



**Faculté des Sciences et Techniques de Tanger (F.S.T.T)**

**Département : Génie Informatique**

**LST – AD**

# **RAPPORT DE PROJET**

**SUJET : Gestion de projets collaboratifs "Freelance"**

**MODULE : FONDAMENTAUX DE BASES DE DONNÉES**

**Conception et Modélisation d'un Système d'Information "Manalytics"**



**- Sous la supervision de :**

Pr. Ezziyyani Mostafa

**- Réalisé par :**

Souhaila Benaouate  
Manal Rhoni Aref

Année Universitaire : 2025/2026

## REMERCIEMENTS

Nous tenons à exprimer nos sincères remerciements à notre professeur, **M. Ezziyyani Mostafa**, pour la qualité de son enseignement et pour nous avoir transmis les bases fondamentales nécessaires à la réalisation de ce projet au sein du module "Fondamentaux de Bases de Données".

Nos remerciements s'adressent également à l'équipe de l'entreprise **Manalytics** pour leur collaboration précieuse:

- À **Mme. Najlae Najdi Akel**, pour son expertise en tant que Chef de Projet.
- À **Mme. Hafsa Eladdam**, pour son support technique et son implication.

Ce travail est le fruit de nos recherches personnelles et de notre investissement dans la conception d'un système d'information robuste pour la gestion des projets collaboratifs.

# SOMMAIRE

<b>1. INTRODUCTION GÉNÉRALE .....</b>	<b>5</b>
1.1. Contexte et Problématique .....	5
1.2. Objectifs du Système .....	5
<b>2. ANALYSE ET DICTIONNAIRE DE DONNÉES .....</b>	<b>5</b>
2.1. Recueil des informations .....	5
2.2. Dictionnaire de Données (DD) .....	6
<b>3. RÈGLES DE GESTION (RG) .....</b>	<b>7</b>
3.1. Gestion des Acteurs .....	7
3.2. Gestion des Projets et Tâches .....	7
3.3. Gestion Contractuelle .....	7
<b>4. FORMALISATION ET NORMALISATION .....</b>	<b>8</b>
4.1. Liste des Dépendances Fonctionnelles (DF) .....	8
4.2. Validation des Formes Normales (1FN, 2FN, 3FN) .....	8
<b>5. MODÈLE CONCEPTUEL DE DONNÉES (MCD) .....</b>	<b>9</b>
5.1. Description des Entités et Associations .....	9
5.2. Schéma Conceptuel .....	10
<b>6. MODÈLE LOGIQUE DE DONNÉES (MLD) .....</b>	<b>11</b>
6.1. Règles de Passage MCD vers MLD .....	11
6.2. Schéma Relationnel Final .....	11

<b>7. IMPLÉMENTATION ET MANIPULATION SQL .....</b>	<b>12</b>
7.1. Script de Création de la Base (DDL) .....	12
7.2. Jeu d'Essai (Insertion de Données) .....	15
7.3. Requêtes d'Interrogation (DML) .....	15
7.4. Optimisation et Vues SQL (Views) .....	17
<b>ANNEXE : MAQUETTES DE L'INTERFACE UTILISATEUR (IHM) .....</b>	<b>18</b>
<b>8. CONCLUSION GÉNÉRALE .....</b>	<b>20</b>

# 1. INTRODUCTION GÉNÉRALE

## 1.1. Contexte et Problématique

Dans un contexte professionnel marqué par la digitalisation et la flexibilité du travail, l'entreprise "**Manalytics**" souhaite structurer ses interactions avec ses collaborateurs externes.

Actuellement, la gestion des projets, des prestataires (freelances) et des contrats se fait de manière dispersée, entraînant des risques d'incohérence et de perte d'information.

La problématique centrale de ce projet est la suivante : *Comment concevoir un système d'information robuste permettant de centraliser la gestion des projets, depuis l'expression du besoin jusqu'à la facturation, tout en assurant le suivi opérationnel des tâches ?*

## 1.2. Objectifs du Système

Le futur système, nommé "**Manalytics Connect**", doit répondre aux impératifs suivants :

- **Gestion administrative** : Identifier de manière unique les clients (entreprises) et les freelances, en respectant les contraintes fiscales marocaines (ICE, CIN).
- **Suivi opérationnel** : Découper les projets en tâches unitaires (Lots) avec un suivi précis de l'avancement et des priorités.
- **Supervision** : Intégrer le rôle de "Chef de Projet" (Administrateur) pour valider les étapes.
- **Adéquation profil/poste** : Gérer les compétences techniques pour affecter le bon freelance à la bonne tâche.

# 2. ANALYSE ET DICTIONNAIRE DE DONNÉES

## 2.1. Recueil des informations

La première phase de notre démarche MERISE a consisté à recenser l'ensemble des données élémentaires manipulées par l'entreprise. Nous avons distingué les données relatives aux tiers (Clients, Freelances) de celles relatives à la production (Projets, Tâches).

## 2.2. Dictionnaire de Données (DD)

Le tableau suivant présente les propriétés retenues après épuration des synonymes et polysèmes.

Code Mnémonique	Libellé (Désignation)	Type	Nature	Taille
Num_Cli	Identifiant Client	N	E	10

<b>Raison_Soc</b>	Raison Sociale (Entr.)	AN	E	60
<b>ICE_Cli</b>	Identifiant Commun Entr.	N	E	15
<b>Adr_Cli</b>	Adresse du Siège	AN	E	100
<b>Num_Admin</b>	Identifiant Chef de Projet	N	E	5
<b>Nom_Admin</b>	Nom Chef de Projet	A	E	30
<b>Num_Free</b>	Identifiant Freelance	N	E	10
<b>Nom_Free</b>	Nom du Freelance	A	E	30
<b>Prenom_Free</b>	Prénom du Freelance	A	E	30
<b>CIN_Free</b>	CIN Freelance	AN	E	10
<b>TJM_Free</b>	Taux Journalier Moyen	M	E	Single
<b>Cod_Comp</b>	Code Compétence	AN	E	5
<b>Lib_Comp</b>	Libellé Compétence	A	E	40
<b>Niv_Maitrise</b>	Niveau de Maîtrise	N	E	1
<b>Ref_Proj</b>	Référence Projet	AN	E	15
<b>Titre_Proj</b>	Titre du Projet	AN	E	80
<b>Date_Deb_P</b>	Date Début Projet	D	E	10
<b>Date_Fin_P</b>	Date Fin Prévue	D	E	10
<b>Statut_Proj</b>	État du Projet	A	E	20
<b>Num_Lot</b>	Numéro de Tâche (Lot)	N	E	8
<b>Desc_Lot</b>	Description Tâche	AN	E	255
<b>Priorite_Lot</b>	Niveau Priorité	A	E	10
<b>Avanc_Lot</b>	Taux d'Avancement (%)	N	E	3

<b>Date_Ech_Lot</b>	Date Échéance Tâche	D	E	10
<b>Ref_Contrat</b>	Référence Contrat	AN	E	20
<b>Typ_Contrat</b>	Type (Forfait/Régie)	A	E	15
<b>Mnt_Contrat</b>	Montant Total	M	E	Double

### 3. RÈGLES DE GESTION (RG)

Les règles de gestion suivantes traduisent la logique métier de "Manalytics" et servent de base à la construction des dépendances fonctionnelles.

#### 3.1. Gestion des Acteurs

- **RG1** : Un **Client** est une entité morale unique, identifiée par son ICE.
- **RG2** : Un **Freelance** est une personne physique unique, identifiée par sa CIN.
- **RG3** : Un **Chef de Projet** (Admin) est un collaborateur interne responsable de la supervision.

#### 3.2. Gestion des Projets et Tâches

- **RG4** : Un **Projet** est commandé par un seul Client et supervisé par un seul Chef de Projet.
- **RG5** : Pour une gestion fine, un **Projet** est décomposé en une ou plusieurs **Tâches (Lots)**. Une tâche appartient à un seul projet.
- **RG6** : Un **Projet** ne peut être considéré comme "Terminé" que si l'avancement de toutes ses tâches est à 100%.

#### 3.3. Gestion Contractuelle et Compétences

- **RG7** : Un **Freelance** possède une ou plusieurs **Compétences** avec un niveau de maîtrise spécifique (1 à 5).
- **RG8** : Une tâche requiert des compétences spécifiques.
- **RG9** : Une tâche est réalisée par un seul **Freelance**, officialisée par un **Contrat** qui définit le montant et le type de facturation.

### 4. FORMALISATION ET NORMALISATION

Afin de garantir la cohérence de la base de données, nous avons établi la liste des dépendances fonctionnelles (DF) et vérifié le respect des formes normales.

## 4.1. Liste des Dépendances Fonctionnelles (DF)

Soit la relation  $A \rightarrow B$  signifiant que la connaissance de A détermine une valeur unique de B,

On a :

- $\text{Num\_Cli} \rightarrow \text{Raison\_Soc}, \text{ICE\_Cli}, \text{Adr\_Cli}$
- $\text{Num\_Admin} \rightarrow \text{Nom\_Admin}$
- $\text{Num\_Free} \rightarrow \text{Nom\_Free}, \text{Prenom\_Free}, \text{CIN\_Free}, \text{TJM\_Free}$
- $\text{Ref\_Proj} \rightarrow \text{Titre\_Proj}, \text{Date\_Deb\_P}, \text{Date\_Fin\_P}, \text{Statut\_Proj}, \text{Num\_Cli}, \text{Num\_Admin}$   
*(DF liant le projet à son client et son superviseur).*
- $\text{Num\_Lot} \rightarrow \text{Desc\_Lot}, \text{Priorite\_Lot}, \text{Avanc\_Lot}, \text{Date\_Ech\_Lot}, \text{Ref\_Proj}$   
*(DF liant la tâche à son projet parent).*
- $(\text{Num\_Free}, \text{Cod\_Comp}) \rightarrow \text{Niv\_Maitrise}$   
*(DF composée : le niveau dépend du couple Freelance + Compétence).*
- $\text{Ref\_Contrat} \rightarrow \text{Typ\_Contrat}, \text{Mnt\_Contrat}, \text{Num\_Lot}, \text{Num\_Free}$   
*(Le contrat lie une tâche spécifique à un freelance).*

## 4.2. Validation des Formes Normales

- **1ère Forme Normale (1FN) :** Tous les attributs sont atomiques (non décomposables) et chaque entité possède un identifiant. **(Validé)**
- **2ème Forme Normale (2FN) :** Toutes les propriétés non-clés dépendent de la totalité de la clé primaire. La DF n°6 a respecté ce principe. **(Validé)**
- **3ème Forme Normale (3FN) :** Il n'existe pas de dépendance transitive entre les attributs non-clés. Par exemple, l'adresse du client n'est pas stockée dans l'entité Projet, mais reste dans l'entité Client. **(Validé)**

## 5. MODÈLE CONCEPTUEL DE DONNÉES (MCD)

Le MCD suivant représente graphiquement la structure statique du système d'information.



## 5.1. Description des Entités et Associations

- **Entités :**

- CLIENT
- ADMIN
- FREELANCE
- COMPÉTENCE
- PROJET
- LOT
- CONTRAT

- **Associations principales :**

- **COMMANDER :**

Un Client commande (1,n) Projets.

- **SUPERVISER :**

Un Admin supervise (1,n) Projets.

- **COMPOSER :**

Un Projet est composé de (1,n) Lots.

- **MAÎTRISER :**

Un Freelance maîtrise (1,n) Compétences (Porteuse de données : Niveau).

- **REQUERIR :**

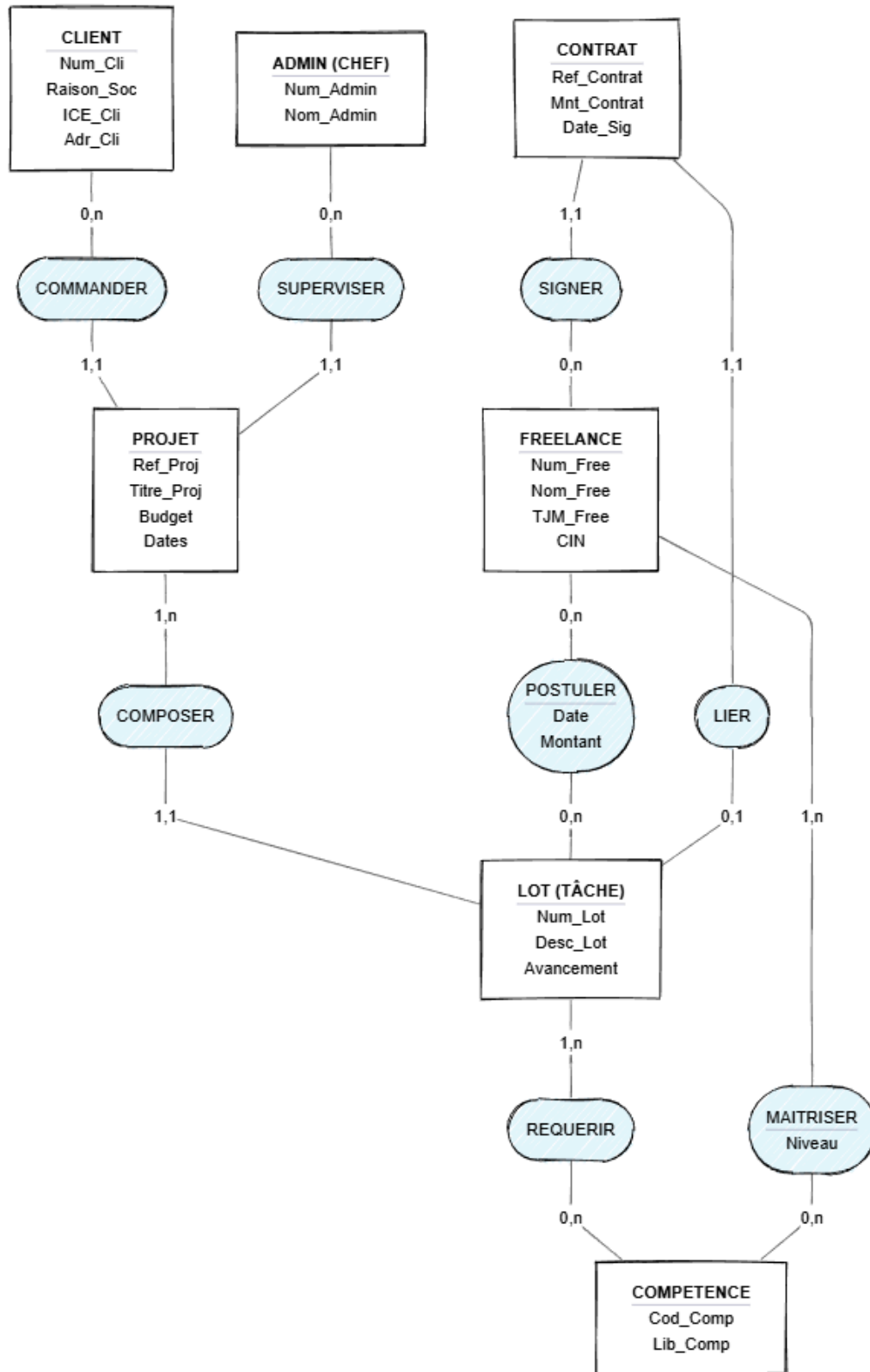
Un lot requiert (1,n) Compétences.

- **CONTRACTUALISER :**

Un Contrat lie (1,1) Lot et (1,1) Freelance.

## 5.2. Schéma Conceptuel

Nous avons voulu gérer la réalité du métier : un projet n'est pas monolithique, il est découpé en tâches (Lots). Et pour affecter un freelance, il ne suffit pas de connaître son nom, il faut valider son niveau de compétence. C'est pour cela que nous avons des relations porteuses de données.



## 6. MODÈLE LOGIQUE DE DONNÉES (MLD)

Le passage au niveau logique consiste à traduire le MCD en un schéma relationnel (Tables), en appliquant les règles de transformation (migration de clés, création de tables de jointure).

### 6.1. Schéma Relationnel

Les clés primaires sont **soulignées** et les clés étrangères sont précédées d'un #.

- **CLIENT** (Num\_Cli, Raison\_Soc, ICE\_Cli, Adr\_Cli)
- **ADMIN** (Num\_Admin, Nom\_Admin)
- **FREELANCE** (Num\_Free, Nom\_Free, Prenom\_Free, CIN\_Free, TJM\_Free)
- **COMPETENCE** (Cod\_Comp, Lib\_Comp)
- **PROJET** (Ref\_Proj, Titre\_Proj, Date\_Deb\_P, Date\_Fin\_P, Statut\_Proj, #Num\_Cli, #Num\_Admin)

*(Les clés étrangères #Num\_Cli et #Num\_Admin migrent ici car les cardinalités côté Projet sont (1,1)).*

- **LOT** (Num\_Lot, Desc\_Lot, Priorite\_Lot, Avanc\_Lot, Date\_Ech\_Lot, #Ref\_Proj)

*(La clé #Ref\_Proj migre ici pour lier la tâche à son projet).*

- **MAÎTRISER** (#Num\_Free, #Cod\_Comp, Niv\_Maitrise)

*(Table issue de la relation N-N entre Freelance et Compétence).*

- **REQUERIR** (#Num\_Lot, #Cod\_Comp)

*(Table de liaison entre Tâche et Compétence).*

- **CONTRAT** (Ref\_Contrat, Typ\_Contrat, Mnt\_Contrat, #Num\_Lot, #Num\_Free)

*(Table centrale de gestion : elle associe le travail (Lot) à la ressource (Freelance)).*

## 7. IMPLÉMENTATION ET MANIPULATION SQL

Cette section présente la concrétisation technique du modèle sur le SGBD MySQL (XAMPP).

### 7.1. Script de Création de la Base (DDL)

#### -- Création des tables principales

```
CREATE TABLE ADMIN (  
    Num_Admin INT PRIMARY KEY,  
    Nom_Admin VARCHAR(50),  
    Email_Admin VARCHAR(100)  
);
```

```
CREATE TABLE CLIENT (  
    Num_Cli INT PRIMARY KEY,  
    Raison_Soc VARCHAR(100),  
    ICE_Cli VARCHAR(20),  
    Adr_Cli VARCHAR(150)  
);
```

```
CREATE TABLE FREELANCE (  
    Num_Free INT PRIMARY KEY,  
    Nom_Free VARCHAR(50),  
    Prenom_Free VARCHAR(50),  
    CIN_Free VARCHAR(20),  
    TJM_Free DECIMAL(10, 2)  
);
```

```
CREATE TABLE COMPETENCE (  
    Cod_Comp VARCHAR(10) PRIMARY KEY,  
    Lib_Comp VARCHAR(50)  
);
```

**-- Création des tables avec clés étrangères**

```
CREATE TABLE PROJET (  
    Ref_Proj VARCHAR(20) PRIMARY KEY,  
    Titre_Proj VARCHAR(100),  
    Date_Deb_P DATE,  
    Date_Fin_P DATE,  
    Statut_Proj VARCHAR(20),  
    Num_Cli INT,  
    Num_Admin INT,  
    FOREIGN KEY (Num_Cli) REFERENCES CLIENT(Num_Cli),  
    FOREIGN KEY (Num_Admin) REFERENCES ADMIN(Num_Admin)  
);
```

```
CREATE TABLE LOT (  
    Num_Lot INT PRIMARY KEY,  
    Desc_Lot VARCHAR(255),  
    Priorite_Lot VARCHAR(20),  
    Avanc_Lot INT,  
    Date_Ech_Lot DATE,  
    Ref_Proj VARCHAR(20),  
    FOREIGN KEY (Ref_Proj) REFERENCES PROJET(Ref_Proj)  
);
```

**-- Tables de liaison**

```
CREATE TABLE MAÎTRISER (  
    Num_Free INT,  
    Cod_Comp VARCHAR(10),  
    Niv_Maitrise INT,  
    PRIMARY KEY (Num_Free, Cod_Comp),  
    FOREIGN KEY (Num_Free) REFERENCES FREELANCE(Num_Free),  
    FOREIGN KEY (Cod_Comp) REFERENCES COMPETENCE(Cod_Comp)  
);
```

```

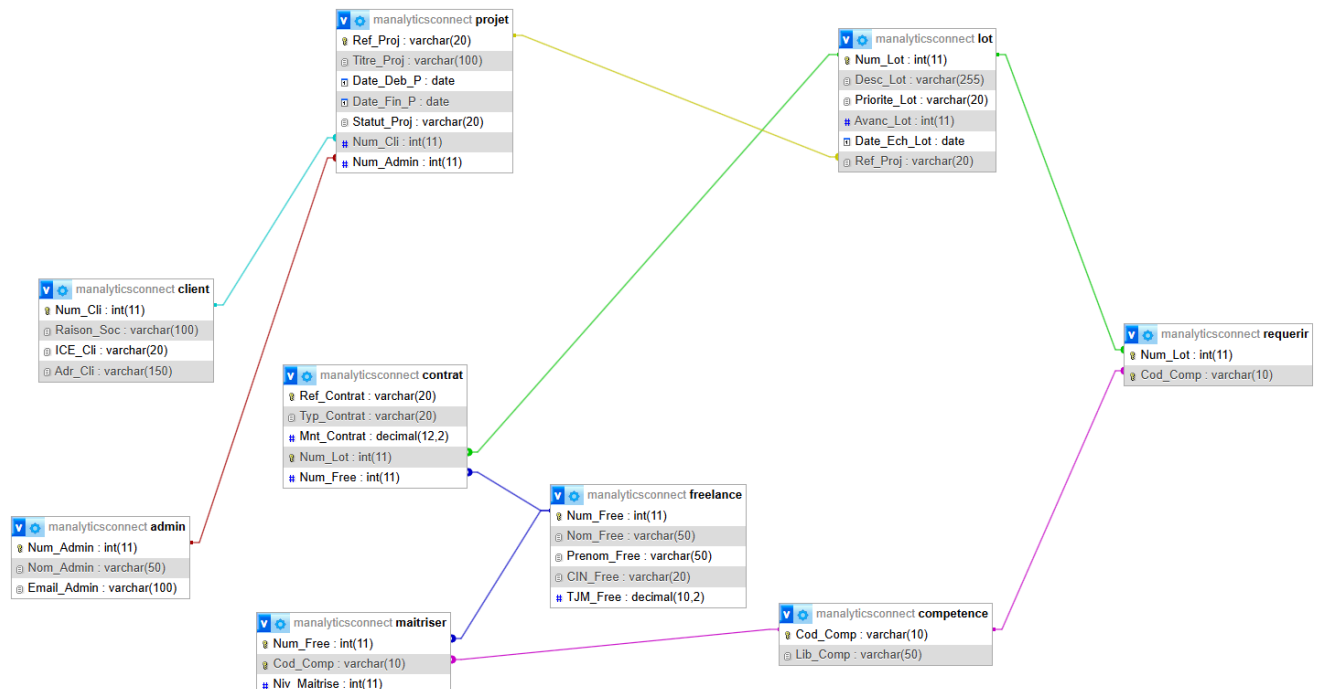
CREATE TABLE REQUERIR (
    Num_Lot INT,
    Cod_Comp VARCHAR(10),
    PRIMARY KEY (Num_Lot, Cod_Comp),
    FOREIGN KEY (Num_Lot) REFERENCES LOT(Num_Lot),
    FOREIGN KEY (Cod_Comp) REFERENCES COMPETENCE(Cod_Comp)
);

```

```

CREATE TABLE CONTRAT (
    Ref_Contrat VARCHAR(20) PRIMARY KEY,
    Typ_Contrat VARCHAR(20),
    Mnt_Contrat DECIMAL(12, 2),
    Num_Lot INT UNIQUE,
    Num_Free INT,
    FOREIGN KEY (Num_Lot) REFERENCES LOT(Num_Lot),
    FOREIGN KEY (Num_Free) REFERENCES FREELANCE(Num_Free)
);

```



## 7.2. Jeu d'Essai (Insertion de Données)

Pour valider le fonctionnement, nous avons inséré un jeu de données représentatif.

### -- Insertion des données de référence

```
INSERT INTO ADMIN VALUES (1, 'Menal Admin', 'menal@manalytics.ma');  
INSERT INTO CLIENT VALUES (100, 'Maroc Telecom', 'ICE123456789', 'Rabat, Maroc');  
INSERT INTO FREELANCE VALUES (50, 'Alami', 'Ahmed', 'K123456', 2500.00);  
INSERT INTO COMPETENCE VALUES ('SQL', 'Base de données SQL');
```

### -- Insertion du flux de travail

```
INSERT INTO MAITRISER VALUES (50, 'SQL', 4);  
INSERT INTO PROJET VALUES ('PRJ-01', 'Refonte Dashboard', '2025-01-01', '2025-06-01',  
'En cours', 100, 1);  
INSERT INTO LOT VALUES (10, 'Extraction Données', 'Haute', 20, '2025-02-01', 'PRJ-01');  
INSERT INTO CONTRAT VALUES ('CTR-001', 'Forfait', 15000.00, 10, 50);
```

## 7.3. Requêtes d'Interrogation (DML)

Voici cinq requêtes types démontrant la capacité du système à répondre aux besoins d'information.

**Q1 : Afficher la liste des projets en cours avec le nom de l'entreprise cliente.**

```
SELECT P.Titre_Proj, P.Statut_Proj, C.Raison_Soc  
FROM PROJET P  
JOIN CLIENT C ON P.Num_Cli = C.Num_Cli  
WHERE P.Statut_Proj = 'En cours';
```

Titre_Proj	Statut_Proj	Raison_Soc
Refonte Dashboard	En cours	Maroc Telecom

**Q2 : Trouver les freelances maîtrisant 'SQL' avec un niveau supérieur ou égal à 4.**

```
SELECT F.Nom_Free, F.Prenom_Free, M.Niv_Maitrise
FROM FREELANCE F
JOIN MAITRISER M ON F.Num_Free = M.Num_Free
WHERE M.Cod_Comp = 'SQL' AND M.Niv_Maitrise >= 4;
```

Nom_Free	Prenom_Free	Niv_Maitrise
Alami	Ahmed	4

**Q3 : Calculer le montant total des contrats par projet.**

```
SELECT P.Titre_Proj, SUM(C.Mnt_Contrat) as Total_Depense
FROM PROJET P
JOIN LOT L ON P.Ref_Proj = L.Ref_Proj
JOIN CONTRAT C ON L.Num_Lot = C.Num_Lot
GROUP BY P.Ref_Proj;
```

Titre_Proj	Total_Depense
Refonte Dashboard	15000.00

## 7.4. Optimisation et Vues SQL (Views)

Afin de simplifier l'accès aux données pour les gestionnaires (qui ne maîtrisent pas le langage SQL complexe), nous avons créé des **Vues** (Tables virtuelles). Ces vues permettent de masquer la complexité des jointures.



## **Vue 1 : Résumé Global des Projets**

Cette vue permet de voir en un coup d'œil l'état d'un projet, son client, et le montant total engagé, sans avoir à faire de jointures manuelles.

```
CREATE VIEW V_RESUME_PROJETS AS
SELECT
    P.Ref_Proj, P.Titre_Proj, C.Raison_Soc AS Client, P.Statut_Proj,
    COUNT(L.Num_Lot) AS Nombre_Taches,
    COALESCE(SUM(CT.Mnt_Contrat), 0) AS Budget_Engage
FROM PROJET P
JOIN CLIENT C ON P.Num_Cli = C.Num_Cli
LEFT JOIN LOT L ON P.Ref_Proj = L.Ref_Proj
LEFT JOIN CONTRAT CT ON L.Num_Lot = CT.Num_Lot
GROUP BY P.Ref_Proj, P.Titre_Proj, C.Raison_Soc, P.Statut_Proj;
```

### **Exemple d'utilisation par un manager :**

-- Le manager veut voir les projets qui dépassent 50 000 DH de budget engagé

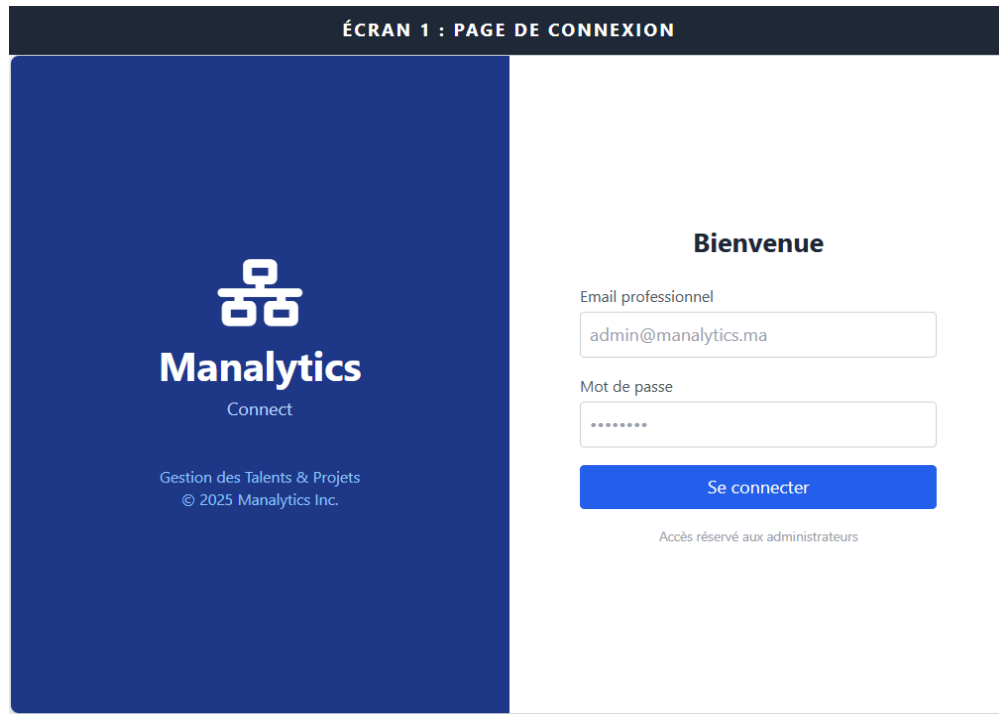
```
SELECT * FROM V_RESUME_PROJETS WHERE Budget_Engage > 50000;
```

Ref_Proj	Titre_Proj	Client	Statut_Proj	Nombre_Taches	Budget_Engage
PRJ-2025-01	Refonte Dashboard	Maroc Telecom	En cours	1	15000.00

## **ANNEXE : MAQUETTES DE L'INTERFACE UTILISATEUR (IHM)**

Afin de projeter l'utilisation réelle de la base de données "Manalytics Connect", nous avons réalisé le prototypage des interfaces principales de la future application Web. Ces interfaces ont été conçues pour être ergonomiques et centrées sur les besoins des chefs de projet.

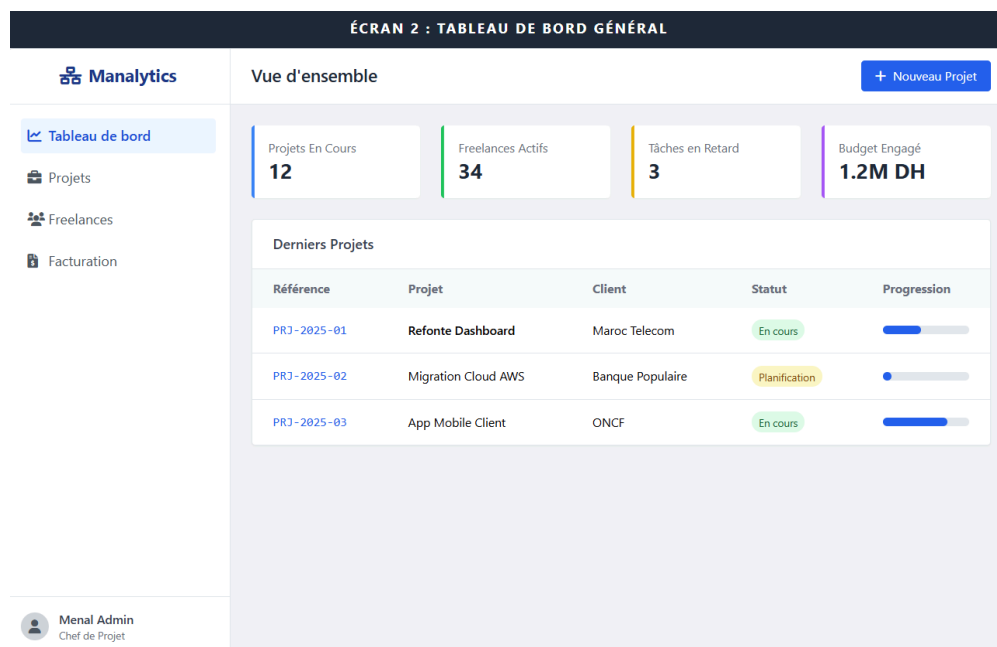
**Figure 1 :** Écran de Connexion Cette interface sécurisée permet aux administrateurs et chefs de projet de s'authentifier pour accéder au back-office de gestion.



**Figure 2 :** Tableau de Bord Général (Dashboard)

Le tableau de bord offre une vue synthétique de l'activité.

Il affiche les indicateurs clés de performance (KPI) calculés en temps réel grâce aux requêtes d'agrégation SQL (Nombre de projets actifs, Budget engagé, Tâches en retard).

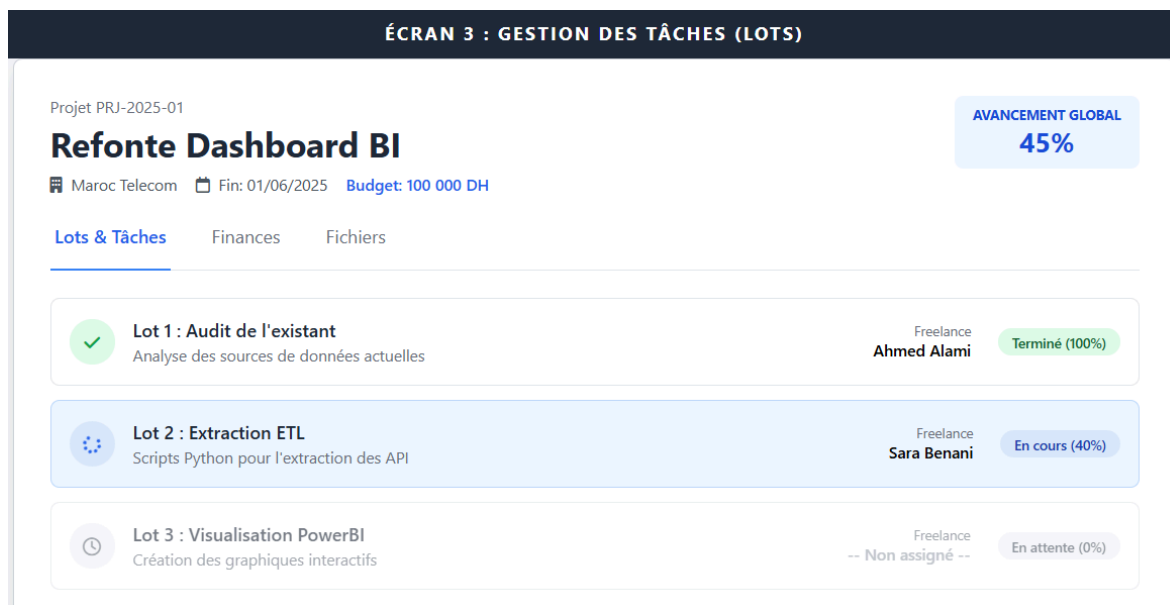


**Figure 3 : Détail d'un Projet et Gestion des Lots**

Cette vue détaillée permet de piloter un projet spécifique.

On y retrouve les informations du client, le budget global, et surtout la liste des lots (tâches).

Pour chaque lot, le système affiche le freelance assigné (résultat de la requête de jointure) et la barre de progression de la tâche.



## 8. CONCLUSION GÉNÉRALE

Ce travail de conception nous a permis de structurer une base de données cohérente et normalisée pour l'entreprise "Manalytics".

L'intégration des besoins fonctionnels (gestion des tâches, supervision par chef de projet) et des contraintes techniques (normalisation 3FN) aboutit à un **Modèle Logique de Données (MLD)** stable. Ce modèle élimine les redondances et garantit l'intégrité des informations.

La phase d'implémentation SQL a validé la faisabilité technique du modèle. Ce dossier constitue le socle indispensable pour le développement futur des interfaces utilisateurs (Front-End) et l'intégration complète du système d'information.