

# Manuel d'utilisation

Le projet a été créé grâce aux commandes d'initialisations de Symfony et respecte donc les normes et PSR de Symfony (dont la documentation est consultable ici:

<https://symfony.com/doc/current/index.html>).

De ce fait, il est simple de trouver comment participer au projet, celui-ci étant fait avec les dernières versions de Symfony.

## MVC

Tout le site est pensé en architecture Modèle-Vue-Controller:

- Les controllers (src/controller) ont été mis en place pour limiter le nombre d'exécutions / routes. Le découpage permet d'éviter d'avoir des fichiers trop confus et de permettre une meilleure visibilité.
- La Vue (src/templates) est l'ensemble des fichiers twig qui permettent l'affichage des pages. Là aussi, les différentes vues sont organisées pour permettre une meilleure visibilité.
- Les Modèles (src/repository) sont ici les repository qui nous permettent de communiquer avec la base de donnée sans devoir écrire les 'requests' directement dans les controllers.

Cette organisation permet une norme plus simple à respecter pour les implémentations futures du projet et la maintenance de celui-ci.

## Sécurité

Les routes ont été organisées de manière à rendre l'implémentation de la sécurité plus simple:

- Les routes /admin ne sont accessibles que par les utilisateurs avec le rôle Admin
- Les routes /board et /tasks ne sont accessibles que par des utilisateurs inscrit avec le rôle User
- La page d'accueil et les formulaires d'inscription et login sont accessibles à tous

Tous ces droits sont présent dans le dossier config/packages/security

```
role_hierarchy:
    - ROLE_ADMIN: ROLE_USER
# Easy way to control access for large sections of your site
# Note: Only the *first* access control that matches will be used
access_control:
    - { path: ^/admin, roles: ROLE_ADMIN }
    - { path: ^/board, roles: ROLE_USER }
    - { path: ^/tasks, roles: ROLE_USER }
```

## Authentication

L'authentification des utilisateurs se fait via la route /login, située dans Controller/Authentication/LoginController.php

Grâce à Symfony 6.2, les contrôles se font via la classe "AuthenticationUtils" qui va comparer le nom de l'utilisateur (username) et envoyer une requête pour confirmer la présence en base de données et confirmer le mot de passe qui lui est rattaché.

Les informations pour retrouver ou sont situer les noms d'utilisateurs peuvent être configurés dans le fichier "security.yaml" sous le paramètre "Providers".

(cf: documentation Symfony "[https://symfony.com/doc/current/security/user\\_providers.html](https://symfony.com/doc/current/security/user_providers.html)")

```
providers:
    # used to reload user from session & other features (e.g. switch_user)
    app_user_provider:
        entity:
            class: App\Entity\User
            property: username
```

## Inscription et base de donnée

L'inscription se fait via la route /signup située dans

/Controller/Authentication/SignupController.php et utilise le formulaire "SignupType.php" situé dans le dossier "Form", ce qui nous permet de contrôler les valeurs entrées (type, longueur min et max...) grâce à des validateurs.

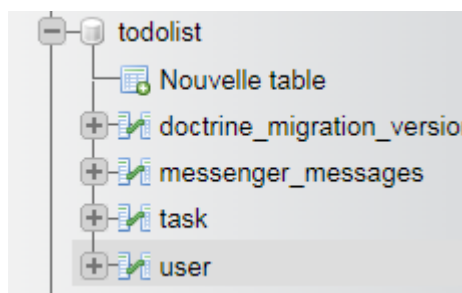
Le contrôle de l'email et du nom d'utilisateur (unicité) se fait via l'entité "User" et l'attribut "unique: true" de Doctrine.

Pour une meilleure protection des données, les mots de passe (password) sont encodé en utilisant le "hashing" par défaut de Symfony via le bundle "password-hasher" et l'injection de la classe UserPasswordHasherInterface.

(cf: <https://symfony.com/doc/current/security/passwords.html>)

```
security:
    # https://symfony.com/doc/current/security.html#registering-the-user-hashing-passwords
    password_hashers:
        Symfony\Component\Security\Core\User\PasswordAuthenticatedUserInterface: 'auto'
```

Une fois validées, les données sont enregistrées dans la base de donnée dans la table "User"



		id	email	roles	password	username
<input type="checkbox"/>	Éditer Copier Supprimer	23	Admin@app.com	["ROLE_ADMIN"]	\$2y\$13\$gVct64p33yvzQSZsEHZqZ.pn4U/DjcWzI9iZMQQCy2D...	Admin
<input type="checkbox"/>	Éditer Copier Supprimer	24	john@doe.com	["ROLE_USER"]	\$2y\$13\$vfUNzWpbie7vHdJnJHJy3OKlgOUz5SNK9dZ5Sc8cZBJ...	John

## Ajout d'un nouvel utilisateur


L'ajout d'un nouvel utilisateur est très simple grâce à l'interface Administrateur.

Une fois le projet installé et les fixtures mise en place, connectez vous avec les identifiants admin:

- nom d'utilisateur : Admin
- mot de passe: Admin

Ces informations doivent être remplacées par des identifiants plus sécurisés.

Vous aurez accès au dashboard des utilisateurs ou il sera possible de créer un nouvel utilisateur/ admin.



Bonjour admin Admin
 Admin Dashboard
Mon tableau
Se déconnecter

## Tableau de bord Administrateur

Liste des utilisateurs
Tâches orphelines

ci-dessus: le tableau de bord Administrateur

### Liste des utilisateurs

#	Nom d'utilisateur	Adresse d'utilisateur	Role	Actions
1	Admin	Admin@app.com	"ROLE_ADMIN" "ROLE_USER"	<button>Edit</button> <button>Delete</button>
2	John	john@doe.com	"ROLE_USER"	<button>Edit</button> <button>Delete</button>

Créer un utilisateur

Copyright © OpenClassrooms

ci-dessus: le bouton "Créer un Utilisateur"