

DOCUMENTATION TECHNIQUE BASE DE DONNÉES

Innov'Events

ECF — Concepteur Développeur d'Applications
Février 2026

1. Présentation générale

Le projet Innov'Events est une plateforme de gestion événementielle développée pour Chloé Dubois, organisatrice d'événements professionnels. L'architecture de données repose sur deux systèmes complémentaires :

MySQL 8.0 — Base relationnelle principale contenant les 12 tables métier (utilisateurs, clients, prospects, événements, devis, prestations, tâches, notes, avis, types d'événements, thèmes, paramètres).

MongoDB — Base documentaire complémentaire dédiée à la journalisation des actions sensibles (audit trail) sous forme de documents JSON semi-structurés.

Cette architecture hybride a été choisie car les données métier nécessitent des relations fortes (clés étrangères, intégrité référentielle) tandis que les logs d'audit, en écriture intensive et sans jointures, sont mieux adaptés au modèle documentaire.

2. Modèle Conceptuel de Données (MCD)

Le MCD ci-dessous décrit les 12 entités du système et leurs relations. Il est fourni au format Mermaid et peut être visualisé gratuitement sur :

<https://mermaid.ink> (rendu image direct) ou <https://kroki.io> (alternative open-source)

2.1 Entités et cardinalités

Entité	Clé primaire	Description	Relations principales
users	id (INT AI)	Comptes utilisateurs (admin, employee, client)	1:1 → clients
clients	id (INT AI)	Profils commerciaux rattachés à un user	1:N → events
prospects	id (INT AI)	Demandes entrantes (formulaire public)	Indépendant
events	id (INT AI)	Événements organisés	N:1 clients, 1:1 quotes
quotes	id (INT AI)	Devis associés aux événements	1:N → services
services	id (INT AI)	Prestations d'un devis	N:1 quotes
tasks	id (INT AI)	Tâches d'un événement	N:1 events, N:1 users
notes	id (INT AI)	Notes (globales ou liées à un événement)	N:1 events, N:1 users
reviews	id (INT AI)	Avis clients modérés	N:1 events, N:1 clients
event_types	id (INT AI)	Types d'événements (référentiel)	Référentiel
themes	id (INT AI)	Thèmes décoratifs (référentiel)	Référentiel
app_settings	id (INT AI)	Paramètres applicatifs	Référentiel

2.2 Diagramme MCD (Mermaid)

Copiez le code ci-dessous dans <https://mermaid.ink> ou <https://kroki.io> pour obtenir le diagramme :

```

erDiagram
    users ||--o| clients : "possède un profil"
    clients ||--o{ events : "organise"
    events ||--o| quotes : "a un devis"
    quotes ||--o{ services : "contient"
    events ||--o{ tasks : "comprend"
    users ||--o{ tasks : "est assigné"
    events ||--o{ notes : "liée à"
    users ||--o{ notes : "écrit par"
    events ||--o{ reviews : "évalué par"
    clients ||--o{ reviews : "laisse un avis"
    users ||--o{ reviews : "modéré par"

```

3. Dictionnaire de données

Table : users

Colonne	Type	NULL	Clé	Défaut	Description
id	INT	NON	PK	AUTO_IN_C	Identifiant unique
last_name	VARCHAR(100)	OUI		NULL	Nom de famille
first_name	VARCHAR(100)	OUI		NULL	Prénom
username	VARCHAR(100)	OUI	UQ	NULL	Nom d'utilisateur (unique)
email	VARCHAR(150)	NON	UQ		Adresse e-mail (unique)
password	VARCHAR(255)	NON			Mot de passe hashé (bcrypt)
role	ENUM	OUI		'client'	Rôle : admin, employee, client
is_active	TINYINT(1)	OUI		1	Compte actif (0/1)
email_verified	TINYINT(1)	OUI		0	E-mail vérifié (0/1)
email_verification_token	VARCHAR(255)	OUI		NULL	Token de vérification e-mail
password_reset_token	VARCHAR(255)	OUI		NULL	Token de réinitialisation MDP
password_reset_expires	DATETIME	OUI		NULL	Expiration du token reset
must_change_password	TINYINT(1)	OUI		0	Forcer le changement MDP
created_at	TIMESTAMP	OUI		CURRENT_TS	Date de création
updated_at	TIMESTAMP	OUI		CURRENT_TS	Date de dernière modification

Table : clients

Colonne	Type	NULL	Clé	Défaut	Description

id	INT	NON	PK	AUTO_IN_C	Identifiant unique
user_id	INT	OUI	FK_UQ	NULL	Référence vers users.id (1:1)
company_name	VARCHAR(100)	OUI		NULL	Nom de l'entreprise
phone	VARCHAR(20)	OUI		NULL	Téléphone
address	VARCHAR(255)	OUI		NULL	Adresse postale
created_at	TIMESTAMP	OUI		CURRENT_TS	Date de création

Table : prospects

Colonne	Type	NULL	Clé	Défaut	Description
id	INT	NON	PK	AUTO_IN_C	Identifiant unique
company_name	VARCHAR(150)	OUI		NULL	Nom de l'entreprise
last_name	VARCHAR(100)	OUI		NULL	Nom du contact
first_name	VARCHAR(100)	OUI		NULL	Prénom du contact
email	VARCHAR(150)	OUI		NULL	E-mail du contact
phone	VARCHAR(20)	OUI		NULL	Téléphone
location	VARCHAR(255)	OUI		NULL	Lieu souhaité
event_type	VARCHAR(100)	OUI		NULL	Type d'événement souhaité
planned_date	DATE	OUI		NULL	Date prévue
estimated_participants	INT	OUI		NULL	Nombre estimé de participants
needs_description	TEXT	OUI		NULL	Description du besoin
image_path	VARCHAR(255)	OUI		NULL	Image jointe au formulaire
status	ENUM	OUI		'to_contact'	Statut : to_contact, qualification, failed, converted
created_at	TIMESTAMP	OUI		CURRENT_TS	Date de création

Table : events

Colonne	Type	NULL	Clé	Défaut	Description
id	INT	NON	PK	AUTO_IN_C	Identifiant unique
client_id	INT	OUI	FK	NULL	Référence vers clients.id
name	VARCHAR(255)	NON			Nom de l'événement
description	TEXT	OUI		NULL	Description
start_date	DATETIME	NON			Date/heure de début
end_date	DATETIME	NON			Date/heure de fin
location	VARCHAR(255)	OUI		NULL	Lieu

attendees_count	INT	OUI		NULL	Nombre de participants
budget	DECIMAL(10,2)	OUI		NULL	Budget total TTC
image_path	VARCHAR(255)	OUI		NULL	Image de l'événement
event_type	VARCHAR(100)	OUI		NULL	Type d'événement
theme	VARCHAR(100)	OUI		NULL	Thème décoratif
status	ENUM	OUI		'draft'	Statut : draft, client_review, accepted, in_progress, completed, cancelled
is_visible	TINYINT(1)	OUI		0	Visible dans le portfolio
created_at	TIMESTAMP	OUI		CURRENT_TS	Date de création
updated_at	TIMESTAMP	OUI		CURRENT_TS	Date de modification

Table : quotes

Colonne	Type	NULL	Clé	Défaut	Description
id	INT	NON	PK	AUTO_IN_C	Identifiant unique
event_id	INT	OUI	FK	NULL	Référence vers events.id
total_ht	DECIMAL(10,2)	OUI		NULL	Total hors taxes
tax_rate	DECIMAL(5,2)	OUI		20.00	Taux de TVA
total_ttc	DECIMAL(10,2)	OUI		NULL	Total TTC
issue_date	DATE	OUI		NULL	Date d'émission
status	ENUM	OUI		'pending'	Statut : pending, modification, accepted, refused
modification_reason	TEXT	OUI		NULL	Motif de demande de modification
counter_proposal	TEXT	OUI		NULL	Contre-proposition du client
counter_proposed_at	DATETIME	OUI		NULL	Date de la contre-proposition
created_at	TIMESTAMP	OUI		CURRENT_TS	Date de création
updated_at	TIMESTAMP	OUI		CURRENT_TS	Date de modification

Table : services

Colonne	Type	NULL	Clé	Défaut	Description
id	INT	NON	PK	AUTO_IN_C	Identifiant unique
quote_id	INT	OUI	FK	NULL	Référence vers quotes.id
label	VARCHAR(255)	NON			Intitulé de la prestation
description	TEXT	OUI		NULL	Description détaillée

unit_price_ht	DECIMAL(10,2)	OUI		NULL	Prix unitaire HT
created_at	TIMESTAMP	OUI		CURRENT_TS	Date de création

Table : tasks

Colonne	Type	NULL	Clé	Défaut	Description
id	INT	NON	PK	AUTO_IN_C	Identifiant unique
event_id	INT	NON	FK		Référence vers events.id
assigned_to	INT	OUI	FK	NULL	Utilisateur assigné (users.id)
title	VARCHAR(255)	NON			Titre de la tâche
description	TEXT	OUI		NULL	Description
status	ENUM	OUI		'todo'	Statut : todo, in_progress, done
due_date	DATE	OUI		NULL	Date d'échéance
created_at	TIMESTAMP	OUI		CURRENT_TS	Date de création
updated_at	TIMESTAMP	OUI		CURRENT_TS	Date de modification

Table : notes

Colonne	Type	NULL	Clé	Défaut	Description
id	INT	NON	PK	AUTO_IN_C	Identifiant unique
event_id	INT	OUI	FK	NULL	Événement lié (NULL = note globale)
author_id	INT	NON	FK		Auteur (users.id)
content	TEXT	NON			Contenu de la note
is_global	TINYINT(1)	OUI		0	Note globale (non liée)
created_at	TIMESTAMP	OUI		CURRENT_TS	Date de création

Table : reviews

Colonne	Type	NULL	Clé	Défaut	Description
id	INT	NON	PK	AUTO_IN_C	Identifiant unique
event_id	INT	NON	FK		Événement évalué
client_id	INT	NON	FK		Client auteur de l'avis
rating	TINYINT	NON			Note (1 à 5)
comment	TEXT	OUI		NULL	Commentaire
status	ENUM	OUI		'pending'	Statut : pending, approved, rejected

reviewed_by	INT	OUI	FK	NULL	Modérateur (users.id)
created_at	TIMESTAMP	OUI		CURRENT_TS	Date de création

Table : event_types

Colonne	Type	NULL	Clé	Défaut	Description
id	INT	NON	PK	AUTO_IN_C	Identifiant unique
name	VARCHAR(100)	NON			Nom du type
description	TEXT	OUI		NULL	Description

Table : themes

Colonne	Type	NULL	Clé	Défaut	Description
id	INT	NON	PK	AUTO_IN_C	Identifiant unique
name	VARCHAR(100)	NON			Nom du thème

Table : app_settings

Colonne	Type	NULL	Clé	Défaut	Description
id	INT	NON	PK	AUTO_IN_C	Identifiant unique
setting_key	VARCHAR(120)	NON	UQ		Clé du paramètre
setting_value	TEXT	NON			Valeur du paramètre
updated_at	DATETIME	NON		CURRENT_TS	Date de modification

4. Règles de nommage

Élément	Convention	Exemple
Tables	snake_case, pluriel, anglais	users, event_types, app_settings
Colonnes	snake_case, anglais	first_name, created_at, is_active
Clés primaires	id (INT AUTO_INCREMENT)	id
Clés étrangères	table_singulier_id	client_id, event_id, user_id
Contraintes FK	fk_table_reference	fk_event_client, fk_note_author
Booléens	Préfixe is_ ou must_	is_active, is_visible, must_change_password
Timestamps	created_at, updated_at	Présents dans toutes les tables
ENUM	snake_case, anglais	to_contact, in_progress, client_review

5. Script DDL — Création des tables

Le script ci-dessous est rédigé manuellement et respecte l'ordre de création imposé par les dépendances de clés étrangères. Il est disponible dans le fichier `create_tables.sql`.

```
-- =====
-- Script DDL Innov'Events - Rédigé manuellement
-- =====
CREATE DATABASE IF NOT EXISTS innovevents
    CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
USE innovevents;

-- 1. Tables de référence (aucune dépendance)
CREATE TABLE event_types (
    id INT NOT NULL AUTO_INCREMENT,
    name VARCHAR(100) NOT NULL,
    description TEXT,
    PRIMARY KEY (id)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

CREATE TABLE themes (
    id INT NOT NULL AUTO_INCREMENT,
    name VARCHAR(100) NOT NULL,
    PRIMARY KEY (id)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

CREATE TABLE app_settings (
    id INT NOT NULL AUTO_INCREMENT,
    setting_key VARCHAR(120) NOT NULL,
    setting_value TEXT NOT NULL,
    updated_at DATETIME NOT NULL DEFAULT CURRENT_TIMESTAMP
        ON UPDATE CURRENT_TIMESTAMP,
    PRIMARY KEY (id),
    UNIQUE KEY (setting_key)
) ENGINE=InnoDB CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;

-- 2. Authentification
CREATE TABLE users (
    id INT NOT NULL AUTO_INCREMENT,
    last_name VARCHAR(100), first_name VARCHAR(100),
    email VARCHAR(100) NOT NULL,
    password VARCHAR(255) NOT NULL,
    created_at DATETIME NOT NULL DEFAULT CURRENT_TIMESTAMP
        ON UPDATE CURRENT_TIMESTAMP,
    PRIMARY KEY (id),
    UNIQUE KEY (email)
) ENGINE=InnoDB CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

```

username VARCHAR(100), email VARCHAR(150) NOT NULL,
password VARCHAR(255) NOT NULL,
role ENUM('admin','employee','client') DEFAULT 'client',
is_active TINYINT(1) DEFAULT 1,
email_verified TINYINT(1) DEFAULT 0,
email_verification_token VARCHAR(255),
password_reset_token VARCHAR(255),
password_reset_expires DATETIME,
must_change_password TINYINT(1) DEFAULT 0,
created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
    ON UPDATE CURRENT_TIMESTAMP,
PRIMARY KEY (id), UNIQUE KEY (email), UNIQUE KEY (username)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

-- 3-12 : clients, prospects, events, quotes, services,
-- tasks, notes, reviews (voir fichier complet)

```

6. Script DML — Jeu d'essai

Le jeu d'essai ci-dessous reflète les données réelles insérées dans l'application lors des tests fonctionnels. Toutes les données sont cohérentes avec le scénario métier d'Innov'Events.

6.1 Utilisateurs

Mot de passe en clair pour tous les comptes de test : Test1234!

ID	Nom	Prénom	Username	E-mail	Rôle
1	Dubois	Chloé	chloe_admin	chloe@innovevents.com	admin
2	Dupont	Alexandre	alexandredupont	alexandre@innovevents.com	employee
3	Renard	Camille	—	ateliernova@nova.fr	client
4	Durand	Sophie	—	sophiedurand@durand.fr	client

6.2 Clients

ID	User ID	Entreprise	Téléphone	Adresse
1	3	Atelier Nova	06 58 20 11 90	Toulouse
2	4	Maison Durand	06 42 19 77 05	12 rue des Chartrons, 33000 Bordeaux

6.3 Prospects

ID	Entreprise	Contact	Lieu	Type	Statut
1	Orbis Conseil	Nina Morel	Lille	séminaire	to_contact
2	Atelier Nova	Camille Renard	Toulouse	conférence	converted

6.4 Événements

ID	Client	Nom	Dates	Statut
1	Atelier Nova	Atelier Nova	2026-01-01 / 2026-01-02	en cours

1	Atelier Nova	Conférence — Atelier Nova	26/03/2026	accepted
2	Atelier Nova	Team Building — Atelier Nova	12/04/2026	draft
3	Maison Durand	Soirée d'entreprise — Maison Durand	05/03/2026	in_progress
4	Maison Durand	Séminaire — Bilan 2025	15/01/2026	completed

6.5 Devis et prestations

ID	Événement	Total HT	Total TTC	Statut
1	Conférence Atelier Nova (ID:1)	7 900,00 €	9 480,00 €	pending
2	Soirée Maison Durand (ID:3)	7 600,00 €	9 120,00 €	accepted

ID	Devis	Prestation	Prix HT
1	1	Location salle + technique	2 800,00 €
2	1	Traiteur (déjeuner + pauses)	4 200,00 €
3	1	Captation vidéo	900,00 €
4	2	DJ & animation	1 500,00 €
5	2	Décoration & éclairage	2 300,00 €
6	2	Cocktail dînatoire	3 800,00 €

6.6 Tâches, notes et avis

ID	Événement	Assigné	Titre	Statut
1	1	Alexandre (2)	Valider prestataire vidéo	todo
2	1	Alexandre (2)	Repérage salle + test projection	in_progress
3	3	Alexandre (2)	Confirmer DJ + matériel	done

ID	Événement	Auteur	Contenu	Globale
1	—	Chloé (1)	Démo soutenance : montrer devis PDF + envoi mail + espace client.	Oui
2	1	Chloé (1)	Prévoir micro HF + répétition intervenants 30 min avant.	Non

ID	Événement	Client	Note	Statut	Modérateur
1	4	Maison Durand (2)	5/5	approved	Alexandre (2)

7. Contraintes d'intégrité référentielle

Chaque clé étrangère dispose d'une stratégie de suppression adaptée au contexte métier :

7.1 Stratégies de suppression

CASCADE — Les données enfants n'ont pas de sens sans le parent. Suppression automatique en cascade.

SET NULL — Les données enfants conservent une valeur indépendante du parent. La FK passe à NULL.

7.2 Tableau des contraintes FK

Contrainte	Colonne	Référence	Action DELETE	Justification
fk_client_user	clients.user_id	users.id	SET NULL	Le profil client reste utile
fk_event_client	events.client_id	clients.id	CASCADE	Événement lié au client
fk_quote_event	quotes.event_id	events.id	CASCADE	Devis lié à l'événement
fk_service_quote	services.quote_id	quotes.id	CASCADE	Prestation liée au devis
fk_task_event	tasks.event_id	events.id	CASCADE	Tâche liée à l'événement
fk_task_user	tasks.assigned_to	users.id	SET NULL	Tâche reste, user détaché
fk_note_event	notes.event_id	events.id	SET NULL	Note globale possible
fk_note_author	notes.author_id	users.id	CASCADE	Note sans auteur inutile
fk_review_event	reviews.event_id	events.id	CASCADE	Avis lié à l'événement
fk_review_client	reviews.client_id	clients.id	CASCADE	Avis lié au client
fk_review_reviewer	reviews.reviewed_by	users.id	SET NULL	Modérateur détachable

8. Sécurité et confidentialité

8.1 Hashage des mots de passe

Tous les mots de passe sont hashés avec bcrypt via `password_hash()` de PHP (coût par défaut 10-12). Aucun mot de passe en clair n'est stocké en base. La vérification se fait via `password_verify()`.

8.2 Tokens de sécurité

Les tokens de vérification d'e-mail et de réinitialisation de mot de passe sont générés via `bin2hex(random_bytes(32))` produisant des chaînes SHA-256 de 64 caractères. Chaque token a une durée de validité limitée (`password_reset_expires`).

8.3 Protection contre les injections SQL

L'ensemble de l'API utilise PDO avec des requêtes préparées (`prepare / bindParam`). Aucune concaténation directe de paramètres dans les requêtes SQL.

8.4 Protection XSS

Côté backend, `htmlspecialchars()` est appliqué sur les sorties. Côté frontend, Angular effectue un auto-échappement natif de toutes les interpolations.

8.5 Contrôle d'accès (RBAC)

Trois rôles sont définis : admin (accès complet), employee (gestion opérationnelle) et client (accès limité à ses données). L'autorisation est vérifiée à chaque endpoint via les tokens JWT.

8.6 Conformité RGPD

Minimisation des données collectées, durées de conservation définies (3 ans comptes inactifs, 12 mois logs, 5 ans données comptables), droit d'accès et de suppression implémentés via l'API. La collecte d'adresses IP dans les logs d'audit est justifiée par l'intérêt légitime (sécurité) avec une durée de rétention de 12 mois conformément aux recommandations CNIL.

9. Stratégie de sauvegarde et restauration

9.1 Sauvegarde MySQL

Commande de sauvegarde (environnement Docker) :

```
docker exec innov-db mysqldump -u root -proot \
--single-transaction --routines --triggers \
innovevents > backup_innovevents_$(date +%Y%m%d).sql
```

9.2 Restauration MySQL

```
docker exec -i innov-db mysql -u root -proot \
innovevents < backup_innovevents_20260218.sql
```

9.3 Politique de sauvegarde recommandée

Fréquence	Rétention	Déclencheur
Quotidienne	7 jours	Cron automatique à 02h00
Hebdomadaire	4 semaines	Dimanche 03h00
Mensuelle	12 mois	1er du mois
Pré-déploiement	Conservation indéfinie	Avant chaque mise en production

9.4 Sauvegarde MongoDB

```
docker exec innov-mongo mongodump \
--db=innovevents_logs \
--out=/backup/mongo_$(date +%Y%m%d)
```

10. Base complémentaire MongoDB

10.1 Justification

MongoDB est utilisé pour la journalisation des actions sensibles (audit trail). Ce choix est motivé par la nature semi-structurée des logs (champs variables selon l'action), le volume d'écriture élevé, l'absence de jointures nécessaires, et la flexibilité du schéma documentaire.

10.2 Structure d'un document d'audit

```
{
  "_id": ObjectId("..."),
  "action": "LOGIN_SUCCESS",
  "user_id": 1,
  "user_email": "chloe@innovevents.com",
  "user_role": "admin",
  "ip_address": "192.168.1.100",
  "user_agent": "Mozilla/5.0...",
  "details": {
    "method": "POST",
    "endpoint": "/api/auth/login.php"
  },
  "timestamp": ISODate("2026-02-18T14:30:00Z")
}
```

10.3 Actions journalisées

Action	Déclencheur	Données enregistrées
LOGIN_SUCCESS	Connexion réussie	user_id, email, rôle, IP, User-Agent
LOGIN_FAILURE	Échec de connexion	email tenté, IP, User-Agent
LOGOUT	Déconnexion	user_id, IP
PASSWORD_CHANGE	Changement de mot de passe	user_id, méthode (reset/volontaire)
USER_CREATE	Création de compte	user_id créé, rôle, créateur
USER_UPDATE	Modification de profil	user_id, champs modifiés
DATA_EXPORT	Export de données (RGPD)	user_id, type d'export

10.4 Rétention

Les logs d'audit sont conservés 12 mois conformément aux recommandations CNIL. Un TTL index MongoDB peut être configuré pour la suppression automatique :

```
db.audit_logs.createIndex(
  { "timestamp": 1 },
  { expireAfterSeconds: 31536000 } // 365 jours
);
```