

# Creation Blog

ABDALLAH Wael

January 15, 2024

## 1 Introduction

### Introduction

Le projet Symfony de création d'un blog représente une exploration approfondie des fonctionnalités du framework Symfony dans le cadre du développement d'une application web dynamique et interactive. Dans cette aventure, nous avons suivi un cahier des charges bien défini, mettant l'accent sur la gestion des articles, l'authentification des utilisateurs, la création d'un formulaire de contact et la gestion subséquente des messages soumis par les visiteurs.

Ce rapport vise à documenter de manière exhaustive notre parcours, depuis la conception initiale jusqu'à l'implémentation pratique, en passant par les défis rencontrés et les solutions apportées. Nous débuterons par une analyse détaillée du cahier des charges qui a guidé notre processus de développement, avant de plonger dans la représentation visuelle de la structure de notre application au moyen d'un diagramme de classe. Ensuite, nous examinerons le schéma relationnel de la base de données, explorant comment les données sont organisées et interconnectées.

Enfin, nous consacrerons une section à la résolution des problèmes rencontrés en cours de route, démontrant ainsi notre capacité à surmonter les défis inhérents au développement d'une application Symfony complexe. Ce rapport constitue un témoignage détaillé de notre parcours, offrant une vision holistique de notre projet de blog Symfony et de ses différentes composantes.

## 2 Cahier des charges

Le cahier des charges du projet Symfony Blog a été établi pour répondre aux besoins suivants :

Gestion des articles : Permettre aux utilisateurs de créer, afficher, éditer et supprimer des articles.

Formulaire de contact : Possibilité pour les visiteurs de soumettre des formulaires de contact.

Gestion des messages : Les administrateurs doivent pouvoir gérer les messages de contact soumis par les visiteurs.

## 3 Schéma relationnel de la base de données

## 4 Authentification avec le Bundle Security

L'authentification dans notre projet Symfony est mise en œuvre en utilisant le Bundle Security, un composant puissant qui offre des fonctionnalités de sécurité avancées. Voici une brève explication du processus :

Configuration de la Sécurité : La configuration de la sécurité est spécifiée dans le fichier `security.yaml`. Nous avons défini les encodages de mot de passe, les fournisseurs d'utilisateurs, et configuré des pare-feux pour contrôler l'accès aux différentes parties de l'application.

Entité Utilisateur (User) : Nous avons créé une entité User qui représente nos utilisateurs. Cette entité implémente l'interface `UserInterface` et contient les propriétés nécessaires telles que le nom d'utilisateur et le mot de passe.

Contrôleur d'Authentification : Nous avons un contrôleur dédié, `SecurityController`, qui gère les actions liées à l'authentification, telles que l'affichage du formulaire de connexion et la gestion des requêtes de connexion et de déconnexion.

blog contact	
id : int	
username : varchar(255)	
message : longtext	

blog doctrine_migration_versions	
version : varchar(191)	
executed_at : datetime	
execution_time : int	

blog article	
id : int	
title : varchar(255)	
content : longtext	
image : varchar(255)	
createdat : datetime	

blog user	
id : int	
username : varchar(180)	
roles : json	
password : varchar(255)	

Vues Twig : Les vues Twig sont utilisées pour rendre les formulaires de connexion. Elles intègrent des mécanismes pour gérer les erreurs d'authentification et afficher des messages pertinents aux utilisateurs.

Ce processus global assure une gestion sécurisée de l'authentification dans notre application Symfony, offrant une expérience utilisateur fluide tout en garantissant la sécurité des données sensibles

## 5 Formulaire de Contact

Affichage du Formulaire : Nous avons créé une interface utilisateur conviviale qui présente un formulaire de contact, accessible à tous les visiteurs de notre blog. Ce formulaire est conçu de manière intuitive, incitant les utilisateurs à fournir les informations nécessaires.

Validation des Données : Avant d'envoyer le formulaire, nous avons mis en place des mécanismes de validation pour garantir que les données fournies sont complètes et correctes. Cela inclut la vérification des champs requis, le format des adresses e-mail, et d'autres critères spécifiques à chaque champ.

Traitement Côté Serveur : Une fois le formulaire soumis, les données sont traitées côté serveur, généralement dans un contrôleur Symfony dédié. Nous nous assurons que les données respectent les règles de validation et sommes prêts à les utiliser pour la suite du processus.

Envoi du Formulaire : Les données validées sont ensuite utilisées pour générer un e-mail de notification ou sont stockées dans la base de données, selon le cas d'utilisation. L'envoi de cet e-mail peut être configuré pour informer les administrateurs du site de la nouvelle soumission.

## 6 Problemes

le seule probleme que j'ai trouve est un probleme de versions . au debut j'ai suivi un video de symfony version 4 et moi j'ai un symfony 6 donc il ya beaucoup de commande qui ont change donc j'ai essaye de translater les commandes et trouver les equivalents .