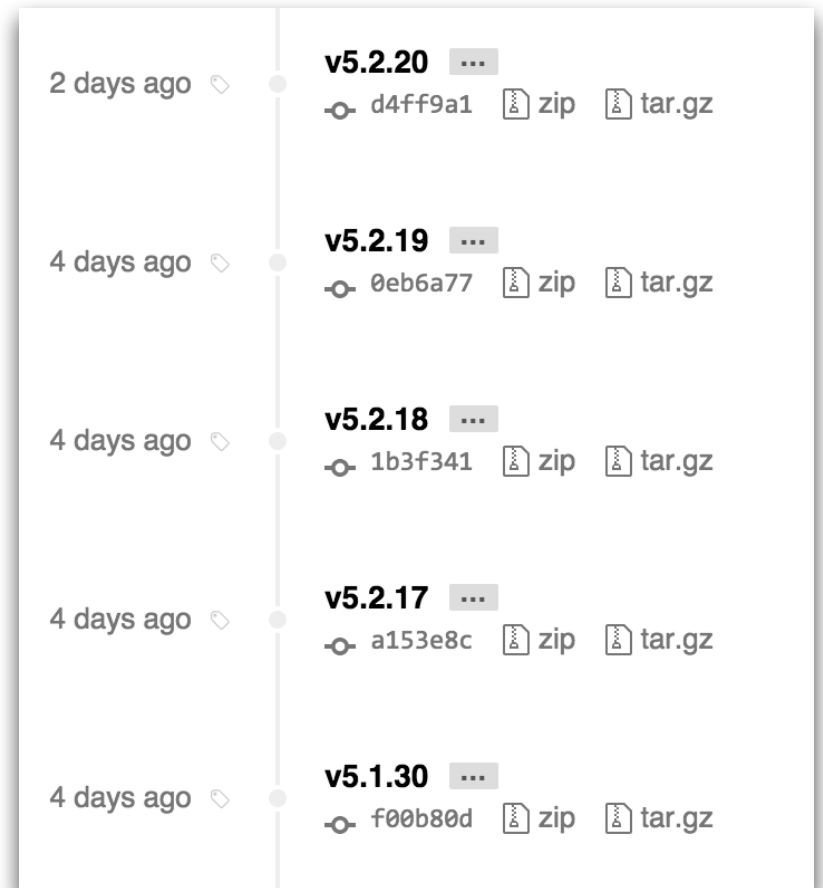
Community - Yii

- Yii Website
- Yii2Tech
- Yii2 Cookbook
- Outdated, mix of 1.1 / 2.0
- Diverse Dev Team



Versioning - Laravel

- Lots of patch updates
 - v4.x - 60 patch updates
 - 1 year 9 months
 - v5.0 - 35 patch updates
 - 4 months
 - v5.1 - 30 patch updates
 - 6 months - LTS
 - v5.2 - 17 patch updates
 - current



Versioning - Yii

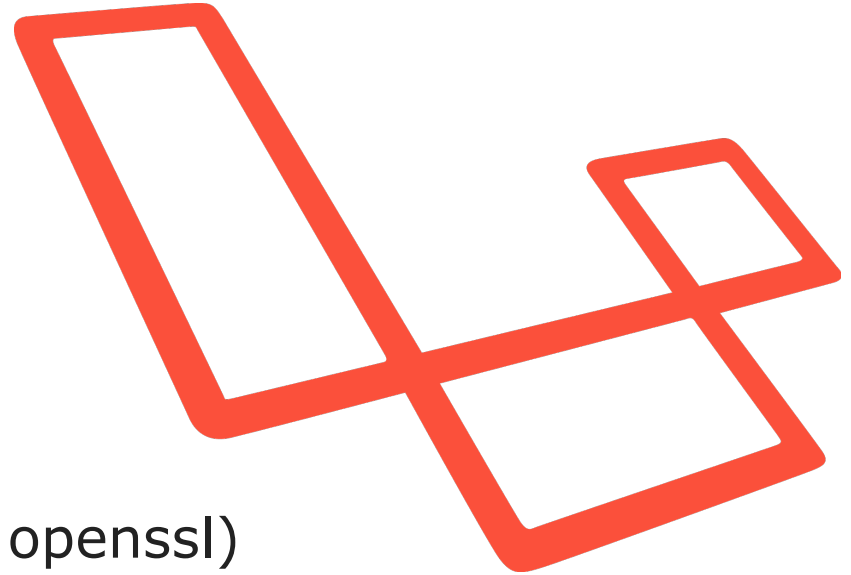
- Moderate patch updates
 - v1.0 - 12 patch updates
 - 1 year 10 months
 - v1.1 - 16 patch updates
 - 4 years 9 months
 - v2.0 - 7 patch updates
 - current

- Supported long term
- Releases every few months



Conclusion - Laravel

- Semantic Sugar
- Large Community
- Lessons via Laracasts
- Quick Updates
- Requires PHP \geq 5.5.9
- A few needed extensions (mb, pdo, openssl)
- Freedom to organize, design on own



Conclusion - Yii

- Built to last
- Enforced, but extendable format
- Mixed plugins/documentation
- PHP \geq 5.4.0
- Hard to test / mock
- Helpful standard to ease multi person projects



sourcetoad.com
info@sourcetoad.com
813.964.7349



sourcetoad
DEVELOPMENT STUDIO