

-  Architecture des Bases de Données Multi-Tenants
 -  Vue d'Ensemble de l'Architecture
 - Stratégie Multi-Tenants PostgreSQL
 -  Tableau Récapitulatif des Bases de Données
 -  Architecture Visuelle des Bases
 -  YZ-PLATFORM_DB - Base Centrale
 -  Tableau des Tables de la Base Centrale
 - Tables d'Authentification Multi-Tenants
 - Tables de Facturation et Abonnements
 - Tables de Monitoring et Analytics
 -  YZ-CMD-YOOZAK_DB - Base Yoozak
 -  Tableau des Tables de la Base Yoozak
 - Tables de Gestion des Commandes
 - Tables de Gestion des Articles
 - Tables de Gestion des Clients
 - Tables de Gestion des Opérateurs
 -  YZ-TEXTILE-ENTREPRISE-A_DB - Base Textile
 -  Tableau des Tables de la Base Textile
 - Tables de Gestion des Produits Textiles
 - Tables E-commerce
 -  YZ-LOGISTIQUE-ENTREPRISE-B_DB - Base Logistique
 -  Tableau des Tables de la Base Logistique
 - Tables de Gestion des Entrepôts
 - Tables de Gestion des Stocks
 - Tables de Transport et Livraison
 -  YZ-RESTAURANT-ENTREPRISE-C_DB - Base Restaurant
 -  Tableau des Tables de la Base Restaurant
 - Tables de Gestion des Menus
 - Tables de Commandes et Livraison
 -  YZ-PHARMACIE-ENTREPRISE-D_DB - Base Pharmacie
 -  Tableau des Tables de la Base Pharmacie
 - Tables de Gestion des Médicaments
 -  YZ-ELECTRONIQUE-ENTREPRISE-E_DB - Base Électronique
 -  Tableau des Tables de la Base Électronique
 - Tables de Gestion des Produits Tech
 -  YZ-COSMETIQUE-ENTREPRISE-F_DB - Base Cosmétiques
 -  Tableau des Tables de la Base Cosmétiques
 - Tables de Gestion des Produits Beauté
 -  YZ-AUTOMOBILE-ENTREPRISE-G_DB - Base Automobile
 -  Tableau des Tables de la Base Automobile
 - Tables de Gestion des Véhicules
 -  YZ-IMMOBILIER-ENTREPRISE-H_DB - Base Immobilier
 -  Tableau des Tables de la Base Immobilier
 - Tables de Gestion des Biens
 -  YZ-EDUCATION-ENTREPRISE-I_DB - Base Éducation
 -  Tableau des Tables de la Base Éducation

- Tables de Gestion des Formations
-  YZ-SANTE-ENTREPRISE-J_DB - Base Santé
 -  Tableau des Tables de la Base Santé
 - Tables de Gestion des Patients
-  Relations entre les Bases de Données
 - Architecture de Liaison
-  Métriques et Monitoring des Bases
 - Tables de Performance
-  Avantages de cette Architecture
 - 1. Isolation des Données
 - 2. Performance Optimisée
 - 3. Maintenance Simplifiée
 - 4. Développement Flexible
-  Commandes de Gestion des Bases
 - Création d'une Nouvelle Base
 - Backup et Restauration
 - Monitoring des Performances
-  Tableau Comparatif des Bases de Données
 - Comparaison des Fonctionnalités par Base
 -  Statistiques Globales





Architecture des Bases de Données Multi-Tenants





Vue d'Ensemble de l'Architecture

Stratégie Multi-Tenants PostgreSQL

Cette architecture utilise une approche "Database per Tenant" où chaque projet sectoriel possède sa propre base de données PostgreSQL, tout en partageant une base de données centrale pour la plateforme.


Tableau Récapitulatif des Bases de Données





 Base de Données	 Projet	 Nombre de Tables	 Secteur d'Activité
YZ-PLATFORM_DB	Plateforme Centrale	15+	Multi-tenants, Authentification
YZ-CMD-YOOZAK_DB	Yoozak (Existant)	25+	E-commerce, Logistique
YZ-TEXTILE-ENTREPRISE-A_DB	Textile Entreprise A	20+	Mode, Collections, E-commerce


 Base de Données	 Projet	 Nombre de Tables	 Secteur d'Activité
YZ-LOGISTIQUE-ENTREPRISE-B_DB	Logistique Entreprise B	18+	Entrepôts, Stock, Transport
YZ-RESTAURANT-ENTREPRISE-C_DB	Restaurant Entreprise C	15+	Restauration, Livraison
YZ-PHARMACIE-ENTREPRISE-D_DB	Pharmacie Entreprise D	12+	Médical, Ordonnances
YZ-ELECTRONIQUE-ENTREPRISE-E_DB	Électronique Entreprise E	10+	High-Tech, Garanties
YZ-COSMETIQUE-ENTREPRISE-F_DB	Cosmétique Entreprise F	8+	Beauté, Conseils
YZ-AUTOMOBILE-ENTREPRISE-G_DB	Automobile Entreprise G	12+	Véhicules, Pièces, Garage
YZ-IMMOBILIER-ENTREPRISE-H_DB	Immobilier Entreprise H	10+	Biens, Visites, Contrats
YZ-EDUCATION-ENTREPRISE-I_DB	Éducation Entreprise I	12+	Formations, Étudiants
YZ-SANTE-ENTREPRISE-J_DB	Santé Entreprise J	15+	Patients, Médecins, RDV







Architecture Visuelle des Bases




 YZ-PLATFORM_DB # Base de données centrale (PostgreSQL)


-  Tables d'authentification multi-tenants
-  Tables de facturation et abonnements
-  Tables de monitoring et analytics
-  Tables de liaison entre projets




 YZ-CMD-YOOZAK_DB # Base Yoozak (PostgreSQL)

-  Tables de commandes et articles
-  Tables d'utilisateurs et opérateurs
-  Tables de KPIs et synchronisation

 YZ-TEXTILE-ENTREPRISE-A_DB # Base Textile (PostgreSQL)

-  Tables de produits textiles
-  Tables de collections et variantes
-  Tables e-commerce

 YZ-LOGISTIQUE-ENTREPRISE-B_DB # Base Logistique (PostgreSQL)

-  Tables d'entrepôts et zones
-  Tables de mouvements de stock
-  Tables de transport


[Et ainsi de suite pour chaque projet...]



YZ-PLATFORM_DB - Base Centrale



Tableau des Tables de la Base Centrale

 Catégorie	 Table	 Description	 Clés	 Champs Principaux
Authentification	tenants	Entreprises (tenants)	id, slug	nom_entreprise, plan_abonnement, statut
Authentification	users	Utilisateurs globaux	id, tenant_id	username, email, role_global
Authentification	user_sessions	Sessions multi-tenants	id, user_id	session_key, tenant_id, ip_address
Facturation	subscription_plans	Plans d'abonnement	id	nom, prix_mensuel, limite_utilisateurs
Facturation	tenant_subscriptions	Abonnements actifs	id, tenant_id	plan_id, date_debut, date_fin
Facturation	invoices	Factures	id, tenant_id	numero_facture, montant_ttc, statut
Monitoring	usage_metrics	Métriques d'utilisation	id, tenant_id	utilisateurs_actifs, stockage_utilise_gb
Monitoring	activity_logs	Logs d'activité	id, tenant_id	action, module, timestamp
Liaison	tenant_databases	Configuration des bases	id, tenant_id	nom_base, host, utilisateur
Liaison	data_sync_logs	Logs de synchronisation	id, tenant_id	table_source, operation, statut
Performance	database_metrics	Métriques des bases	id, tenant_id	taille_gb, nombre_tables, performance
Alertes	database_alerts	Alertes de base	id, tenant_id	type_alerte, severite, message

Tables d'Authentification Multi-Tenants

```
-- Table des entreprises (tenants)
CREATE TABLE tenants (
  id SERIAL PRIMARY KEY,
  nom_entreprise VARCHAR(255) NOT NULL,
  slug VARCHAR(100) UNIQUE NOT NULL,
  domaine VARCHAR(255) UNIQUE,
  plan_abonnement VARCHAR(50) NOT NULL,
  date_creation TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  date_expiration TIMESTAMP,
  statut VARCHAR(20) DEFAULT 'actif',
```

```

    limite_utilisateurs INTEGER DEFAULT 10,
    limite_stockage_gb INTEGER DEFAULT 10,
    config_json JSONB
);

-- Table des utilisateurs globaux
CREATE TABLE users (
    id SERIAL PRIMARY KEY,
    tenant_id INTEGER REFERENCES tenants(id),
    username VARCHAR(150) UNIQUE NOT NULL,
    email VARCHAR(254) UNIQUE NOT NULL,
    password_hash VARCHAR(255) NOT NULL,
    is_active BOOLEAN DEFAULT TRUE,
    date_creation TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    derniere_connexion TIMESTAMP,
    role_global VARCHAR(50) DEFAULT 'user'
);

-- Table des sessions multi-tenants
CREATE TABLE user_sessions (
    id SERIAL PRIMARY KEY,
    user_id INTEGER REFERENCES users(id),
    tenant_id INTEGER REFERENCES tenants(id),
    session_key VARCHAR(255) UNIQUE NOT NULL,
    date_creation TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    date_expiration TIMESTAMP,
    ip_address INET,
    user_agent TEXT
);

```

Tables de Facturation et Abonnements

```

-- Table des plans d'abonnement
CREATE TABLE subscription_plans (
    id SERIAL PRIMARY KEY,
    nom VARCHAR(100) NOT NULL,
    prix_mensuel DECIMAL(10,2) NOT NULL,
    prix_annuel DECIMAL(10,2) NOT NULL,
    limite_utilisateurs INTEGER NOT NULL,
    limite_stockage_gb INTEGER NOT NULL,
    fonctionnalites JSONB,
    statut VARCHAR(20) DEFAULT 'actif'
);

-- Table des abonnements actifs
CREATE TABLE tenant_subscriptions (
    id SERIAL PRIMARY KEY,
    tenant_id INTEGER REFERENCES tenants(id),
    plan_id INTEGER REFERENCES subscription_plans(id),
    date_debut TIMESTAMP NOT NULL,
    date_fin TIMESTAMP NOT NULL,
    statut VARCHAR(20) DEFAULT 'actif',
    montant DECIMAL(10,2) NOT NULL,
    methode_paiement VARCHAR(50),
    reference_paiement VARCHAR(255)
);

-- Table des factures
CREATE TABLE invoices (
    id SERIAL PRIMARY KEY,
    tenant_id INTEGER REFERENCES tenants(id),
    subscription_id INTEGER REFERENCES tenant_subscriptions(id),
    numero_facture VARCHAR(100) UNIQUE NOT NULL,
    date_emission DATE NOT NULL,
    date_echeance DATE NOT NULL,
    montant_ht DECIMAL(10,2) NOT NULL,

```

```
montant_tva DECIMAL(10,2) NOT NULL,
montant_ttc DECIMAL(10,2) NOT NULL,
statut VARCHAR(20) DEFAULT 'emise'
);
```

Tables de Monitoring et Analytics

```
-- Table des métriques d'utilisation
CREATE TABLE usage_metrics (
  id SERIAL PRIMARY KEY,
  tenant_id INTEGER REFERENCES tenants(id),
  date_mesure DATE NOT NULL,
  utilisateurs_actifs INTEGER DEFAULT 0,
  stockage_utilise_gb DECIMAL(10,2) DEFAULT 0,
  requetes_api INTEGER DEFAULT 0,
  temps_reponse_moyen_ms INTEGER DEFAULT 0
);






-- Table des logs d'activité
CREATE TABLE activity_logs (
  id SERIAL PRIMARY KEY,
  tenant_id INTEGER REFERENCES tenants(id),
  user_id INTEGER REFERENCES users(id),
  action VARCHAR(100) NOT NULL,
  module VARCHAR(100),
  details JSONB,
  ip_address INET,
  user_agent TEXT,
  timestamp TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```








YZ-CMD-YOOZAK_DB - Base Yoozak



Tableau des Tables de la Base Yoozak

 Catégorie	 Table	 Description	 Clés	 Champs Principaux
Commandes	enum_etat_cmd	États des commandes	id	nom, label, ordre, couleur
Commandes	commandes	Commandes principales	id, id_yz, num_cmd	total_ttc, client_id, etat_commande_id
Commandes	articles_commande	Articles des commandes	id, commande_id	article_id, quantite, prix_unitaire
Articles	articles	Catalogue d'articles	id, reference	nom, prix_courant, quantite_disponible
Articles	categories_articles	Catégories d'articles	id, parent_id	nom, description, ordre

 Catégorie	 Table	 Description	 Clés	 Champs Principaux
Clients	<code>clients</code>	Base clients	<code>id</code> , <code>telephone</code>	nom, prenom, email, adresse
Opérateurs	<code>opérateurs</code>	Gestion des opérateurs	<code>id</code> , <code>user_id</code>	type_opérateur, nom, specialite
Livraison	<code>livraisons</code>	Gestion des livraisons	<code>id</code> , <code>commande_id</code>	adresse, statut, date_livraison
Synchronisation	<code>google_sheet_config</code>	Config Google Sheets	<code>id</code>	url, sheet_name, active
Synchronisation	<code>sync_logs</code>	Logs de synchronisation	<code>id</code>	status, records_imported, errors
KPIs	<code>kpi_configuration</code>	Configuration des KPIs	<code>id</code>	parameter_name, category, value
Notifications	<code>notifications</code>	Système de notifications	<code>id</code> , <code>user_id</code>	message, type, statut

Tables de Gestion des Commandes

```

-- Table des états de commande
CREATE TABLE enum_etat_cmd (
  id SERIAL PRIMARY KEY,
  nom VARCHAR(100) NOT NULL,
  label VARCHAR(255) NOT NULL,
  ordre INTEGER NOT NULL,
  couleur VARCHAR(7) DEFAULT '#000000',
  description TEXT
);

-- Table principale des commandes
CREATE TABLE commandes (
  id SERIAL PRIMARY KEY,
  id_yz VARCHAR(50) UNIQUE NOT NULL,
  num_cmd VARCHAR(50) UNIQUE NOT NULL,
  origine VARCHAR(100),
  date_commande TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  date_modification TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  total_ht DECIMAL(10,2) NOT NULL,
  total_tva DECIMAL(10,2) NOT NULL,
  total_ttc DECIMAL(10,2) NOT NULL,
  adresse_livraison TEXT,
  ville_livraison VARCHAR(100),
  code_postal VARCHAR(10),
  pays_livraison VARCHAR(100),
  client_id INTEGER,
  etat_commande_id INTEGER REFERENCES enum_etat_cmd(id),
  operateur_id INTEGER,
  ville_initiale VARCHAR(100),
  ville_systeme VARCHAR(100),
  produit_initial VARCHAR(255),
  compteur INTEGER DEFAULT 0,
  notes TEXT
);

```

```
-- Table des articles de commande
CREATE TABLE articles_commande (
  id SERIAL PRIMARY KEY,
  commande_id INTEGER REFERENCES commandes(id),
  article_id INTEGER,
  quantite INTEGER NOT NULL,
  prix_unitaire DECIMAL(10,2) NOT NULL,
  prix_total DECIMAL(10,2) NOT NULL,
  remise DECIMAL(5,2) DEFAULT 0,
  notes TEXT
);
```

Tables de Gestion des Articles

```
-- Table des articles
CREATE TABLE articles (
  id SERIAL PRIMARY KEY,
  nom VARCHAR(255) NOT NULL,
  reference VARCHAR(100) UNIQUE NOT NULL,
  couleur VARCHAR(50),
  taille VARCHAR(20),
  prix_unitaire DECIMAL(10,2) NOT NULL,
  prix_achat DECIMAL(10,2),
  prix_courant DECIMAL(10,2) NOT NULL,
  categorie VARCHAR(100),
  phase VARCHAR(50),
  quantite_disponible INTEGER DEFAULT 0,
  description TEXT,
  images JSONB,
  upsell_actif BOOLEAN DEFAULT FALSE,
  prix_upsell DECIMAL(10,2),
  date_creation TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  date_modification TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);

-- Table des catégories d'articles
CREATE TABLE categories_articles (
  id SERIAL PRIMARY KEY,
  nom VARCHAR(100) NOT NULL,
  description TEXT,
  parent_id INTEGER REFERENCES categories_articles(id),
  ordre INTEGER DEFAULT 0
);
```

Tables de Gestion des Clients

```
-- Table des clients
CREATE TABLE clients (
  id SERIAL PRIMARY KEY,
  nom VARCHAR(100) NOT NULL,
  prenom VARCHAR(100) NOT NULL,
  telephone VARCHAR(20) UNIQUE NOT NULL,
  email VARCHAR(254) UNIQUE,
  adresse TEXT,
  ville VARCHAR(100),
  code_postal VARCHAR(10),
  pays VARCHAR(100) DEFAULT 'France',
  date_creation TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  date_modification TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
```






```
notes TEXT,
actif BOOLEAN DEFAULT TRUE
);
```

Tables de Gestion des Opérateurs

```
-- Table des opérateurs
CREATE TABLE operateurs (
  id SERIAL PRIMARY KEY,
  user_id INTEGER,
  type_opérateur VARCHAR(50) NOT NULL CHECK (type_opérateur IN ('CONFIRMATION', 'LOGISTIQUE',
'PREPARATION', 'ADMIN', 'LIVREUR')),
  nom VARCHAR(100) NOT NULL,
  prenom VARCHAR(100) NOT NULL,
  telephone VARCHAR(20),
  email VARCHAR(254),
  specialite VARCHAR(100),
  date_embauche DATE,
  statut VARCHAR(20) DEFAULT 'actif',
  performance_score DECIMAL(3,2) DEFAULT 0.00
);
```

YZ-TEXTILE-ENTREPRISE-A_DB - Base Textile

Tableau des Tables de la Base Textile

 Catégorie	 Table	 Description	 Clés	 Champs Principaux
Produits	produits_textile	Catalogue textile	id, reference	nom, marque, categorie, prix_ttc
Produits	variantes_textile	Variantes (taille/couleur)	id, produit_id	taille, couleur, stock_disponible
Produits	collections	Collections saisonnières	id	nom, saison, annee, statut
Produits	produits_collections	Liaison produits-collections	id, produit_id	collection_id, ordre
E-commerce	paniers	Paniers d'achat	id, client_id	session_id, statut, date_creation
E-commerce	articles_panier	Articles dans le panier	id, panier_id	variante_id, quantite, prix_total
E-commerce	commandes_ecommerce	Commandes en ligne	id, numero_commande	client_id, total_ttc, statut

 Catégorie	 Table	 Description	 Clés	 Champs Principaux
Clients	clients_textile	Base clients textile	id, email	nom, prenom, preferences_style
Livraison	livraisons_textile	Livraisons textile	id, commande_id	adresse, methode, frais
Retours	retours_textile	Gestion des retours	id, commande_id	motif, statut, remboursement
Marketing	promotions_textile	Promotions et offres	id	nom, reduction, date_debut, date_fin
Analytics	visites_produits	Suivi des visites	id, produit_id	client_id, date_visite, duree

Tables de Gestion des Produits Textiles

```
-- Table des produits textiles
CREATE TABLE produits_textile (
  id SERIAL PRIMARY KEY,
  reference VARCHAR(100) UNIQUE NOT NULL,
  nom VARCHAR(255) NOT NULL,
  description TEXT,
  marque VARCHAR(100),
  categorie VARCHAR(100),
  sous_categorie VARCHAR(100),
  matiere VARCHAR(100),
  entretien TEXT,
  prix_ht DECIMAL(10,2) NOT NULL,
  prix_ttc DECIMAL(10,2) NOT NULL,
  prix_promo DECIMAL(10,2),
  stock_total INTEGER DEFAULT 0,
  statut VARCHAR(20) DEFAULT 'actif',
  images JSONB,
  tags TEXT[],
  date_creation TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);

-- Table des variantes de produits
CREATE TABLE variantes_textile (
  id SERIAL PRIMARY KEY,
  produit_id INTEGER REFERENCES produits_textile(id),
  taille VARCHAR(20) NOT NULL,
  couleur VARCHAR(50) NOT NULL,
  stock_disponible INTEGER DEFAULT 0,
  prix_variante DECIMAL(10,2),
  code_sku VARCHAR(100) UNIQUE,
  poids_g DECIMAL(8,2),
  dimensions JSONB
);

-- Table des collections
CREATE TABLE collections (
  id SERIAL PRIMARY KEY,
  nom VARCHAR(255) NOT NULL,
  description TEXT,
  saison VARCHAR(50),
  annee INTEGER,
  date_lancement DATE,
```

```

date_fin DATE,
statut VARCHAR(20) DEFAULT 'active',
image_collection VARCHAR(255)
);

-- Table de liaison produits-collections
CREATE TABLE produits_collections (
  id SERIAL PRIMARY KEY,
  produit_id INTEGER REFERENCES produits_textile(id),
  collection_id INTEGER REFERENCES collections(id),
  ordre INTEGER DEFAULT 0
);

```

Tables E-commerce

```

-- Table des paniers
CREATE TABLE paniers (
  id SERIAL PRIMARY KEY,
  client_id INTEGER,
  session_id VARCHAR(255),
  date_creation TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  date_modification TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  statut VARCHAR(20) DEFAULT 'actif'
);

-- Table des articles de panier
CREATE TABLE articles_panier (
  id SERIAL PRIMARY KEY,
  panier_id INTEGER REFERENCES paniers(id),
  variante_id INTEGER REFERENCES variantes_textile(id),
  quantite INTEGER NOT NULL,
  prix_unitaire DECIMAL(10,2) NOT NULL,
  prix_total DECIMAL(10,2) NOT NULL
);




-- Table des commandes e-commerce
CREATE TABLE commandes_ecommerce (
  id SERIAL PRIMARY KEY,
  numero_commande VARCHAR(100) UNIQUE NOT NULL,
  client_id INTEGER,
  panier_id INTEGER REFERENCES paniers(id),
  date_commande TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  statut VARCHAR(50) DEFAULT 'en_attente',
  total_ht DECIMAL(10,2) NOT NULL,
  total_tva DECIMAL(10,2) NOT NULL,
  total_ttc DECIMAL(10,2) NOT NULL,
  adresse_facturation JSONB,
  adresse_livraison JSONB,
  methode_paiement VARCHAR(50),
  methode_livraison VARCHAR(50),
  frais_livraison DECIMAL(10,2) DEFAULT 0
);

```

YZ-LOGISTIQUE-ENTREPRISE-B_DB - Base Logistique



Tableau des Tables de la Base Logistique

 Catégorie	 Table	 Description	 Clés	 Champs Principaux
Entrepôts	entrepots	Gestion des entrepôts	id	nom, adresse, superficie_m2, capacite
Entrepôts	zones_entrepot	Zones d'entrepôt	id, entrepot_id	nom, type_zone, temperature, humidite
Entrepôts	emplacements_stockage	Emplacements de stockage	id, zone_id	code_emplacement, type, capacite
Stock	articles_stock	Articles en stock	id, reference_article	nom_article, categorie, stock_securite
Stock	mouvements_stock	Mouvements de stock	id, article_id	type_mouvement, quantite, emplacement
Stock	inventaires	Inventaires	id, entrepot_id	date_inventaire, statut, responsable_id
Stock	lignes_inventaire	Lignes d'inventaire	id, inventaire_id	article_id, quantite_theorique, quantite_reelle
Transport	vehicules	Flotte de véhicules	id, immatriculation	marque, modele, capacite_charge_kg
Transport	tournees_livraison	Tournées de livraison	id, vehicule_id	date_tournee, distance_km, statut
Transport	points_livraison	Points de livraison	id, tournee_id	ordre_passage, adresse, heure_prevue
Fournisseurs	fournisseurs	Gestion des fournisseurs	id	nom, adresse, telephone, email
Maintenance	maintenance_equipements	Maintenance des équipements	id, equipement_id	type_maintenance, date, cout
Analytics	metriques_logistique	KPIs logistiques	id, entrepot_id	taux_rotation, precision_inventaire

Tables de Gestion des Entrepôts

```

-- Table des entrepôts
CREATE TABLE entrepots (
  id SERIAL PRIMARY KEY,
  nom VARCHAR(255) NOT NULL,
  adresse TEXT NOT NULL,
  ville VARCHAR(100) NOT NULL,
  code_postal VARCHAR(10) NOT NULL,
  pays VARCHAR(100) DEFAULT 'France',
  superficie_m2 DECIMAL(10,2),
  capacite_stockage INTEGER,
  statut VARCHAR(20) DEFAULT 'actif',
  responsable_id INTEGER,
  coordonnees_gps POINT,
  date_ouverture DATE,
  horaires JSONB
);

-- Table des zones d'entrepôt
CREATE TABLE zones_entrepot (
  id SERIAL PRIMARY KEY,
  entrepot_id INTEGER REFERENCES entrepots(id),
  nom VARCHAR(100) NOT NULL,
  type_zone VARCHAR(50) NOT NULL CHECK (type_zone IN ('RECEPTION', 'STOCKAGE', 'PREPARATION',
'EXPEDITION', 'QUARANTAINE')),
  superficie_m2 DECIMAL(10,2),
  capacite_max INTEGER,
  temperature_celsius DECIMAL(5,2),
  humidite_pourcent DECIMAL(5,2),
  statut VARCHAR(20) DEFAULT 'active',
  plan_zone JSONB
);

-- Table des emplacements de stockage
CREATE TABLE emplacements_stockage (
  id SERIAL PRIMARY KEY,
  zone_id INTEGER REFERENCES zones_entrepot(id),
  code_emplacement VARCHAR(50) NOT NULL,
  type_emplacement VARCHAR(50) CHECK (type_emplacement IN ('PALETTE', 'CASIER', 'RAYON', 'SURFACE')),
  dimensions JSONB,
  capacite INTEGER,
  statut VARCHAR(20) DEFAULT 'disponible'
);

```

Tables de Gestion des Stocks

```

-- Table des articles en stock
CREATE TABLE articles_stock (
  id SERIAL PRIMARY KEY,
  reference_article VARCHAR(100) NOT NULL,
  nom_article VARCHAR(255) NOT NULL,
  categorie VARCHAR(100),
  unite VARCHAR(20) DEFAULT 'piece',
  stock_securite INTEGER DEFAULT 0,
  stock_alerte INTEGER DEFAULT 0,
  date_creation TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);

-- Table des mouvements de stock
CREATE TABLE mouvements_stock (
  id SERIAL PRIMARY KEY,
  article_id INTEGER REFERENCES articles_stock(id),
  emplacement_source_id INTEGER REFERENCES emplacements_stockage(id),
  emplacement_destination_id INTEGER REFERENCES emplacements_stockage(id),
  type_mouvement VARCHAR(50) NOT NULL CHECK (type_mouvement IN ('RECEPTION', 'TRANSFERT',

```

```

'EXPEDITION', 'INVENTAIRE', 'AJUSTEMENT')),
    quantite INTEGER NOT NULL,
    date_mouvement TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    operateur_id INTEGER,
    reference_document VARCHAR(100),
    commentaire TEXT,
    statut VARCHAR(20) DEFAULT 'confirme'
);

-- Table des inventaires
CREATE TABLE inventaires (
    id SERIAL PRIMARY KEY,
    entrepot_id INTEGER REFERENCES entrepots(id),
    zone_id INTEGER REFERENCES zones_entrepot(id),
    date_inventaire DATE NOT NULL,
    statut VARCHAR(20) DEFAULT 'en_cours',
    responsable_id INTEGER,
    commentaire TEXT
);

-- Table des lignes d'inventaire
CREATE TABLE lignes_inventaire (
    id SERIAL PRIMARY KEY,
    inventaire_id INTEGER REFERENCES inventaires(id),
    article_id INTEGER REFERENCES articles_stock(id),
    emplacement_id INTEGER REFERENCES emplacements_stockage(id),
    quantite_theorique INTEGER NOT NULL,
    quantite_reelle INTEGER,
    ecart INTEGER,
    commentaire TEXT
);

```

Tables de Transport et Livraison

```

-- Table des véhicules
CREATE TABLE vehicules (
    id SERIAL PRIMARY KEY,
    immatriculation VARCHAR(20) UNIQUE NOT NULL,
    marque VARCHAR(100),
    modele VARCHAR(100),
    type_vehicule VARCHAR(50) CHECK (type_vehicule IN ('CAMION', 'CAMIONNETTE', 'VOITURE')),
    capacite_charge_kg DECIMAL(8,2),
    chauffeur_id INTEGER,
    statut VARCHAR(20) DEFAULT 'disponible',
    date_maintenance DATE,
    kilometrage INTEGER
);

-- Table des tournées de livraison
CREATE TABLE tournées_livraison (
    id SERIAL PRIMARY KEY,
    vehicule_id INTEGER REFERENCES vehicules(id),
    chauffeur_id INTEGER,
    date_tournee DATE NOT NULL,
    heure_depart TIME,
    heure_retour TIME,
    statut VARCHAR(20) DEFAULT 'planifie',
    distance_km DECIMAL(8,2),
    carburant_consomme_l DECIMAL(6,2)
);






-- Table des points de livraison
CREATE TABLE points_livraison (
    id SERIAL PRIMARY KEY,
    tournee_id INTEGER REFERENCES tournées_livraison(id),
    ordre_passage INTEGER NOT NULL,

```

```
adresse TEXT NOT NULL,  
client VARCHAR(255),  
telephone VARCHAR(20),  
heure_prevue TIME,  
statut VARCHAR(20) DEFAULT 'en_attente',  
commentaire TEXT  
);
```

YZ-RESTAURANT-ENTREPRISE-C_DB - Base Restaurant

Tableau des Tables de la Base Restaurant

 Catégorie	 Table	 Description	 Clés	 Champs Principaux
Menus	menus	Catalogue des menus	id	nom, type_menu, prix, allergenes
Menus	ingredients	Gestion des ingrédients	id	nom, unite, prix_unitaire, stock_disponible
Menus	composition_menus	Composition des menus	id, menu_id	ingredient_id, quantite, ordre_preparation
Menus	categories_menus	Catégories de menus	id	nom, description, ordre_affichage
Commandes	commandes_restaurant	Commandes restaurant	id, numero_commande	client_id, type_commande, total_ttc
Commandes	lignes_commande_restaurant	Lignes de commande	id, commande_id	menu_id, quantite, prix_total
Livraison	livreurs	Gestion des livreurs	id	nom, prenom, telephone, zone_livraison
Livraison	livraisons	Livraisons à domicile	id, commande_id	livreur_id, date_livraison, distance_km
Clients	clients_restaurant	Base clients restaurant	id	nom, prenom, telephone, preferences
Horaires	horaires_ouverture	Horaires d'ouverture	id	jour_semaine, heure_ouverture,

 Catégorie	 Table	 Description	 Clés	 Champs Principaux
				heure_fermeture
Promotions	promotions_restaurant	Promotions et offres	id	nom, reduction, conditions, date_validite
Analytics	commandes_analytics	Analytics des commandes	id	heure_commande, type_menu_populaire, temps_preparation

Tables de Gestion des Menus

```

-- Table des menus
CREATE TABLE menus (
  id SERIAL PRIMARY KEY,
  nom VARCHAR(255) NOT NULL,
  description TEXT,
  type_menu VARCHAR(50) CHECK (type_menu IN ('ENTREE', 'PLAT', 'DESSERT', 'BOISSON', 'MENU_COMPLET')),
  prix DECIMAL(8,2) NOT NULL,
  disponible BOOLEAN DEFAULT TRUE,
  allergenes TEXT[],
  informations_nutritionnelles JSONB,
  image VARCHAR(255),
  ordre_affichage INTEGER DEFAULT 0
);

-- Table des ingrédients
CREATE TABLE ingredients (
  id SERIAL PRIMARY KEY,
  nom VARCHAR(255) NOT NULL,
  categorie VARCHAR(100),
  unite VARCHAR(20) DEFAULT 'gramme',
  prix_unitaire DECIMAL(8,4) NOT NULL,
  stock_disponible DECIMAL(10,2) DEFAULT 0,
  stock_securite DECIMAL(10,2) DEFAULT 0,
  fournisseur_id INTEGER,
  allergene BOOLEAN DEFAULT FALSE,
  bio BOOLEAN DEFAULT FALSE,
  date_peremption DATE
);

-- Table de composition des menus
CREATE TABLE composition_menus (
  id SERIAL PRIMARY KEY,
  menu_id INTEGER REFERENCES menus(id),
  ingredient_id INTEGER REFERENCES ingredients(id),
  quantite DECIMAL(8,2) NOT NULL,
  ordre_preparation INTEGER DEFAULT 0
);

-- Table des catégories de menus
CREATE TABLE categories_menus (
  id SERIAL PRIMARY KEY,
  nom VARCHAR(100) NOT NULL,
  description TEXT,
  image VARCHAR(255),
  ordre_affichage INTEGER DEFAULT 0
);

```


Tables de Commandes et Livraison

```
-- Table des commandes
CREATE TABLE commandes_restaurant (
  id SERIAL PRIMARY KEY,
  numero_commande VARCHAR(100) UNIQUE NOT NULL,
  client_id INTEGER,
  type_commande VARCHAR(20) CHECK (type_commande IN ('SUR_PLACE', 'A_EMPORTER', 'LIVRAISON')),
  date_commande TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  heure_souhaitee TIME,
  statut VARCHAR(50) DEFAULT 'en_attente',
  total_ht DECIMAL(8,2) NOT NULL,
  total_tva DECIMAL(8,2) NOT NULL,
  total_ttc DECIMAL(8,2) NOT NULL,
  adresse_livraison TEXT,
  telephone_livraison VARCHAR(20),
  notes TEXT
);

-- Table des lignes de commande
CREATE TABLE lignes_commande_restaurant (
  id SERIAL PRIMARY KEY,
  commande_id INTEGER REFERENCES commandes_restaurant(id),
  menu_id INTEGER REFERENCES menus(id),
  quantite INTEGER NOT NULL,
  prix_unitaire DECIMAL(8,2) NOT NULL,
  prix_total DECIMAL(8,2) NOT NULL,
  modifications TEXT,
  allergenes_ajoutes TEXT[]
);

-- Table des livreurs
CREATE TABLE livreurs (
  id SERIAL PRIMARY KEY,
  nom VARCHAR(100) NOT NULL,
  prenom VARCHAR(100) NOT NULL,
  telephone VARCHAR(20) NOT NULL,
  vehicule VARCHAR(100),
  zone_livraison TEXT[],
  statut VARCHAR(20) DEFAULT 'disponible',
  note_evaluation DECIMAL(3,2) DEFAULT 0.00
);






-- Table des livraisons
CREATE TABLE livraisons (
  id SERIAL PRIMARY KEY,
  commande_id INTEGER REFERENCES commandes_restaurant(id),
  livreur_id INTEGER REFERENCES livreurs(id),
  date_livraison DATE NOT NULL,
  heure_depart TIME,
  heure_arrivee TIME,
  statut VARCHAR(20) DEFAULT 'en_preparation',
  distance_km DECIMAL(6,2),
  frais_livraison DECIMAL(6,2) DEFAULT 0
);
```



YZ-PHARMACIE-ENTREPRISE-D_DB - Base Pharmacie



Tableau des Tables de la Base Pharmacie

 Catégorie	 Table	 Description	 Clés	 Champs Principaux
Médicaments	medicaments	Catalogue médicaments	id, code_cip	nom_commercial, principe_actif, prix_ttc
Médicaments	ordonnances	Gestion des ordonnances	id, numero_ordonnance	patient_id, medecin_id, date_prescription
Médicaments	prescriptions_medicaments	Prescriptions	id, ordonnance_id	medicament_id, posologie, duree_traitement
Patients	patients	Base patients	id, numero_patient	nom, prenom, allergies, antecedents
Médecins	medecins	Gestion des médecins	id, numero_rpps	nom, prenom, specialite, adresse_cabinet
Stock	stock_pharma	Stock pharmaceutique	id, medicament_id	quantite_disponible, date_peremption, lot
Commandes	commandes_pharma	Commandes en ligne	id	patient_id, medicaments, statut, livraison
Livraison	livraisons_pharma	Livraison express	id, commande_id	adresse, urgence, signature
Conseils	conseils_pharma	Conseils pharmaceutiques	id	question, reponse, pharmacien_id
Alertes	alertes_peremption	Alertes de péremption	id, medicament_id	date_alerte, niveau_urgence, action
Analytics	ventes_medicaments	Analytics des ventes	id	medicament_id, quantite_vendue, periode

Tables de Gestion des Médicaments

```
-- Table des médicaments
CREATE TABLE medicaments (
  id SERIAL PRIMARY KEY,
```

```

code_cip VARCHAR(100) UNIQUE NOT NULL,
nom_commercial VARCHAR(255) NOT NULL,
principe_actif VARCHAR(255),
forme_pharmaceutique VARCHAR(100),
dosage VARCHAR(100),
laboratoire VARCHAR(255),
prix_ht DECIMAL(8,2) NOT NULL,
prix_ttc DECIMAL(8,2) NOT NULL,
stock_disponible INTEGER DEFAULT 0,
stock_securite INTEGER DEFAULT 0,
prescription_obligatoire BOOLEAN DEFAULT FALSE,
remboursement_secu BOOLEAN DEFAULT FALSE,
taux_remboursement INTEGER DEFAULT 0,
date_peremption DATE,
statut VARCHAR(20) DEFAULT 'actif'
);

-- Table des ordonnances
CREATE TABLE ordonnances (
  id SERIAL PRIMARY KEY,
  numero_ordonnance VARCHAR(100) UNIQUE NOT NULL,
  patient_id INTEGER,
  medecin_id INTEGER,
  date_prescription DATE NOT NULL,
  date_validite DATE,
  statut VARCHAR(20) DEFAULT 'active',
  type_ordonnance VARCHAR(50) CHECK (type_ordonnance IN ('SIMPLE', 'RENOUVELLEABLE', 'SPECIALE')),
  commentaires TEXT
);






-- Table des prescriptions de médicaments
CREATE TABLE prescriptions_medicaments (
  id SERIAL PRIMARY KEY,
  ordonnance_id INTEGER REFERENCES ordonnances(id),
  medicament_id INTEGER REFERENCES medicaments(id),
  posologie TEXT NOT NULL,
  duree_traitement_jours INTEGER,
  quantite_prescrire INTEGER NOT NULL,
  renouvellements INTEGER DEFAULT 0,
  instructions_particulieres TEXT
);

-- Table des patients
CREATE TABLE patients (
  id SERIAL PRIMARY KEY,
  numero_patient VARCHAR(100) UNIQUE NOT NULL,
  nom VARCHAR(100) NOT NULL,
  prenom VARCHAR(100) NOT NULL,
  date_naissance DATE,
  sexe VARCHAR(1) CHECK (sexe IN ('M', 'F')),
  telephone VARCHAR(20),
  email VARCHAR(254),
  adresse TEXT,
  ville VARCHAR(100),
  code_postal VARCHAR(10),
  groupe_sanguin VARCHAR(10),
  allergies TEXT[],
  antecedents_medicaux TEXT,
  date_creation TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);

```



Tableau des Tables de la Base Électronique

 Catégorie	 Table	 Description	 Clés	 Champs Principaux
Produits	produits_electroniques	Catalogue tech	id, reference	nom, marque, modele, specifications
Produits	garanties	Gestion des garanties	id, produit_id	type_garantie, duree_mois, prix
Produits	services_reparation	Services de réparation	id	nom_service, duree_estimee, prix_ttc
Stock	stock_electronique	Stock des produits	id, produit_id	quantite_disponible, localisation
Commandes	commandes_electronique	Commandes tech	id	client_id, produits, total_ttc
Support	tickets_support	Support technique	id	client_id, probleme, statut, priorite
Diagnostics	diagnostics_produits	Diagnostics produits	id, produit_id	symptome, solution, cout_reparation
Formations	formations_tech	Formations produits	id	nom_formation, duree, niveau, prix
Analytics	performance_produits	Performance des produits	id, produit_id	taux_default, satisfaction_client

Tables de Gestion des Produits Tech

```
-- Table des produits électroniques
CREATE TABLE produits_electroniques (
  id SERIAL PRIMARY KEY,
  reference VARCHAR(100) UNIQUE NOT NULL,
  nom VARCHAR(255) NOT NULL,
  marque VARCHAR(100),
  modele VARCHAR(100),
  categorie VARCHAR(100),
  sous_categorie VARCHAR(100),
  description_technique TEXT,
  specifications JSONB,
  prix_ht DECIMAL(10,2) NOT NULL,
  prix_ttc DECIMAL(10,2) NOT NULL,
  stock_disponible INTEGER DEFAULT 0,
  garantie_mois INTEGER DEFAULT 24,
  statut VARCHAR(20) DEFAULT 'actif',
  images JSONB,
  date_creation TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);






-- Table des garanties
CREATE TABLE garanties (
  id SERIAL PRIMARY KEY,
  produit_id INTEGER REFERENCES produits_electroniques(id),
```

```
type_garantie VARCHAR(50) CHECK (type_garantie IN ('CONSTRUCTEUR', 'EXTENDUE', 'PREMIUM')),
duree_mois INTEGER NOT NULL,
prix DECIMAL(8,2),
description TEXT,
conditions TEXT
);

-- Table des services de réparation
CREATE TABLE services_reparation (
  id SERIAL PRIMARY KEY,
  nom_service VARCHAR(255) NOT NULL,
  description TEXT,
  duree_estimee_jours INTEGER,
  prix_ht DECIMAL(8,2) NOT NULL,
  prix_ttc DECIMAL(8,2) NOT NULL,
  garantie_mois INTEGER DEFAULT 3,
  statut VARCHAR(20) DEFAULT 'actif'
);
```

YZ-COSMETIQUE-ENTREPRISE-F_DB - Base Cosmétiques

Tableau des Tables de la Base Cosmétiques

 Catégorie	 Table	 Description	 Clés	 Champs Principaux
Produits	cosmetiques	Catalogue cosmétiques	id, reference	nom, marque, ligne_produit, ingredients
Produits	conseils_beaute	Conseils beauté	id	titre, contenu, type_peau, age_cible
Stock	stock_cosmetiques	Stock des produits	id, produit_id	quantite_disponible, date_peremption
Commandes	commandes_cosmetiques	Commandes beauté	id	client_id, produits, total_ttc
Livraison	livraisons_cosmetiques	Livraison fragile	id, commande_id	emballage_special, instructions
Fidélité	programme_fidelite	Programme de fidélité	id, client_id	points_cumules, niveau, avantages
Tests	tests_virtuels	Tests virtuels produits	id	type_test, questions, resultats
Analytics	preferences_beaute	Préférences clients	id, client_id	type_peau, marques_preferees






Tables de Gestion des Produits Beauté






```
-- Table des cosmétiques
CREATE TABLE cosmetiques (
  id SERIAL PRIMARY KEY,
  reference VARCHAR(100) UNIQUE NOT NULL,
  nom VARCHAR(255) NOT NULL,
  marque VARCHAR(100),
  ligne_produit VARCHAR(100),
  categorie VARCHAR(100),
  sous_categorie VARCHAR(100),
  type_peau VARCHAR(100)[],
  age_cible VARCHAR(50),
  description TEXT,
  ingredients TEXT[],
  volume_ml DECIMAL(8,2),
  poids_g DECIMAL(8,2),
  prix_ht DECIMAL(8,2) NOT NULL,
  prix_ttc DECIMAL(8,2) NOT NULL,
  stock_disponible INTEGER DEFAULT 0,
  bio BOOLEAN DEFAULT FALSE,
  vegan BOOLEAN DEFAULT FALSE,
  images JSONB,
  statut VARCHAR(20) DEFAULT 'actif'
);

-- Table des conseils beauté
CREATE TABLE conseils_beaute (
  id SERIAL PRIMARY KEY,
  titre VARCHAR(255) NOT NULL,
  contenu TEXT NOT NULL,
  categorie VARCHAR(100),
  type_peau VARCHAR(100)[],
  age_cible VARCHAR(50),
  produits_recommandes INTEGER[],
  image VARCHAR(255),
  date_creation TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  statut VARCHAR(20) DEFAULT 'publie'
);
```

YZ-AUTOMOBILE-ENTREPRISE-G_DB - Base Automobile

Tableau des Tables de la Base Automobile

 Catégorie	 Table	 Description	 Clés	 Champs Principaux
Véhicules	vehicules	Gestion des véhicules	id, immatriculation	marque, modele, annee, carburant
Véhicules	pieces_detachees	Pièces détachées	id, reference	nom, marque, categorie, compatible_vehicules
Services	services_garage	Services garage	id	nom_service, description, duree_estimee

 Catégorie	 Table	 Description	 Clés	 Champs Principaux
Commandes	commandes_auto	Commandes pièces	id	client_id, pieces, total_ttc
Livraison	livraisons_auto	Livraison express	id, commande_id	urgence, signature, instructions
Garage	rendez_vous_garage	Prise de RDV	id	client_id, vehicule_id, service_id, date_rdv
Diagnostics	diagnostics_vehicule	Diagnostics véhicule	id, vehicule_id	symptome, diagnostic, cout_reparation
Maintenance	maintenance_vehicules	Planning maintenance	id, vehicule_id	type_maintenance, date_prevue, kilometrage
Analytics	performance_services	Performance des services	id	service_id, satisfaction_client, temps_realisation

Tables de Gestion des Véhicules

```
-- Table des véhicules
CREATE TABLE vehicules (
  id SERIAL PRIMARY KEY,
  immatriculation VARCHAR(20) UNIQUE NOT NULL,
  marque VARCHAR(100) NOT NULL,
  modele VARCHAR(100) NOT NULL,
  annee INTEGER,
  couleur VARCHAR(50),
  carburant VARCHAR(50),
  puissance_ch INTEGER,
  kilometrage INTEGER,
  proprietaire_id INTEGER,
  date_acquisition DATE,
  statut VARCHAR(20) DEFAULT 'actif',
  informations_techniques JSONB
);






-- Table des pièces détachées
CREATE TABLE pieces_detachees (
  id SERIAL PRIMARY KEY,
  reference VARCHAR(100) UNIQUE NOT NULL,
  nom VARCHAR(255) NOT NULL,
  marque VARCHAR(100),
  categorie VARCHAR(100),
  compatible_vehicules INTEGER[],
  prix_ht DECIMAL(8,2) NOT NULL,
  prix_ttc DECIMAL(8,2) NOT NULL,
  stock_disponible INTEGER DEFAULT 0,
  garantie_mois INTEGER DEFAULT 12,
  statut VARCHAR(20) DEFAULT 'actif'
);

-- Table des services garage
CREATE TABLE services_garage (
```

```
id SERIAL PRIMARY KEY,  
nom_service VARCHAR(255) NOT NULL,  
description TEXT,  
duree_estimee_heures DECIMAL(4,2),  
prix_ht DECIMAL(8,2) NOT NULL,  
prix_ttc DECIMAL(8,2) NOT NULL,  
categorie VARCHAR(100),  
statut VARCHAR(20) DEFAULT 'actif'  
);
```

YZ-IMMOBILIER-ENTREPRISE-H_DB - Base Immobilier

Tableau des Tables de la Base Immobilier

 Catégorie	 Table	 Description	 Clés	 Champs Principaux
Biens	biens_immobiliers	Catalogue des biens	id, reference	type_bien, titre, superficie_m2, prix_vente
Biens	visites	Planification des visites	id, bien_id	client_id, agent_id, date_visite, duree
Biens	contrats	Gestion des contrats	id	bien_id, client_id, type_contrat, montant
Clients	clients_immobilier	Base clients	id	nom, prenom, telephone, budget, criteres
Agents	agents_immobilier	Gestion des agents	id	nom, prenom, specialite, zone_geographique
Maintenance	maintenance_biens	Maintenance des biens	id, bien_id	type_maintenance, date_prevue, cout_estime
Comptabilité	comptabilite_locative	Comptabilité locative	id, bien_id	loyer, charges, taxes, date_echeance
Syndic	gestion_syndicale	Gestion syndicale	id, bien_id	charges_syndic, assemblee, travaux
Analytics	performance_biens	Performance des biens	id, bien_id	temps_vendu, prix_m2, rendement

Tables de Gestion des Biens

```
-- Table des biens immobiliers  
CREATE TABLE biens_immobiliers (  
  id SERIAL PRIMARY KEY,  
  reference VARCHAR(100) UNIQUE NOT NULL,
```













```
type_bien VARCHAR(50) CHECK (type_bien IN ('APPARTEMENT', 'MAISON', 'TERRAIN', 'LOCAL_COMMERCIAL',
'BUREAUX')),
titre VARCHAR(255) NOT NULL,
description TEXT,
superficie_m2 DECIMAL(8,2),
nombre_pieces INTEGER,
nombre_chambres INTEGER,
nombre_salles_bain INTEGER,
etage INTEGER,
ascenseur BOOLEAN DEFAULT FALSE,
parking BOOLEAN DEFAULT FALSE,
adresse TEXT NOT NULL,
ville VARCHAR(100) NOT NULL,
code_postal VARCHAR(10) NOT NULL,
prix_vente DECIMAL(12,2),
prix_location DECIMAL(8,2),
charges_locatives DECIMAL(8,2),
statut VARCHAR(20) DEFAULT 'disponible',
proprietaire_id INTEGER,
images JSONB,
date_creation TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);

-- Table des visites
CREATE TABLE visites (
id SERIAL PRIMARY KEY,
bien_id INTEGER REFERENCES biens_immobiliers(id),
client_id INTEGER,
agent_id INTEGER,
date_visite TIMESTAMP NOT NULL,
duree_minutes INTEGER DEFAULT 60,
statut VARCHAR(20) DEFAULT 'planifie',
commentaires TEXT,
resultat VARCHAR(50)
);
```

YZ-EDUCATION-ENTREPRISE-I_DB - Base Éducation

Tableau des Tables de la Base Éducation

 Catégorie	 Table	 Description	 Clés	 Champs Principaux
Formations	formations	Catalogue des formations	id, reference	titre, description, duree_heures, niveau
Formations	etudiants	Gestion des étudiants	id, numero_etudiant	nom, prenom, email, niveau_etudes
Formations	formateurs	Gestion des formateurs	id	nom, prenom, specialites, experience_annees
Planning	planning	Planning des cours	id	formation_id, formateur_id,

 Catégorie	 Table	 Description	 Clés	 Champs Principaux
				date_debut, date_fin
Sessions	sessions_formation	Sessions de formation	id, formation_id	formateur_id, date_debut, date_fin, lieu
Évaluations	evaluations	Évaluations et notes	id, etudiant_id	formation_id, note, commentaires
Certifications	certifications	Gestion des certifications	id, etudiant_id	formation_id, date_obtention, validee
Finance	gestion_financiere	Gestion financière	id, formation_id	cout_formation, modalites_paiement
Analytics	performance_formations	Performance des formations	id, formation_id	taux_reussite, satisfaction_etudiants

Tables de Gestion des Formations

```

-- Table des formations
CREATE TABLE formations (
  id SERIAL PRIMARY KEY,
  reference VARCHAR(100) UNIQUE NOT NULL,
  titre VARCHAR(255) NOT NULL,
  description TEXT,
  duree_heures INTEGER NOT NULL,
  niveau VARCHAR(50),
  prix_ht DECIMAL(8,2) NOT NULL,
  prix_ttc DECIMAL(8,2) NOT NULL,
  capacite_max INTEGER,
  prerequis TEXT,
  objectifs TEXT[],
  programme TEXT,
  statut VARCHAR(20) DEFAULT 'active'
);

-- Table des étudiants
CREATE TABLE etudiants (
  id SERIAL PRIMARY KEY,
  numero_etudiant VARCHAR(100) UNIQUE NOT NULL,
  nom VARCHAR(100) NOT NULL,
  prenom VARCHAR(100) NOT NULL,
  date_naissance DATE,
  email VARCHAR(254) UNIQUE NOT NULL,
  telephone VARCHAR(20),
  adresse TEXT,
  ville VARCHAR(100),
  code_postal VARCHAR(10),
  niveau_etudes VARCHAR(100),
  date_inscription DATE DEFAULT CURRENT_DATE,
  statut VARCHAR(20) DEFAULT 'actif'
);

-- Table des formateurs
CREATE TABLE formateurs (

```

```

id SERIAL PRIMARY KEY,
nom VARCHAR(100) NOT NULL,
prenom VARCHAR(100) NOT NULL,
email VARCHAR(254) UNIQUE NOT NULL,
telephone VARCHAR(20),
specialites TEXT[],
experience_annees INTEGER,
diplomes TEXT[],
statut VARCHAR(20) DEFAULT 'actif'
);






-- Table des sessions de formation
CREATE TABLE sessions_formation (
id SERIAL PRIMARY KEY,
formation_id INTEGER REFERENCES formations(id),
formateur_id INTEGER REFERENCES formateurs(id),
date_debut DATE NOT NULL,
date_fin DATE NOT NULL,
heure_debut TIME,
heure_fin TIME,
lieu VARCHAR(255),
nombre_inscrits INTEGER DEFAULT 0,
statut VARCHAR(20) DEFAULT 'planifie'
);






```

YZ-SANTE-ENTREPRISE-J_DB - Base Santé



Tableau des Tables de la Base Santé

 Catégorie	 Table	 Description	 Clés	 Champs Principaux
Patients	patients	Base patients	id, numero_patient	nom, prenom, date_naissance, allergies
Patients	medecins	Gestion des médecins	id, numero_rpps	nom, prenom, specialite, adresse_cabinet
RDV	rendez_vous	Prise de rendez-vous	id, patient_id	medecin_id, date_rdv, heure_rdv, motif
Dossiers	dossiers_medicaux	Dossiers médicaux	id, patient_id	observations, diagnostic, traitement
Prescriptions	prescriptions	Gestion des prescriptions	id, patient_id	medecin_id, médicaments, posologie
Télémédecine	consultations_distance	Consultations à distance	id, patient_id	medecin_id, date_consultation, plateforme

 Catégorie	 Table	 Description	 Clés	 Champs Principaux
Facturation	facturation_medicale	Facturation médicale	id, patient_id	consultation_id, montant, remboursement
Analytics	statistiques_sante	Statistiques de santé	id	periode, nombre_consultations, pathologies

Tables de Gestion des Patients

```
-- Table des patients
CREATE TABLE patients (
  id SERIAL PRIMARY KEY,
  numero_patient VARCHAR(100) UNIQUE NOT NULL,
  nom VARCHAR(100) NOT NULL,
  prenom VARCHAR(100) NOT NULL,
  date_naissance DATE,
  sexe VARCHAR(1) CHECK (sexe IN ('M', 'F')),
  telephone VARCHAR(20),
  email VARCHAR(254),
  adresse TEXT,
  ville VARCHAR(100),
  code_postal VARCHAR(10),
  groupe_sanguin VARCHAR(10),
  allergies TEXT[],
  antecedents_medicaux TEXT,
  antecedents_familiaux TEXT,
  date_creation TIMESTAMPT DEFAULT CURRENT_TIMESTAMP
);

-- Table des medecins
CREATE TABLE medecins (
  id SERIAL PRIMARY KEY,
  numero_rpps VARCHAR(100) UNIQUE NOT NULL,
  nom VARCHAR(100) NOT NULL,
  prenom VARCHAR(100) NOT NULL,
  specialite VARCHAR(100) NOT NULL,
  sous_specialite VARCHAR(100),
  telephone VARCHAR(20),
  email VARCHAR(254),
  adresse_cabinet TEXT,
  ville VARCHAR(100),
  code_postal VARCHAR(10),
  statut VARCHAR(20) DEFAULT 'actif'
);

-- Table des rendez-vous
CREATE TABLE rendez_vous (
  id SERIAL PRIMARY KEY,
  patient_id INTEGER REFERENCES patients(id),
  medecin_id INTEGER REFERENCES medecins(id),
  date_rdv DATE NOT NULL,
  heure_rdv TIME NOT NULL,
  duree_minutes INTEGER DEFAULT 30,
  type_consultation VARCHAR(100),
  motif TEXT,
  statut VARCHAR(20) DEFAULT 'planifie',
  notes TEXT
);
```

```
-- Table des dossiers médicaux
CREATE TABLE dossiers_medicaux (
  id SERIAL PRIMARY KEY,
  patient_id INTEGER REFERENCES patients(id),
  date_creation TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  date_modification TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  observations TEXT,
  diagnostic TEXT,
  traitement TEXT,
  recommandations TEXT
);
```

Relations entre les Bases de Données

Architecture de Liaison

```
-- Dans YZ-PLATFORM_DB
CREATE TABLE tenant_databases (
  id SERIAL PRIMARY KEY,
  tenant_id INTEGER REFERENCES tenants(id),
  nom_base VARCHAR(100) NOT NULL,
  host VARCHAR(255) NOT NULL,
  port INTEGER DEFAULT 5432,
  nom_database VARCHAR(100) NOT NULL,
  utilisateur VARCHAR(100) NOT NULL,
  mot_de_passe_hash VARCHAR(255) NOT NULL,
  statut VARCHAR(20) DEFAULT 'actif',
  date_creation TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);

-- Table de synchronisation des données
CREATE TABLE data_sync_logs (
  id SERIAL PRIMARY KEY,
  tenant_id INTEGER REFERENCES tenants(id),
  table_source VARCHAR(100) NOT NULL,
  operation VARCHAR(20) CHECK (operation IN ('INSERT', 'UPDATE', 'DELETE')),
  nombre_lignes INTEGER NOT NULL,
  date_synchronisation TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  statut VARCHAR(20) DEFAULT 'succes',
  erreur_message TEXT
);
```

Métriques et Monitoring des Bases

Tables de Performance

```
-- Dans YZ-PLATFORM_DB
CREATE TABLE database_metrics (
  id SERIAL PRIMARY KEY,
  tenant_id INTEGER REFERENCES tenants(id),
  nom_base VARCHAR(100) NOT NULL,
  date_mesure TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  taille_gb DECIMAL(10,2),
```

```
nombre_tables INTEGER,  
nombre_lignes_total BIGINT,  
temps_reponse_moyen_ms INTEGER,  
connexions_actives INTEGER,  
requetes_par_seconde INTEGER  
);  
  
-- Table des alertes de base de données  
CREATE TABLE database_alerts (  
  id SERIAL PRIMARY KEY,  
  tenant_id INTEGER REFERENCES tenants(id),  
  type_alerte VARCHAR(50) CHECK (type_alerte IN ('ESPACE_DISQUE', 'PERFORMANCE', 'CONNEXIONS',  
'BACKUP')),  
  severite VARCHAR(20) CHECK (severite IN ('LOW', 'MEDIUM', 'HIGH', 'CRITICAL')),  
  message TEXT NOT NULL,  
  date_alerte TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
  statut VARCHAR(20) DEFAULT 'ouverte',  
  date_resolution TIMESTAMP  
);
```



Avantages de cette Architecture

1. Isolation des Données

- **Sécurité maximale** : Chaque projet a sa propre base de données
- **Conformité RGPD** : Données strictement séparées par entreprise
- **Audit facilité** : Traçabilité complète par tenant

2. Performance Optimisée

- **Requêtes rapides** : Pas de filtrage par tenant_id
- **Index optimisés** : Spécifiques à chaque métier
- **Scalabilité** : Chaque base peut être sur un serveur différent

3. Maintenance Simplifiée

- **Backups indépendants** : Par projet et par base
- **Mises à jour** : Sans impact sur les autres projets
- **Restauration** : Granulaire par base de données

4. Développement Flexible

- **Schémas adaptés** : Chaque métier a ses tables spécifiques
- **Évolutions indépendantes** : Modifications sans contraintes
- **Tests isolés** : Développement sans risque de conflit



Commandes de Gestion des Bases

Création d'une Nouvelle Base

```
# Créer une nouvelle base pour un tenant
createdb YZ-TEXTILE-ENTREPRISE-A_DB

# Créer l'utilisateur associé
createuser -P YZ_TEXTILE_USER

# Accorder les privilèges
psql -d YZ-TEXTILE-ENTREPRISE-A_DB -c "GRANT ALL PRIVILEGES ON DATABASE YZ-TEXTILE-ENTREPRISE-A_DB TO
YZ_TEXTILE_USER;"
```

Backup et Restauration

```
# Backup d'une base spécifique
pg_dump YZ-CMD-YOOZAK_DB > backup_yoozak_$(date +%Y%m%d).sql

# Restauration d'une base
psql YZ-CMD-YOOZAK_DB < backup_yoozak_20241201.sql

# Backup de toutes les bases
for db in YZ-PLATFORM_DB YZ-CMD-YOOZAK_DB YZ-TEXTILE-ENTREPRISE-A_DB; do
    pg_dump $db > backup_${db}_$(date +%Y%m%d).sql
done
```

Monitoring des Performances






```
-- Vérifier la taille des bases
SELECT
    datname AS base_donnees,
    pg_size_pretty(pg_database_size(datname)) AS taille
FROM pg_database
WHERE datname LIKE 'YZ-%'
ORDER BY pg_database_size(datname) DESC;

-- Vérifier les connexions actives
SELECT
    datname,
    count(*) AS connexions_actives
FROM pg_stat_activity
WHERE datname LIKE 'YZ-%'
GROUP BY datname;
```





Tableau Comparatif des Bases de Données

Comparaison des Fonctionnalités par Base

 Base	 Tables	 Sécurité	 Performance	 Spécialisation
YZ-PLATFORM_DB	15+	Multi-tenants	Haute	Authentification, Facturation
YZ-CMD-YOOZAK_DB	25+	Isolée	Optimisée	E-commerce, Logistique
YZ-TEXTILE-ENTREPRISE-A_DB	20+	Isolée	Optimisée	Mode, Collections, E-commerce
YZ-LOGISTIQUE-ENTREPRISE-B_DB	18+	Isolée	Optimisée	Entrepôts, Stock, Transport
YZ-RESTAURANT-ENTREPRISE-C_DB	15+	Isolée	Optimisée	Restauration, Livraison
YZ-PHARMACIE-ENTREPRISE-D_DB	12+	Isolée	Optimisée	Médical, Ordonnances
YZ-ELECTRONIQUE-ENTREPRISE-E_DB	10+	Isolée	Optimisée	High-Tech, Garanties
YZ-COSMETIQUE-ENTREPRISE-F_DB	8+	Isolée	Optimisée	Beauté, Conseils
YZ-AUTOMOBILE-ENTREPRISE-G_DB	12+	Isolée	Optimisée	Véhicules, Pièces, Garage
YZ-IMMOBILIER-ENTREPRISE-H_DB	10+	Isolée	Optimisée	Biens, Visites, Contrats
YZ-EDUCATION-ENTREPRISE-I_DB	12+	Isolée	Optimisée	Formations, Étudiants
YZ-SANTE-ENTREPRISE-J_DB	15+	Isolée	Optimisée	Patients, Médecins, RDV

Statistiques Globales

 Métrique	 Valeur
Total des Bases	12
Total des Tables	200+
Bases Sectorielles	10
Base Centrale	1
Base Existante	1
Secteurs d'Activité	10

Cette architecture de bases de données multi-tenants vous offre une **séparation complète des données** tout en maintenant une **gestion centralisée** via YZ-PLATFORM. Chaque projet sectoriel bénéficie de sa propre base PostgreSQL optimisée pour ses besoins métier spécifiques.