

# LIVRABLE PROJET ORGANI'ZEUR

Organiz'heure est une application web intuitive et performante conçue pour optimiser la gestion des tâches et des listes au quotidien. Cette solution s'adresse aussi bien aux particuliers qu'aux professionnels cherchant un outil simple et efficace pour organiser leurs activités.

Nathan  
Razafindrakoto

## Table des matières

Dictionnaire de données : .....	2
Dépendance fonctionnelle.....	4
MCD.....	5
Script SQL .....	5

# Dictionnaire de données :

## Utilisateur

- **Nom de la table** : utilisateurs
- **Description** : Représente un utilisateur de l'application.
- **Attributs** :
  - id\_utilisateur (INT, PK) : Identifiant unique de l'utilisateur.
  - nom (VARCHAR) : Nom de l'utilisateur.
  - prenom (VARCHAR) : Prénom de l'utilisateur.
  - email (VARCHAR, UNIQUE) : Adresse e-mail de l'utilisateur (identifiant de connexion).
  - mot\_de\_passe (VARCHAR) : Mot de passe crypté pour la connexion.
  - role (ENUM: 'administrateur', 'utilisateur') : Rôle de l'utilisateur, défini par l'administrateur.
  - date\_creation (DATETIME) : Date de création de l'utilisateur.
  - date\_modification (DATETIME) : Date de la dernière modification des informations de l'utilisateur.
  - actif (BOOLEAN) : Indique si le compte est actif ou désactivé.

## 2. Liste

- **Nom de la table** : listes
- **Description** : Représente une liste de tâches créée par un utilisateur.
- **Attributs** :
  - id\_liste (INT, PK) : Identifiant unique de la liste.
  - nom\_liste (VARCHAR) : Nom de la liste.
  - id\_utilisateur\_createur (INT, FK) : Identifiant de l'utilisateur ayant créé la liste.
  - date\_creation (DATETIME) : Date de création de la liste.

- date\_modification (DATETIME) : Date de la dernière modification de la liste.
- date\_archivage (DATETIME, NULL) : Date d'archivage de la liste, NULL si la liste n'est pas archivée.
- archivée (BOOLEAN) : Indique si la liste est archivée.
- id\_utilisateur\_derniere\_modification (INT, FK) : Identifiant de l'utilisateur ayant effectué la dernière modification sur la liste.

### 3. Tâche

- **Nom de la table** : taches
- **Description** : Représente une tâche rattachée à une liste de tâches.
- **Attributs** :
  - id\_tache (INT, PK) : Identifiant unique de la tâche.
  - libelle (VARCHAR) : Description de la tâche.
  - date\_echeance (DATETIME) : Date et heure d'échéance de la tâche.
  - etat (ENUM: 'fait', 'non fait') : État de la tâche.
  - date\_creation (DATETIME) : Date de création de la tâche.
  - date\_modification (DATETIME) : Date de la dernière modification de la tâche.
  - date\_completement (DATETIME, NULL) : Date à laquelle la tâche a été marquée comme "faite", NULL si non faite.
  - id\_utilisateur\_createur (INT, FK) : Identifiant de l'utilisateur qui a créé la tâche.
  - id\_utilisateur\_derniere\_modification (INT, FK) : Identifiant de l'utilisateur ayant effectué la dernière modification de la tâche.
  - id\_utilisateur\_completement (INT, FK, NULL) : Identifiant de l'utilisateur ayant marqué la tâche comme "faite" (NULL si non fait).
  - id\_liste (INT, FK) : Identifiant de la liste à laquelle la tâche appartient.

### 4. Catégorie

- **Nom de la table** : categories
- **Description** : Représente une catégorie à laquelle des listes de tâches peuvent être associées.

- **Attributs :**

- id\_categorie (INT, PK) : Identifiant unique de la catégorie.
- nom\_categorie (VARCHAR) : Nom de la catégorie.
- date\_creation (DATETIME) : Date de création de la catégorie.
- date\_modification (DATETIME) : Date de la dernière modification de la catégorie.

## Dépendance fonctionnelle

### ? **Utilisateur :**

- ID\_Utilisateur → Nom, Prénom, Email, Mot\_de\_passe, Rôle, Date\_creation, Date\_modification, Actif

### ? **Liste :**

- ID\_Liste → Nom\_Liste, ID\_Utilisateur\_Creator, Date\_Creation, Date\_Modification, Date\_Archivage, Archivée, ID\_Utilisateur\_Dernière\_Modification

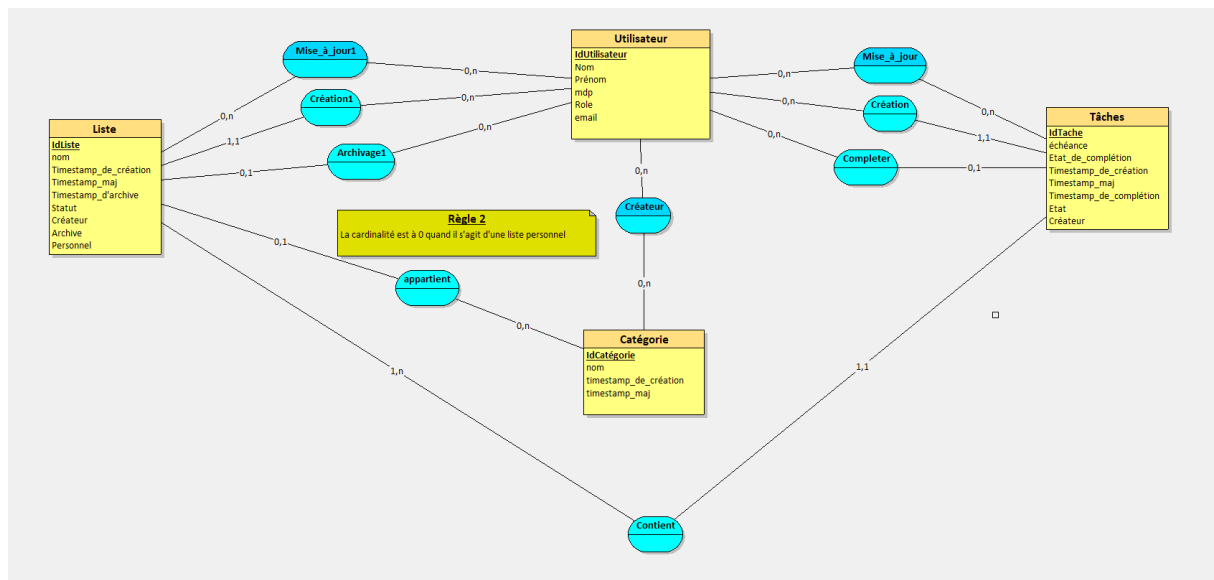
### ? **Tâche :**

- ID\_Tache → Libelle, Date\_Echeance, Etat, Date\_Creation, Date\_Modification, Date\_Completement, ID\_Utilisateur\_Creator, ID\_Utilisateur\_Dernière\_Modification, ID\_Utilisateur\_Completement, ID\_Liste
- ID\_Tache → (ID\_Utilisateur\_Completement, Date\_Completement) [Si la tâche est marquée comme "faite", cette dépendance est remplie].

### ? **Catégorie :**

- ID\_Categorie → Nom\_Categorie, Date\_Creation, Date\_Modification

# MCD



## Script SQL

```
CREATE TABLE Utilisateur(
    IdUtilisateur INT AUTO_INCREMENT,
    Nom VARCHAR(50),
    Prénom VARCHAR(50),
    mdp VARCHAR(50),
    Role BOOLEAN,
    email VARCHAR(50),
    PRIMARY KEY(IdUtilisateur)
);
```

```
CREATE TABLE Catégorie(
```

```
IdCatégorie INT AUTO_INCREMENT,  
nom VARCHAR(50) NOT NULL,  
timestamp_de_création DATETIME,  
timestamp_maj DATETIME,  
PRIMARY KEY(IdCatégorie)  
);
```

```
CREATE TABLE Liste(  
    IdListe INT AUTO_INCREMENT,  
    nom VARCHAR(50),  
    Timestamp_de_création DATETIME,  
    Timestamp_maj DATETIME,  
    Timestamp_d_archive DATETIME,  
    Statut BOOLEAN,  
    Créateur VARCHAR(50),  
    Archive BOOLEAN,  
    Personnel BOOLEAN,  
    IdUtilisateur INT NOT NULL,  
    IdUtilisateur_1 INT,  
    IdCatégorie INT,  
    PRIMARY KEY(IdListe),  
    FOREIGN KEY(IdUtilisateur) REFERENCES Utilisateur(IdUtilisateur),  
    FOREIGN KEY(IdUtilisateur_1) REFERENCES Utilisateur(IdUtilisateur),  
    FOREIGN KEY(IdCatégorie) REFERENCES Catégorie(IdCatégorie)  
);
```

```
CREATE TABLE Tâches(  
    IdTache INT AUTO_INCREMENT,
```

```
    échéance DATETIME,  
    Etat_de_complétion BOOLEAN,  
    Timestamp_de_création DATETIME,  
    Timestamp_maj DATETIME,  
    Timestamp_de_complétion DATETIME,  
    Etat BOOLEAN,  
    Créateur VARCHAR(50),  
    IdUtilisateur INT,  
    IdListe INT NOT NULL,  
    IdUtilisateur_1 INT NOT NULL,  
    PRIMARY KEY(IdTache),  
    FOREIGN KEY(IdUtilisateur) REFERENCES Utilisateur(IdUtilisateur),  
    FOREIGN KEY(IdListe) REFERENCES Liste(IdListe),  
    FOREIGN KEY(IdUtilisateur_1) REFERENCES Utilisateur(IdUtilisateur)  
);
```

```
CREATE TABLE Mise_à_jour1(  
    IdUtilisateur INT,  
    IdListe INT,  
    PRIMARY KEY(IdUtilisateur, IdListe),  
    FOREIGN KEY(IdUtilisateur) REFERENCES Utilisateur(IdUtilisateur),  
    FOREIGN KEY(IdListe) REFERENCES Liste(IdListe)  
);
```

```
CREATE TABLE Mise_à_jour(  
    IdUtilisateur INT,  
    IdTache INT,  
    PRIMARY KEY(IdUtilisateur, IdTache),
```



```
FOREIGN KEY(IdUtilisateur) REFERENCES Utilisateur(IdUtilisateur),  
FOREIGN KEY(IdTache) REFERENCES Tâches(IdTache)  
);
```

```
CREATE TABLE Créateur(  
  IdUtilisateur INT,  
  IdCatégorie INT,  
  PRIMARY KEY(IdUtilisateur, IdCatégorie),  
  FOREIGN KEY(IdUtilisateur) REFERENCES Utilisateur(IdUtilisateur),  
  FOREIGN KEY(IdCatégorie) REFERENCES Catégorie(IdCatégorie)  
);
```

## Schéma d'architecture fonctionnelle

