# Schéma relationnel simplifié et tables de données

**Application :** Suivi FTI EMGI
**Projet :** Suivi facturation / échéancier / paiement (EMGI)
**Date :** 2026-02-09
**Référence physique :** `app/sql/schema.sql`

---

## 1. Schéma relationnel simplifié

Représentation des tables et des relations (clés primaires et étrangères). Les cardinalités sont indiquées (1:N, 0:1).



**Relations :**

- **clients** 1 —— N **factures** (un client a plusieurs factures ; une facture appartient à un client).
- **factures** 1 —— N **paiements** (une facture a plusieurs paiements ; un paiement est rattaché à une facture).
- **factures** 1 —— 0..1 **echeanciers** (une facture a au plus un écheancier ; un écheancier est rattaché à une facture).

**Contraintes :**

- Suppression en cascade : suppression d'un client → suppression de ses factures, puis de leurs paiements et écheanciers. Suppression d'une facture → suppression de ses paiements et de son écheancier.

---

## 2. Tables – Détail des champs

Format : pour chaque table, description courte puis pour chaque colonne : **Nom**, **Type**, **Description**, **Clé**.

---

### Table `clients`

Table des clients (sociétés ou entités facturées). Une facture est toujours liée à un client.

| Nom | Type | Description | Clé |
|---|---|---|---|
| id | int UNSIGNED | Identifiant unique auto incrémenté | Primaire |
| nom | varchar(255) | Nom du client (raison sociale ou libellé) | |
| created_at | datetime | Date et heure de création de l'enregistrement | |

---

### Table `factures`

Table des factures pour le suivi. La facturation (document, PDF) est gérée dans Pennylane ; ici on enregistre les données de suivi (n° facture, montant, type produit, cycle, quantités, etc.).

| Nom | Type | Description | Clé |
|---|---|---|---|
| id | int UNSIGNED | Identifiant unique auto incrémenté | Primaire |
| client_id | int UNSIGNED | Référence du client concerné | Étrangère (→ clients.id), NOT NULL |
| n_facture | varchar(64) | Numéro de facture (ex. FA-001) | NOT NULL |
| n_vente | varchar(64) | Numéro de vente ou référence vente (optionnel) | |
| date_facture | date | Date de la facture | NOT NULL |
| type_produit | varchar(32) | Type de produit : alevins, larves (ou NULL) | |
| cycle | varchar(16) | Cycle de référence (ex. 24-25, 25-26) | |
| quantites | decimal(12,2) | Quantité facturée | |
| qte_livree | decimal(12,2) | Quantité déjà livrée | |
| montant_ht | decimal(12,2) | Montant HT de la facture (€) | NOT NULL |
| created_at | datetime | Date et heure de création de l'enregistrement dans l'appli | |

**Index :** `client_id`, `type_produit`, `cycle`.

---

### Table `paiements`

Table des paiements enregistrés pour une facture. L'encours d'une facture est calculé comme (montant_ht − somme des montants des paiements de cette facture).

| Nom | Type | Description | Clé |
|---|---|---|---|
| id | int UNSIGNED | Identifiant unique auto incrémenté | Primaire |
| facture_id | int UNSIGNED | Référence de la facture concernée | Étrangère (→ factures.id), NOT NULL |
| montant | decimal(12,2) | Montant du paiement (€) | NOT NULL |
| date_paiement | date | Date du paiement | NOT NULL |
| created_at | datetime | Date et heure de création de l'enregistrement | |

**Index :** `facture_id`, `date_paiement` (idx_paiements_date).

**Contraintes de validation :** `CHECK (montant > 0)`.

---

### Table `echeanciers`

Table des échéanciers de paiement associés à une facture. Une facture peut avoir au plus un échéancier. Les dates d'échéance sont dérivées à l'affichage : date_debut_echeancier + 0 mois, +1 mois, +2 mois, etc. (échéances mensuelles). Le champ `montant_par_echeance` est calculé à la création (encours / nb_echeances) et stocké.

| Nom | Type | Description | Clé |
|---|---|---|---|
| id | int UNSIGNED | Identifiant unique auto incrémenté | Primaire |
| facture_id | int UNSIGNED | Référence de la facture concernée | Étrangère (→ factures.id), NOT NULL, UNIQUE |
| nb_echeances | int UNSIGNED | Nombre d'échéances prévues | NOT NULL |
| date_debut_echeancier | date | Date de début de l'échéancier (1ère échéance) | NOT NULL |
| montant_par_echeance | decimal(12,2) | Montant par échéance (calculé à la création : encours / nb_echeances) | NOT NULL |
| created_at | datetime | Date et heure de création de l'enregistrement | |

**Index :** `facture_id` (UNIQUE pour garantir un échéancier par facture).

**Contraintes de validation :** `CHECK (nb_echeances > 0)`, `CHECK (montant_par_echeance > 0)`.

---

## 3. Résumé des clés et contraintes

| Table | Clé primaire | Clés étrangères | Contrainte cascade |
|---|---|---|---|
| clients | id | — | — |
| factures | id | client_id → clients | ON DELETE CASCADE (client supprimé) |
| paiements | id | facture_id → factures | ON DELETE CASCADE (facture supprimée) |
| echeanciers | id | facture_id → factures | ON DELETE CASCADE (facture supprimée) |

### 3.1 Contraintes de validation (CHECK)

| Table | Contrainte | Description |
|---|---|---|
| paiements | chk_paiements_montant | montant > 0 |
| echeanciers | chk_echeanciers_nb_echeances | nb_echeances > 0 |
| echeanciers | chk_echeanciers_montant | montant_par_echeance > 0 |

### 3.2 Index (référence app/sql/schema.sql)

| Table | Index | Colonne(s) |
|---|---|---|
| factures | idx_factures_client_id | client_id |
| factures | idx_factures_type_produit | type_produit |
| factures | idx_factures_cycle | cycle |
| paiements | idx_paiements_facture_id | facture_id |
| paiements | idx_paiements_date | date_paiement |
| echeanciers | uk_echeanciers_facture_id (UNIQUE) | facture_id |
| echeanciers | idx_echeanciers_facture_id | facture_id |

**Note :** Les statuts « payé » / « en attente » ne sont pas stockés ; ils sont calculés (encours facture = montant_ht − somme des paiements ; si encours ≤ 0 alors payé, sinon en attente).