

DESCRIPTION INTÉGRALE DE PRESTALIA

Plateforme de mise en relation de prestataires professionnels

VISION GLOBALE DU SYSTÈME

Prestalia est une application web de mise en relation entre :

- Des **utilisateurs (clients)** recherchant des services
- Des **prestataires professionnels** validés
- Un ou plusieurs **administrateurs**

Le système repose sur :

- Une gestion des rôles
- Une validation des prestataires
- Une gestion des services proposés
- Une recherche multicritère
- Une gestion des demandes de contact

ACTEURS DU SYSTÈME

1. Visiteur (non authentifié)

Peut :

- Rechercher des prestataires
- Voir les fiches publiques
- Créer un compte
- Demander à devenir prestataire

2. Utilisateur (client authentifié)

Peut :

- Modifier son profil
- Ajouter des prestataires en favoris
- Envoyer des demandes de contact
- Consulter son historique

3. Prestataire (validé)

Peut :

- Gérer son profil professionnel
- Ajouter / modifier / supprimer ses prestations
- Gérer ses zones d'intervention
- Recevoir et répondre aux demandes
- Suspendre son compte

4. Administrateur

Peut :

- Valider ou refuser les demandes prestataires
- Gérer les utilisateurs
- Gérer les catégories
- Gérer les signalements
- Accéder aux statistiques

STRUCTURE LOGIQUE DES DONNÉES

Voici les entités principales nécessaires.

USERS (utilisateurs)

Représente tous les comptes (clients, prestataires, admins)

Champs :

- id
- first_name
- last_name
- email (unique)
- password_hash
- phone
- role (client | prestataire | admin)
- status (active | suspended | pending_validation)
- created_at
- updated_at

- last_login_at

Relations :

- 1 utilisateur peut avoir 1 profil prestataire
- 1 utilisateur peut avoir plusieurs favoris
- 1 utilisateur peut envoyer plusieurs demandes

PROVIDER_PROFILES

Profil professionnel lié à un utilisateur

Champs :

- id
- user_id (FK → USERS)
- business_name
- description
- logo_url
- cover_image_url
- years_of_experience
- siret (optionnel)
- website
- validation_status (pending | approved | rejected)
- validation_date
- rejection_reason
- is_active (boolean)

- created_at
- updated_at

Relations :

- 1 profil prestataire appartient à 1 utilisateur
- 1 prestataire peut proposer plusieurs services
- 1 prestataire peut avoir plusieurs zones d'intervention
- 1 prestataire peut recevoir plusieurs demandes

CATEGORIES

Liste des métiers

Champs :

- id
- name
- slug
- icon
- parent_id (nullable → pour sous-catégories)
- created_at

Relation :

- 1 catégorie peut avoir plusieurs prestataires
- 1 catégorie peut avoir plusieurs services

SERVICES

Services spécifiques proposés par un prestataire

Champs :

- id
- provider_id (FK → PROVIDER_PROFILES)
- category_id (FK → CATEGORIES)
- title
- description
- price_from (optionnel)
- is_active
- created_at
- updated_at

Relation :

- 1 prestataire peut avoir plusieurs services

SERVICE_AREAS

Zones d'intervention

Champs :

- id
- provider_id (FK → PROVIDER_PROFILES)
- city
- postal_code

- region
- country
- created_at

Relation :

- 1 prestataire peut avoir plusieurs zones

FAVORITES

Favoris utilisateurs

Champs :

- id
- user_id (FK → USERS)
- provider_id (FK → PROVIDER_PROFILES)
- created_at

Relation :

- Relation many-to-many entre utilisateurs et prestataires

CONTACT_REQUESTS

Demandes envoyées à un prestataire

Champs :

- id

- sender_id (FK → USERS)
- provider_id (FK → PROVIDER_PROFILES)
- message
- status (sent | read | replied | archived)
- created_at
- updated_at

Relation :

- 1 utilisateur peut envoyer plusieurs demandes
- 1 prestataire peut recevoir plusieurs demandes

PROVIDER_APPLICATIONS

Demandes pour devenir prestataire

Champs :

- id
- user_id (FK → USERS)
- business_name
- description
- documents_url
- status (pending | approved | rejected)
- reviewed_by (FK → USERS admin)
- review_date
- rejection_reason

- created_at

Relation :

- 1 utilisateur peut faire 1 demande active

REPORTS (modération)

Signalements

Champs :

- id
- reporter_id (FK → USERS)
- provider_id (FK → PROVIDER_PROFILES)
- reason
- status (open | resolved | dismissed)
- created_at

ADMIN_LOGS

Journal des actions admin

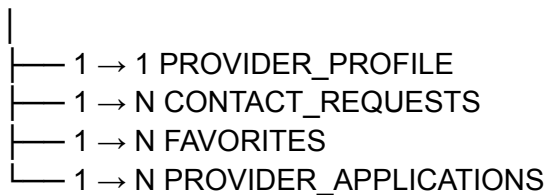
Champs :

- id
- admin_id (FK → USERS)
- action_type
- target_id

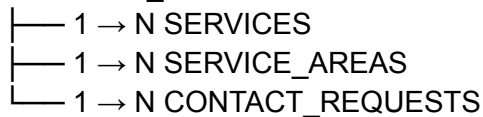
- description
- created_at

LOGIQUE DES RELATIONS (SIMPLIFIÉ)

USERS



PROVIDER_PROFILE



RÈGLES MÉTIER IMPORTANTES

- Un prestataire doit être validé avant visibilité
- Un email est unique
- Un utilisateur peut devenir prestataire
- Un prestataire peut désactiver son profil
- Une demande prestataire doit être validée par un admin
- Suppression d'un utilisateur = gestion des dépendances

ÉVOLUTIONS FUTURES PRÉVUES (à prévoir dans la BDD)

Prévoir extensibilité pour : **tu rajoute tout ça KELLY**

- ★ Avis / ratings
 - 💬 Messagerie interne
 - 📅 Abonnements
 - 📅 17 Prise de rendez-vous
 - 🔔 Notifications
 - 🌐 Multi-pays
-

VUE ERD SIMPLIFIÉE (conceptuelle)

Users



Provider_Profile



Services



Categories

Users



Contact_Requests



Provider_Profile

Users

↔ Favorites ↔ Provider_Profile

Résultat

Avec cette description on peut :

- Construire un MCD (Merise)

- Créer un diagramme UML
- Générer un schéma SQL
- Créer ton modèle Prisma / ORM
- Structurer ton backend proprement **NATHAN MD**