

## Description du Projet

**Nom du Projet :** I can fly

**Type de Projet :** Plateforme de réservation de vols spatiaux

**Technologie :** Symfony

**Objectif :**

Créer une plateforme interactive permettant aux utilisateurs de réserver des vols spatiaux, consulter les détails des missions et suivre leurs réservations en temps réel. La gestion des rôles et permissions garantit une séparation claire des responsabilités entre clients, opérateurs, et administrateurs.

---

## Objectifs du Projet

1. Permettre aux clients de réserver des sièges pour des missions spatiales.
  2. Fournir aux opérateurs un moyen d'ajouter et de gérer les missions.
  3. Offrir un espace administrateur pour superviser l'ensemble des opérations.
  4. Intégrer des APIs externes (SpaceX, paiement).
  5. Assurer une sécurité robuste avec des rôles et une authentification sécurisée.
  6. Mettre en place un suivi des réservations en temps réel.
- 

## Fonctionnalités Principales

### 1. Gestion des Utilisateurs

- **Rôles :**
  - **Client :** Peut réserver des vols et gérer ses réservations.
  - **Opérateur :** Peut ajouter, modifier ou supprimer des missions et gérer les réservations.
  - **Administrateur :** Supervise les utilisateurs, les réservations et les statistiques.
- **Authentification Sécurisée :**
  - Login/mot de passe.
  - 2FA (Authentification à Deux Facteurs) / ~ SSO
- **Permissions :**
  - Utilisation d'un voter personnalisé pour contrôler les accès et actions spécifiques.

### 2. Réservation de Vols Spatiaux

- **Exploration des Missions :**
  - Liste des vols disponibles (Starship, Falcon 9, Dragon, etc.).

- Détails des missions :
  - Destination (orbite terrestre, Lune, Mars).
  - Nombre de sièges disponibles.
  - Prix par siège.
  - Date et durée du vol.
- **Système de Réservation :**
  - Sélection du vol et choix du siège (classe économique ou premium).
  - Simulation de paiement via une API (Stripe).

### 3. Suivi des Missions en Temps Réel

- **Tableau de Bord :**
  - Compte à rebours avant le lancement.
  - Notifications en temps réel sur l'état de la mission (préparation, décollage, atterrissage).
  - Intégration de streaming vidéo via SpaceX Live.

### 4. Gestion des Entités et Relations

**Entités :**

1. **Utilisateur** (classe de base) ✓
  - **Client** (héritage) ✓
  - **Opérateur** (héritage) ✓
  - **Administrateur** (héritage) ✓
2. **Mission** (classe de base) ✓
  - **MissionTouristique** (héritage) ✓
  - **MissionScientifique** (héritage) ✓
3. **Réservation** ✓
4. **Siège** ✓
5. **Paieement** ✓
6. **Notification** ✓
7. **Vaisseau** (classe de base) ✓
  - **Starship** (héritage) ✓
  - **Falcon9** (héritage) ✓
  - **Dragon** (héritage) ✓
8. **Statistiques** ✓

**Relations :**

- **OneToMany :**
  - Missions ↔ Sièges
  - Réservations ↔ Utilisateurs
  - Missions ↔ Notifications
- **ManyToMany :**
  - Utilisateurs ↔ Missions (les clients peuvent réserver plusieurs missions)

### 5. API

- **API Interne :**
  - Endpoint pour consulter les vols disponibles.
  - Endpoint sécurisé pour réserver un siège.
  - Endpoint pour vérifier l'état des réservations.
- **API Externe :**
  - **API SpaceX :** Récupération des informations des missions.
  - **API de paiement :** Simulation de paiements avec Stripe.

## 6. Espace Administrateur

- Gestion des utilisateurs.
- Gestion des missions et réservations.
- Accès aux statistiques globales.

## 7. Documentation

- **Cahier des charges .**
- **Schéma de la base de données.**
- **Fixtures** pour initialiser les données.
- **Guide d'installation** ([README.md](#)) avec :
  - Instructions pour l'installation et le démarrage du projet.
  - Comptes de tests.
  - Processus de validation.

## 8. Tests et CI/CD

- **Tests :**
  - **Test unitaire :** Vérification de la disponibilité des sièges et du calcul des prix.
  - **Test fonctionnel :** Simulation complète d'une réservation.
- **CI/CD :**
  - Pipeline d'intégration continue (CI) :
    - Exécution des tests.
    - Analyse statique (PHPStan).
    - Linter (respect des normes de code).

## 9. Pages Différentes

1. Page d'accueil
2. Page de connexion
3. Tableau de bord utilisateur
4. Liste des missions
5. Détails d'une mission
6. Page de réservation
7. Page de paiement
8. Gestion des réservations
9. Espace admin
10. Profil utilisateur

---

## ★ Points Bonus

### 1. Fonctionnalités Avancées

- **Mode Premium :**  
Expérience VIP avec choix de repas et simulation de gravité zéro.
- **Commandes Symphony :**  
Commande pour importer automatiquement les missions SpaceX depuis l'API.
- **Gamification :**  
Système de points de fidélité pour débloquer des avantages (visites des locaux SpaceX, souvenirs numériques).

### 2. Temps Réel et Asynchrone

- **Chat Intégré :**  
Permet aux utilisateurs de poser des questions en direct sur les missions.
- **Notifications en Temps Réel :**  
Suivi des statuts de lancement en direct.

### 3. Support Multilingue

- **Internationalisation :**  
Interface disponible en plusieurs langues.

### 4. Tests Avancés

- **Tests de Mutations :**  
Pour vérifier la robustesse des tests existants.
- **TDD (Développement Piloté par les Tests) :**  
Développement basé sur des tests pour certaines fonctionnalités critiques.

### 5. DDD (Domain-Driven Design)

- Structuration du projet en suivant les principes du DDD pour une meilleure maintenabilité.