

Criando APIs com Silex

API

O que é?

"[] conjunto de rotinas e padrões estabelecidos por um software para a utilização das suas funcionalidades por aplicativos que não pretendem envolver-se em detalhes da implementação do software, mas apenas usar seus serviços []"

Wikipedia

Por que criar?

Múltiplas interfaces
(web, mobile, CLI)



O uFun é a maneira mais divertida de você encontrar e escutar seus sound memes favoritos. Baixe agora mesmo o aplicativo para iOS ou Android e divirta-se compartilhando os sons com seus amigos.



Tem alguma sugestão para o uFun? Mande um e-mail para ufun@coderockr.com

Integração com outros serviços da sua empresa
Venda de recursos
(Amazon, Parse, Pusher, Filepicker, etc)

Exemplo

RestBeer

API de informações sobre cerveja
Múltiplos formatos de resposta (json, html)

<http://restbeer.com/cerveja/>

<http://restbeer.com/estilo/>

<http://restbeer.com/cerveja/Guinness>

<http://restbeer.com/cerveja/Heineken>

<http://restbeer.com/estilo/Pilsen>

<http://restbeer.com/estilo/Stout>

Silex

Micro-framework para PHP 5.3 (e superiores) construído com base nos componentes do Symfony e inspirado no Sinatra (ruby)

Instalando

Criando o composer.json

```
{  
    "require": {  
        "silex/silex": "1.0.*"  
    },  
    "minimum-stability": "dev"  
}
```

Instalando as dependências

```
curl -s http://getcomposer.org/installer | php  
php composer.phar install
```

Mostrando as
cervejas


```
<?php
use Silex\Application;

//loader do Composer
$loader = require_once __DIR__.'./vendor/autoload.php';

$app = new Application();

$cervejas = array(
    'marcas' => array('Heineken', 'Guinness', 'Skol',
        'Colorado'),
    'estilos' => array('Pilsen' , 'Stout')
);

$app->get('/cerveja', function () use ($cervejas) {
    return implode(',', $cervejas['marcas']);
});

$app->get('/estilo', function () use ($cervejas) {
    return implode(',', $cervejas['estilos']);
});
$app->run();
```

Testando com o servidor
do PHP (5.4 ou superior)

```
php -S localhost:8080
```

Mostrando uma em
específico

```
$app->get('/cerveja/{id}', function ($id) use ($cervejas) {  
    if ($id == null) {  
        return implode(',', $cervejas['marcas']);  
    }  
  
    $key = array_search($id, $cervejas['marcas']);  
    if ($key === false) {  
        return 'Não encontrada';  
    }  
    return $cervejas['marcas'][$key];  
})->value('id', null);
```

Formatando o
resultado

<https://gist.github.com/eminetto/510d63a085d775fe31af>

Simplificando

<https://gist.github.com/eminetto/42eb692f19a51a356cac>

Autenticando

```
$app->before(function (Request $request) use ($app) {  
    if( ! $request->headers->has('authorization')){  
        return new Response('Unauthorized', 401);  
    }  
  
    $clients = require_once 'config/clients.php';  
    if (!in_array($request->headers->get('authorization'),  
array_keys($clients))) {  
        return new Response('Unauthorized', 401);  
    }  
});
```

```
<?php
//config/clients.php
return ['api-1', 'api-2'];
```

Adicionando uma
cerveja

<https://gist.github.com/eminetto/21cc7d793d018f128616>

Alterando e excluindo
(exercício)

```
$app->put('/cerveja/{id}', function (Request $request, $id) use ($app) {  
  
});
```

```
$app->delete('/cerveja/{id}', function (Request $request, $id) use ($app) {  
  
});
```


Formatando em HTML

```
{  
  "require": {  
    "silex/silex": "1.0.*",  
    "twig/twig": "*"   
  }  
}
```

```
<!-- views/content.twig -->
{% for c in content %}
    {{c}}<br>
{% endfor %}
```

[https://gist.github.com/eminetto/
23b32eb7e88a0de891e7](https://gist.github.com/eminetto/23b32eb7e88a0de891e7)

Referências

<http://www.slideshare.net/hhamon/silex-meets-soap-rest>

<http://www.slideshare.net/Alganet/rest-faa-o-servio-direito>

<http://silex.sensiolabs.org>

<https://github.com/eminetto/silex-sample>

<https://github.com/Coderockr/SOA-Server>

<https://github.com/Coderockr/SOA-Client>