

# Rapport de description – La Gestion des documents

## 1. Présentation générale

Cette application de gestion de documents permet :

- L'ajout, la modification, la suppression et l'archivage des documents.
- La gestion des utilisateurs et des emprunts de documents.
- La consultation de documents purgés.

### Frontend :

- Développé en HTML/CSS/JS pur.
- Affichage dynamique des documents avec tri, filtres, recherche et modals pour ajouter ou modifier un document.

### Backend :

- Node.js + Express pour la logique serveur.
- PostgreSQL pour le stockage des données.

## 2. Gestion des documents

### Routes principales :

- **GET /documents** → Liste tous les documents.
- **POST /documents** → Ajouter un document.
- **PATCH /documents/:id** → Modifier un document.
- **PATCH /documents/:id/archive** → Archiver un document.
- **DELETE /documents/:id** → Supprimer (soft delete) un document.
- **DELETE /documents/:id/purge** → Supprimer définitivement un document.
- **PATCH /documents/:id/restore** → Restaurer un document archivé ou supprimé.
- **GET /documents/purges** → Liste des documents purgés

### Fonctionnalités frontend :

- Affichage d'un tableau avec tri par date.
- Filtres par statut : disponible, archivé, supprimé, purgé, tous.
- Barre de recherche sur le titre.
- Formulaire modal pour ajouter ou modifier un document
- Boutons d'actions : modifier, archiver, supprimer, restaurer, purger.

### 3. Base de données (PostgreSQL)

Tables principales :

#### **Documents :**

Colonne	Type	Description
id	SERIAL	Identifiant unique
titre	TEXT	Titre du document
auteur	TEXT	Auteur du document
Date_arrivee	DATE	Date d'ajout
statut	TEXT	disponible / archivé / supprimé

#### **Documents\_purges :**

Colonne	Type	Description
id	SERIAL	Identifiant unique
titre	TEXT	Titre du document
auteur	TEXT	Auteur du document
Ancien_statut	TEXT	Statut avant purge
date_purge	TIMESTAMP	Date de purge (automatique)

## 4. Flux des données

### Ajout / modification

- Frontend → formulaire modal → POST ou PATCH /documents → Base de données → retour JSON → mise à jour du tableau.

### Archivage / suppression / restauration

- Boutons → PATCH /documents/:id/archive ou DELETE /documents/:id ou PATCH /documents/:id/restore → Base de données → retour JSON → rechargement tableau.

### Purge définitive

- Bouton Purger → DELETE /documents/:id/purge → sauvegarde dans documents\_purges → suppression définitive → mise à jour tableau.

### Filtrage et recherche

- Filtre par statut → modification de statusFilter côté frontend → fetch /documents ou /documents/purges → affichage tableau.
- Recherche par titre → filtrage côté frontend sur currentDocuments.

## 5. Conclusion

- Tous les cas d'usage du module 5 sont couverts : ajout, modification, archivage, suppression, restauration et purge.
- Les interactions frontend ↔ backend sont cohérentes.
- La base PostgreSQL est normalisée et permet de conserver l'historique des purges.