



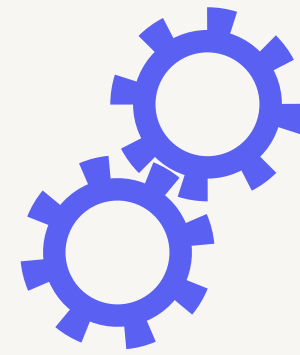
**Session**



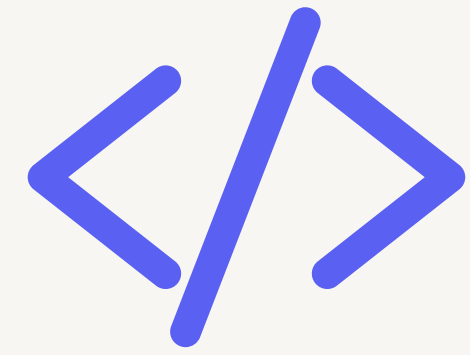
**Error  
Handler**



**C'est quoi une  
Session ?**



**Configuration**



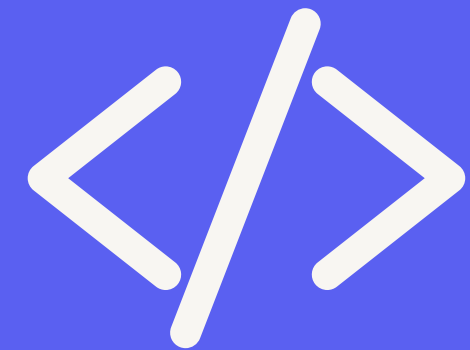
**Utilisation**



**Gestion des  
erreurs**

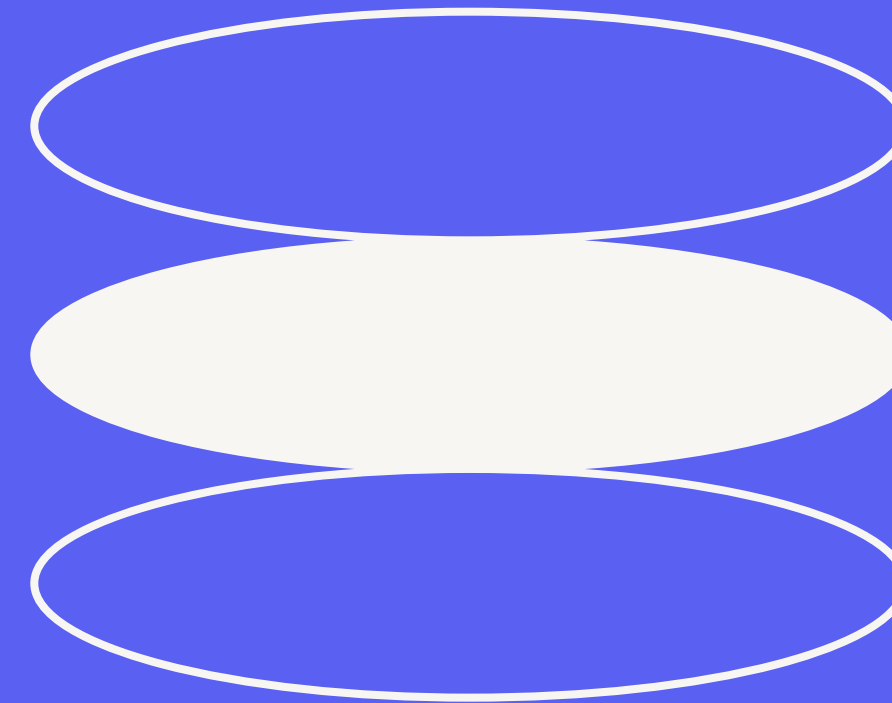


**personnalisées  
les erreurs**



**Exemple d'un  
modèle**

# C'est quoi une Session ?



- Session est un object fournit par symfony pour stocker les information
- Les sessions démarrent automatique avec chaque innteraction
- Les sessions sont crée par le composant HttpFoundation

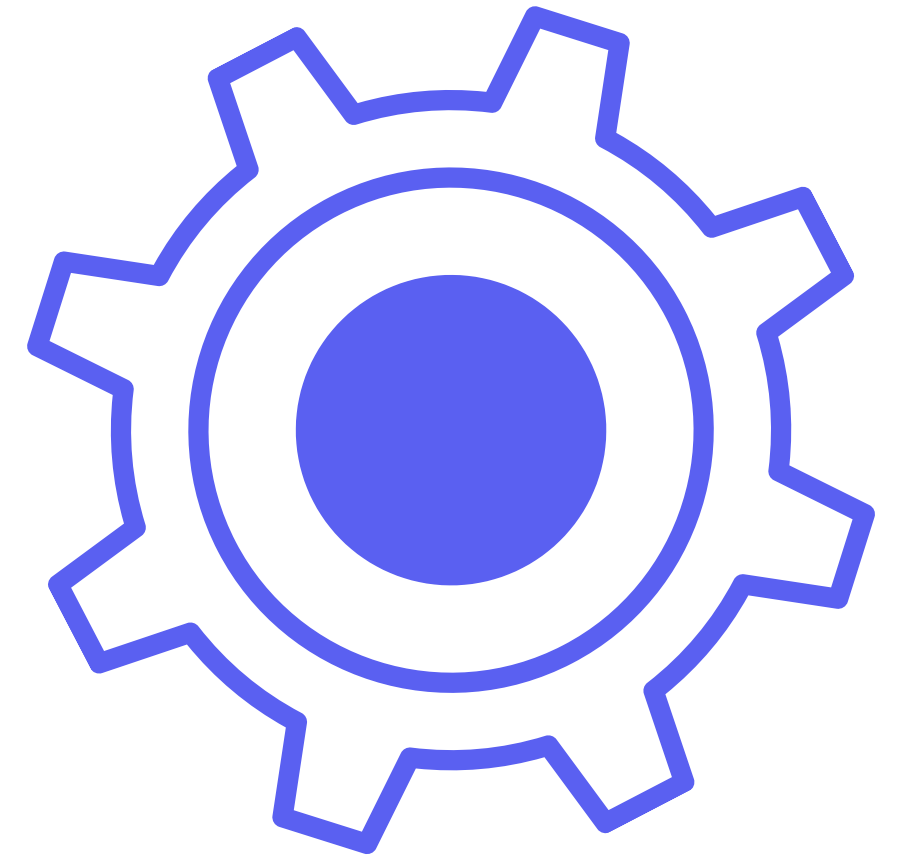
# Configuration

YAML XML PHP

```
# config/packages/framework.yaml
framework:
  session:
    # enables the support of sessions in the app
    enabled: true
    # ID of the service used for session storage.
    # NULL means that Symfony uses PHP default session mechanism
    handler_id: null
    # improves the security of the cookies used for sessions
    cookie_secure: 'auto'
    cookie_samesite: 'lax'
```

YAML XML PHP

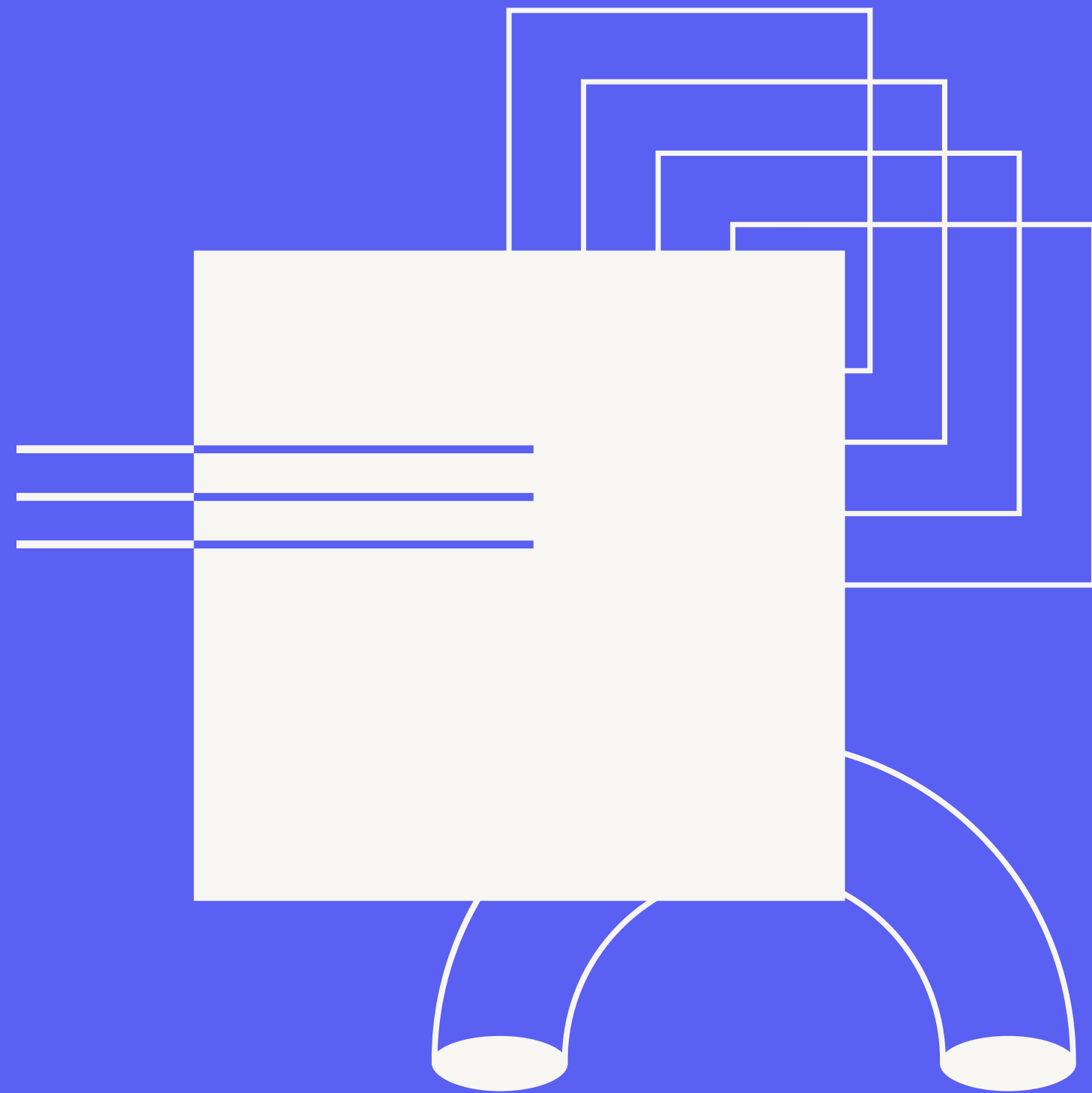
```
1 # config/packages/framework.yaml
2 framework:
3     session:
4         # ...
5         handler_id: 'session.handler.native_file'
6         save_path: '%kernel.project_dir%/var/sessions/%kernel.environment%'
```



# UTILISATION

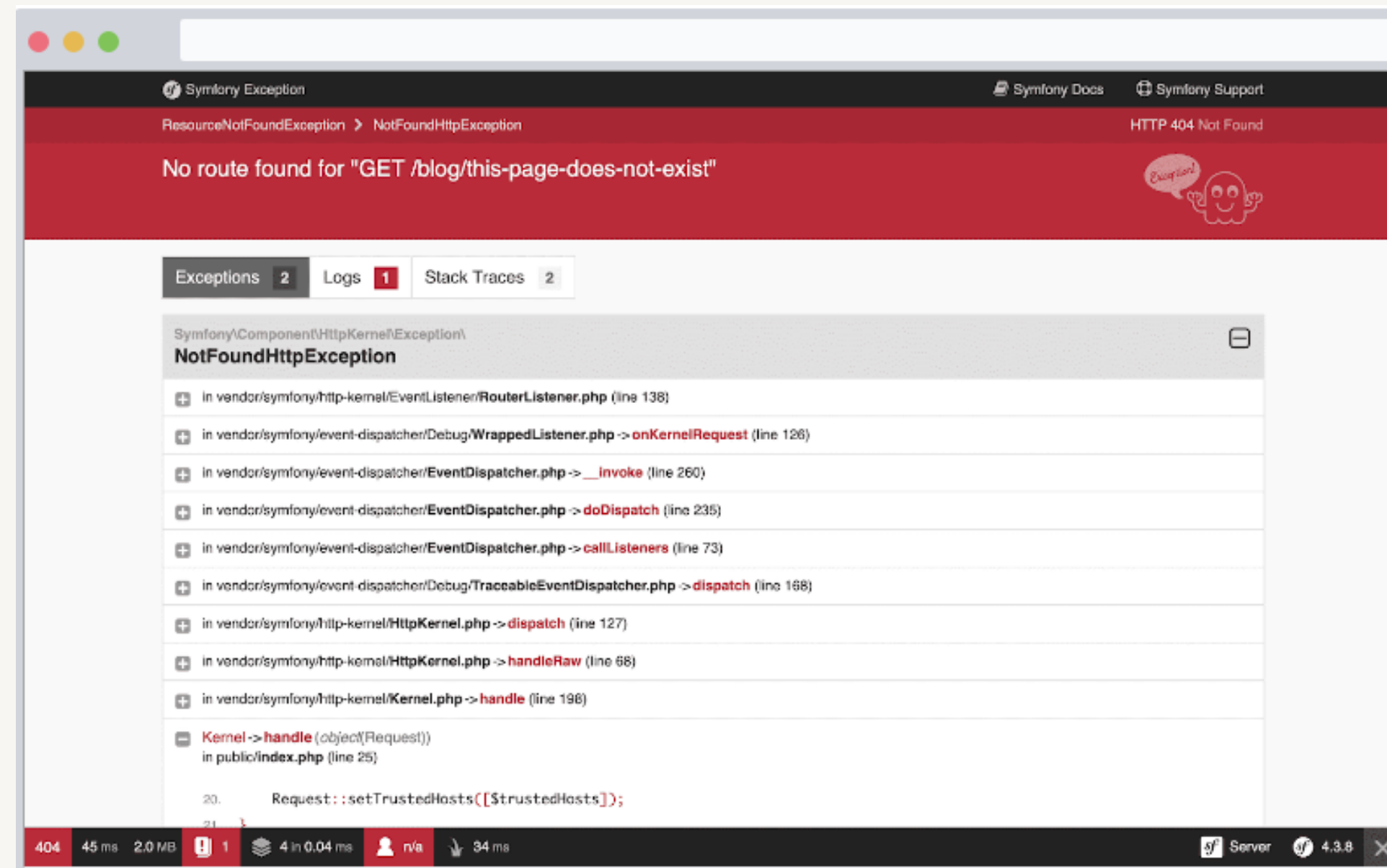
```
1 use Symfony\Component\HttpFoundation\RequestStack;
2
3 class SomeService
4 {
5     private $requestStack;
6
7     public function __construct(RequestStack $requestStack)
8     {
9         $this->requestStack = $requestStack;
10    }
11
12    public function someMethod()
13    {
14        $session = $this->requestStack->getSession();
15
16        // stores an attribute in the session for later reuse
17        $session->set('attribute-name', 'attribute-value');
18
19        // gets an attribute by name
20        $foo = $session->get('foo');
21
22        // the second argument is the value returned when the attribute doesn't exist
23        $filters = $session->get('filters', []);
24
25        // ...
26    }
27 }
```

# UTILISATION



# Gestion des erreurs

Dans les applications Symfony, toutes les erreurs sont traitées comme des exceptions, qu'il s'agisse d'une erreur 404 Not Found ou d'une erreur fatale déclenchée en lançant une exception dans votre code



## Oops! An Error Occurred

The server returned a "404 Not Found".

Something is broken. Please let us know what you were doing when this error occurred. We will fix it as soon as possible. Sorry for any inconvenience caused.

## **Remplacement de model par défaut**

modification de style et  
contenu des pages par défaut

## **Creation d'un nouveau normalisateur**

Modification du contenu de la  
sortie d'erreur non-HTML

## **Remplacement de controleur d'erreur par défaut**

Modification du logique utilisee  
par symfony pour les page  
d'erreur

## **Utilisation de l'événement kernel.exception**

Control total de gestion des  
exeptions.



## 1 - twig-pack

pour remplacer les modèles d'erreur par défaut. TwigBundle et TwigBridge doivent être installés

## 2 - Creation du fichier d'erreur

créez un nouveau error404.html.twig modèle situé à templates/bundles/TwigBundle/Exception/:

## 3 - Configuration pour le mode de developement

la valeur par défaut ErrorController vous permet de prévisualiser vos pages d'erreur pendant le développement. Pour utiliser cette fonctionnalité, vous devez charger certaines routes spéciales fournies par FrameworkBundle

```
1  {# templates/bundles/TwigBundle/Exception/error404.html.twig #}
2  {% extends 'base.html.twig' %}
3
4  {% block body %}
5      <h1>Page not found</h1>
6
7      <p>
8          The requested page couldn't be located. Checkout for any URL
9          misspelling or <a href="{{ path('homepage') }}">return to the homepage</a>.
10     </p>
11 {% endblock %}
```

**MERCI !**