

# Évolutions techniques & corrections de fonctionnalités

<b>1. Evolutions</b>	<b>2</b>
1.1. Upgrade Symfony 3.1 LTS => Symfony 5.2 Stable	2
1.2. Pages d'erreurs personnalisées	2
1.3. Sélection du rôle d'un utilisateur	2
1.4. Sécurité	2
1.4.1. Security.yaml	2
1.5. Affichage du nom de l'auteur de la tâche et de la date de création	3
1.6. Affichage du rôle de l'utilisateur	3
1.7. Évolutions du code source	3
1.8. Bonne pratiques	3
1.9. Mise en place d'un environnement de test	4
<b>2. Corrections de bugs</b>	<b>4</b>
2.1. Evolutions à prévoir	5

A la reprise du projet en 2021, la version de Symfony 3.1 est dépréciée, plus maintenue, et ne fonctionne plus comme on l'attend.

Il a donc fallu tout d'abord migrer vers une version 3 LTS à savoir la 3.4, pour permettre de voir s'afficher un résultat.

# 1. Evolutions

## 1.1. Upgrade Symfony 3.1 LTS => Symfony 5.2 Stable

- Correction des dépréciations sur la version 3.1
- Update composer.json jusqu'aux dernières versions en cours (3.4)
- Corrections d'anomalies ou dépendances
- Mise à jour jusqu'aux versions 4.\* de Symfony
- Corrections d'anomalies ou dépendances
- Mise à jour jusqu'aux versions 5.2.\* de Symfony
- Adapter le code pour respecter les bonnes pratiques des dernières versions de php / symfony / twig
- Supprimer 'Action' du nom des méthodes pour nouer avec les conventions des dernières versions de symfony et php

## 1.2. Pages d'erreurs personnalisées

- 403
- 404
- Toutes les autres (500, ...)

## 1.3. Sélection du rôle d'un utilisateur

- Possibilité de choisir entre administrateur ou utilisateur courant lors de la création ou modification d'un utilisateur par un administrateur
- Seul un administrateur a accès à la route /users

## 1.4. Sécurité

### 1.4.1. Security.yaml

- Roles
  - Ajout du rôle **ROLE\_ADMIN**
- Restrictions
  - /users/\*
    - Uniquement les administrateurs
- Voter
  - TaskVoter

- `canRemove`
  - Vérifie si l'utilisateur qui supprime la tâche est bien l'auteur de la tâche en question
- `canRemoveTaskWithoutAuthor`
  - Vérifie si l'utilisateur qui supprime la tâche sans auteur est administrateur

## 1.5. Affichage du nom de l'auteur de la tâche et de la date de création

- Les tâches déjà créées n'ayant pas d'auteur affichent 'Anonymous' pour auteur, les autres affichent l'utilisateur connecté ayant créé la tâche.
- On affiche également la date de création de la tâche.

## 1.6. Affichage du rôle de l'utilisateur

- Pour plus de commodité, j'ai ajouté la colonne rôle dans le tableau de la liste des utilisateurs.

## 1.7. Évolutions du code source

- Entités
  - typage des propriétés
  - User
    - Ajout de la propriété **roles**
    - Ajout des méthodes
      - `getRoles()`
      - `setRoles()`
        - On définit un tableau contenant 'ROLE\_USER' au minimum à la création d'un utilisateur
  - Task
    - Ajout des méthodes
      - `getIsDone()`
      - `setIsDone()`
      - `getAuthor()`
      - `setAuthor()`

## 1.8. Bonne pratiques

- Ajout de `php-cs-fixer`
  - `respect des standards PHP`
- Bonnes pratiques avec la version en cours (Symfony 5.2)
  - Optimisation des appels aux repositories, utilisation d'injections de dépendance, etc dans les contrôleurs

- Suppression du suffixe 'Action' dans les noms des méthodes dans les controllers

## 1.9. Mise en place d'un environnement de test

- Codacy + Code Climate
- Blackfire
- Travis-CI (Intégration continue)
- PHPUnit
  - tests fonctionnels
  - tests unitaires
  - BDD de tests
  - Installation du bundle **liip/test-fixtures-bundle**
    - permet de charger des fixtures spécifiques dans les tests fonctionnels
  - Test Coverage
    - Installation du module php-xdebug

## 2. Corrections de bugs

- Fix provider namespace dans security.yaml
- Fix erreurs de namespace dans
  - TaskController.php
- Régénération des repositories
- Fix templating
  - list.html.twig
    - affichage boutons
- Fix controller
  - TaskController.php
    - createAction
      - On vérifie si le formulaire a été soumis lorsque on arrive sur la page
      - On assigne l'auteur de la tâche en passant un User via la méthode setAuthor et getUser() du abstractController pour récupérer l'utilisateur connecté
    - EditAction
      - On vérifie si le formulaire a été soumis lorsque on arrive sur la page
  - UserController.php
    - createAction
      - On vérifie si le formulaire a été soumis lorsque on arrive sur la page

- EditAction
  - On vérifie si le formulaire a été soumis lorsque on arrive sur la page
- Fix bouton “créer un utilisateur” sur la page de login
  - On affiche le bouton uniquement si
    - connecté
    - utilisateur administrateur

## 2.1. Evolutions à prévoir

Les futures évolutions à prévoir sont listées dans les **issues** sur Github.

<https://github.com/MerciMathieu/toDoList/issues>