

SIMPLON.CO  
YOUCODE  
-SAFI-

# Projet Fil Rouge

## 1ère année Développement Web

---

# RÉALISATION D'UNE PLATEFORME DE LOCATION DE PROPRIÉTÉS

---

**Filière :** Développement Web

**Réalisé par :**

WASSIM YAZZA

**Encadré par :**

M. ABID Abdeladim

# **Cahier des charges pour le projet de plateforme de location de propriétés**

## ***1. Présentation du projet :***

### ***1.1 Contexte :***

Le marché mondial de la location de propriétés à court terme connaît une croissance rapide. De nombreux voyageurs recherchent des hébergements uniques tandis que les propriétaires cherchent à générer des revenus complémentaires. Ce projet a pour objectif de développer une plateforme web permettant de connecter propriétaires et voyageurs de manière simple et sécurisée.

### ***1.2 Nom du site :***

staymaroc

### ***1.3 Objectifs :***

- *Permettre aux hôtes de publier et gérer leurs propriétés.*
- *Permettre aux voyageurs de rechercher, réserver et payer en ligne des hébergements.*
- *Offrir une messagerie interne pour faciliter la communication.*
- *Offrir aux hôtes un suivi de leurs revenus et la possibilité de demander des retraits.*

- *Fournir un espace administrateur pour la modération et la supervision.*
- *Assurer la sécurité des transactions et la protection des données utilisateurs.*

## 2. Description Fonctionnelle :

### 2.1 Utilisateurs et Rôles :

<i>Rôle</i>	<i>Description</i>
<i>Visiteur</i>	<i>Parcours les propriétés, utilise la recherche</i>
<i>Voyageur</i>	<i>Réserve des propriétés, paie, laisse des avis</i>
<i>Hôte</i>	<i>Propose des propriétés, gère les réservations, suit ses gains</i>
<i>Administrateur</i>	<i>Supervise l'ensemble de la plateforme, modère contenus et transactions</i>

### 2.2 Fonctionnalités par rôle:

#### Visiteur

- Recherche de propriétés par ville, capacité, prix.
- Consultation des fiches détaillées de propriétés.

- Inscription pour devenir voyageur ou hôte.

### Voyageur

- Inscription / Connexion.
- Recherche de propriétés avec filtres.
- Réservation d'une propriété avec choix de dates.
- Paiement en ligne sécurisé (Stripe).
- Gestion des réservations (annulation possible avant validation).
- Publication d'avis sur les propriétés visitées.
- Communication avec les hôtes via messagerie.

### Hôte

- Création, modification et suppression de propriétés.
- Ajout de photos, description, disponibilités et tarifs.
- Gestion des demandes de réservation (accepter/refuser).
- Suivi des revenus.
- Demande de retrait de fonds.
- Communication avec les voyageurs.

### Administrateur

- Tableau de bord avec statistiques globales.
- Gestion des utilisateurs (activation/désactivation).

- Validation des propriétés proposées.
- Modération des avis avant publication.
- Supervision des transactions et demandes de retrait.

### ***3. Pages & Interfaces à Développer :***

#### **3.1 Pages Publiques :**

- Page d'accueil (présentation + propriétés populaires).
- Page de résultats de recherche.
- Page de détail d'une propriété.
- Page "À propos" / "Contact".

#### **3.2 Authentification:**

- Inscription (avec choix du rôle hôte ou voyageur).
- Connexion.
- Réinitialisation du mot de passe.

#### **3.3 Espace Voyageur :**

- Tableau de bord personnel.
- Liste de ses réservations.
- Détail d'une réservation.
- Formulaire de paiement en ligne.

- Historique des avis déposés.
- Boîte de réception (messagerie).

### **3.4 Espace Hôte:**

- Tableau de bord (propriétés et revenus).
- Gestion des propriétés (ajout/édition/suppression).
- Suivi des réservations reçues.
- Historique financier et demandes de retrait.
- Messagerie avec les voyageurs.

### **3.5 Espace Admin :**

- Tableau de bord global.
- Gestion des utilisateurs.
- Validation/modération des propriétés et avis.
- Gestion des transactions et retraits.

## **4. Fonctionnalités Techniques :**

### **4.1 Backend (Laravel) :**

- Framework : Laravel 10+.

- Architecture MVC avec Repository Pattern.
- Authentification Laravel + gestion des rôles via middlewares personnalisés.
- Upload et gestion des images (Laravel Storage).
- Validation serveur via Laravel Validator.
- API internes pour la messagerie et les paiements.
- Intégration de Stripe pour les paiements.
- Simulation de retrait (confirmation manuelle par admin).

#### **4.2 Base de données (PostgreSQL) :**

Tables principales :

- **users** (infos, rôle, statut).
- **properties** (description, ville, hôte\_id, prix, capacités, disponibilités).
- **property\_images** (liées aux propriétés).
- **reservations** (user\_id, property\_id, dates, statut, paiement).
- **reviews** (note, commentaire, user\_id, reservation\_id).
- **transactions** (montant, type, statut, reservation\_id).
- **withdrawals** (hôte\_id, montant, statut).
- **messages** (conversation entre utilisateurs).

#### **4.1 Frontend :**

- Blade pour les vues.
- Tailwind CSS pour le design responsive.
- JavaScript natif pour les interactions.
- Validation client côté formulaire.

### **5- Contraintes et Spécificités**

- Interface en français uniquement.
- Toutes les transactions doivent être sécurisées.
- Validation manuelle des propriétés et avis par l'administrateur.
- Aucune utilisation de bibliothèques JavaScript externes (pas de Vue.js, Livewire...).
- Aucune utilisation de packages Laravel externes type Spatie.
- Design totalement responsive (desktop, tablette, mobile).
- Paiement obligatoire avant validation d'une réservation.
- Retraits simulés (pas de transfert bancaire réel).
- Monnaie unique : MAD (Dirham Marocain).



## 6- Planning de Réalisation

Semaine	Tâches
1	Configuration du projet Laravel, architecture, auth, rôles.
2	Base de données, création des modèles, seeders, migrations.
3	Pages publiques : accueil, recherche, détail.
4	Espace voyageur : réservations + paiements.
5	Espace hôte : gestion propriétés + réservations
6	Système de messagerie interne.
7	Espace administrateur complet.
8	Tests, corrections, optimisations, déploiement.