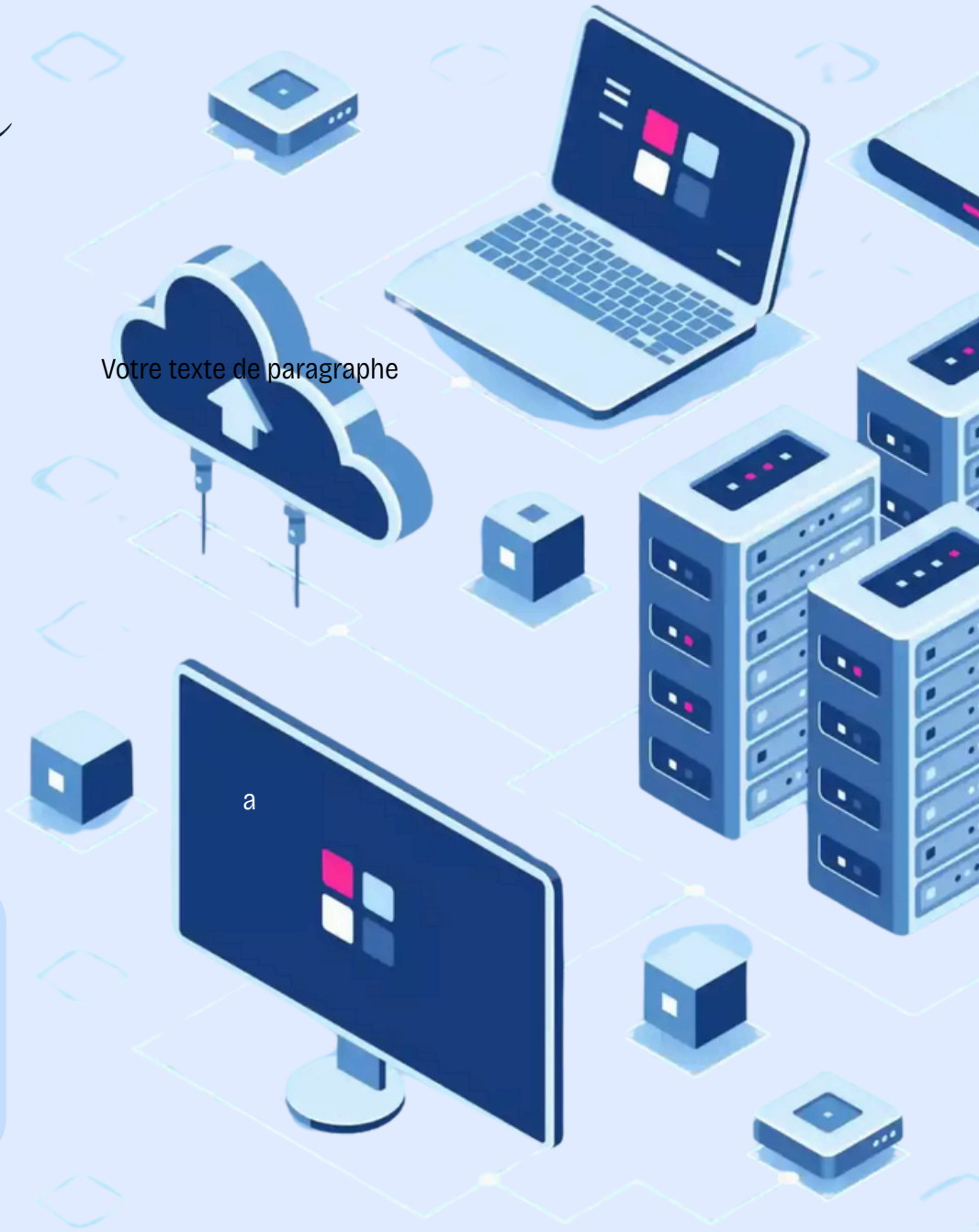
RAPPORT: SÉCURITÉ DES BASES DE DONNÉES

Implémentation RBAC

Lamouroux Pierre
de Cazes Louis



TRAVAIL EFFECTUÉ EN GROUPE DE 2, SUR LA BASE DE DONNÉES DE L'HOPITAL

Lundi 19 MAI 2025
année 2024-225

SOMMAIRE

PRÉSENTATION DE LA BASE DE DONNÉES

Pour Notre sujet nous avons choisie la base de donnée de l'hôpital qui modélise son fonctionnement, Dans laquelle nous pouvons retrouver des entités liée aux soins aux patients, aux professionels de santé et même à des données administratives

- **La base de donnée sert :**
  - Elle peut gérer les patients et leurs rendez vous
  - Suivre ce que font les médecins aux patients (diagnostique,soins)
  - Enregistrer les achats de médicaments
  - Associer les personnels (médecins, infirmiers, réceptionnistes) aux hôpitaux
  - Suivre les visites, examens, traitements et consultations

- **Les tables présentes dans la base de donnée:**
  - **HOSPITAL :** contient les informations générales des hôpitaux (nom, pays, adresse)
  - **DOCTOR :** identifie les médecins, leur spécialité et l'hôpital auquel ils sont rattachés
  - **NURSE :** enregistre les infirmiers et leur lien avec un médecin
  - **PATIENT :** stocke les informations personnelles des patients
  - **RECEPTION :** centralise les points d'accueil (avec numéro de téléphone et email)
  - **APPOINTMENT :** gère les rendez-vous médicaux (date, heure, réception)
  - **DIAGNOSIS :** suit les traitements prescrits par les médecins
  - **MEDICINE :** catalogue des médicaments (nom, prix, date d'expiration)
  - **PURCHASE :** lien entre les patients et les médicaments achetés
  - **VISIT, EXAMINE :** suivent les interactions entre patients et professionnels
  - **MEDICINE_COUNTRY:** elle sert à gèrer l'origine géographique des médicaments

ANALYSE DES RÔLES ET DES ACCÈS

Afin de répondre au problème posé on a décidé de créer différents rôles pour proposer une réponse claire et à disposition de toute les personnes ayant accès à cette base de donnée. Pour cela on c'est basé sur des exemples concrets de relation dans un hôpital.

	Rôles	Accès
Patient	• Identifié par son SSN, il peut se connecter et obtenir les information qui le concerne	• Le patient peut : • Consulter ses informations personnelles • Consulter la liste des hôpitaux • Voir ses visites passées. Grâce à RLS, il ne voit que les données qui le concernent personnellement.
Docteur	• Identifie les médecins, leurs permettants d'accéder à certaines informations concernant les patients	• Le médecin peut: • Consulter les informations des patients qu'il a examinés • Consulter les examens et diagnostics qu'il a fait • Il peut également ajouter de nouveaux diagnostics et enregistrer des visites médicales. • ils peuvent prèscrire des médicaments
infirmière	• Épaule les médecins, donc est très utile, elle peut accéder à plusieurs informations	• L'infirmier peut : • Consulter les informations de base des patients • Les diagnostic réalisées par les médecins sur les patients • Cependant, il ne peut rien modifier. Son accès est restreint par RLS aux patients pris en charge par le médecin auquel il est associé.
Secrétaire	• Permet de mettre une secrétaire, permettant de gérer les rendez-vous, elle a des rôles assez primaires	• Le/la secrétaire peut • Enregistrer les rendez-vous • Enregistrer les achats de médicaments pour les patients. • Elle peut aussi consulter la liste des médicaments disponibles et les informations des réceptions. • Elle n'a pas accès aux données médicales confidentielles, conformément aux restrictions de rôle.
Chef de service	• Ce rôle a été crée uniquement pour avoir un rôle permettant de lui attribuer des rôles	• Il accès à tout sauf aux patients

MISE EN PLACE DANS POSTGRES

Afin de mettre en place la base de données dans Postgres, nous avons dû convertir certaines choses :
VARCHAR2() → VARCHAR : On a dû changer cela car cela posez des problèmes directement dans la base de données
Nous avons également dû changer le nombre de caractère possibles dans ces VARCHAR(), en les passant à 100, pour éviter d'avoir des problèmes d'hôpital ou même ds problèmes de noms.

Nous avons donc mettre cette base de données dans postgresql, nous avons utiliser pgadmin, afin de nous faciliter la tâche.

Cela nous a pris un petit peu de temps, de tout convertir, également nous avions des erreurs, concernant les pays, leur ID et leur nom étaient inversées, ce qui nous a pris beaucoup de temps à comprendre avant de trouver cette petite heure toute bête.

Pour finir nous avons écrit les requêtes ainsi qu'ajouter les utilisateurs directement dand pgadmin 4, encore une fois pour nous faciliter la tâche, cependant, nous avons utiliser psql afin de nous connecter.

Nous avons également eu un petit problème, car nous avions créer des utilisateurs avec une majuscule, cependant, cela ne les prenais pas en charge, donc nous mis beaucoup de temps à comprendre cela et à utiliser l'utilisateur sans majuscule dans l'id de connexion.

SCÉNARIOS D'UTILISATIONS

Pour tester nos codes et nos roles créer nous avons donc décidé de créer un petit scénario détailler ci dessous:

- **Utilisateurs créer :**
  - **patient :** C'est un patient aléatoire lié à son SSN (son nom d'utilisateurs est son ssn il aura dans le scénario le 10001)
  - **docteur_pierre:** C'est un docteur, il a le rôle docteur
  - **secretaire_paul :** c'est le secrétaire qui s'occupe de l'hôpital et du patient de notre exemple
  - **infirmier_louis :** Il a le rôle de l'infirmier, il a les permissions qui lui sont nécessaires et s'occupe aussi du patient du scénario
  - **chef_personnel_phileas :** On lui a mis ce rôle car il le voulait

- **Scénario :**

Le patient avec le SSN 10001 est malade et contacte l'hôpital pour un rendez-vous. Le processus implique plusieurs intervenants avec des droits différents selon leur rôle.

1. Le patient contacte la secrétaire
   - Utilisateur : secretaire_paul (rôle : secretaire)
   - Actions autorisées :
     - INSERT dans la table APPOINTMENT
     - SELECT sur PATIENT pour identifier le patient
   - Action effectuée :

secretaire_paul planifie un rendez-vous dans la table APPOINTMENT pour le patient 10001.

2. L'infirmier prépare le dossier
   - Utilisateur : infirmier_louis (rôle : infirmier)
   - Actions autorisées :
     - SELECT sur PATIENT et DIAGNOSIS
     - INSERT dans DIAGNOSIS
   - Action effectuée :

Louis accède au dossier du patient, vérifie les données, puis prépare l'environnement pour le médecin.

3. Le médecin fait le diagnostic
   - Utilisateur : docteur_pierre (rôle : doctor)
   - Actions autorisées :
     - INSERT, SELECT sur DIAGNOSIS,MEDICINE
     - SELECT sur PATIENT
   - Action effectuée :
   - Pierre examine le patient, saisit un nouveau diagnostic
   - (table DIAGNOSIS), et prescrit un traitement dnas MEDICINE

4. Le patient récupère l'ordonnance
   - Utilisateur : patient_10001 (rôle : patient)
   - Actions autorisées :
     SELECT sur ses propres données (limité via RLS si activé)
   - Action effectuée :

Le patient consulte le diagnostic et se rend en pharmacie pour acheter les médicaments.

5. La secrétaire planifie un nouveau suivi
   - Utilisateur : secretaire_paul
   - Action :
   - Il ajoute une nouvelle entrée dans APPOINTMENT pour un suivi dans un mois.

EXPLICATIONS DE CODE

- **Création d'utilisateurs :**

Afin de créer l'utilisateur patient, nous avons utilisé ceci, le nom est bien évidemment comme expliqué au dessus, le SSN (Social Security Number), et comme mot de passe, nous avons mis un mot de passe facile à retenir, bien entendu, ce n'est pas le mot de passe le plus sécurisé

```
CREATE USER 100000001 WITH PASSWORD 'motdepasse1';
```

- **Création de Rôle :**

La création de rôle est très semblable à la création d'utilisateurs, nous mettons juste le nom du rôle à la place du nom d'utilisateurs

- **Ajout de Rôle :**

Pour ajouter le rôle également, nous pouvons voir sur l'exemple de code suivant, il faut d'abord mettre le rôle que l'on souhaite ajouter, et ensuite ajouter à qui on doit l'ajouter

```
CREATE ROLE patient NOINHERIT;
GRANT patient TO 100000001;
```

CONCLUSION- SYNTHÈSE

Ce projet a été une très bonne opportunité pour voir les différences entre les différents services de SQL disponible, et cela nous à permis de transformer une base de données en une base de données postgres, ce qui, si on utilise postgres plus tard, est très important

Pour continuer, on peut dire que ce projet était très enrichissant, que ce soit pour le côté sécurité mais également pour le côté SQL, qui nous permet d'apprendre et de réviser les bases, cela nous donne une assez bonne idée de ce que nous devrions faire si un jour on nous demande de faire cela, que ce soit dans une entreprise ou bien même dans les années à venir.

Ce projet qui en plus de cela nous permet directement de rentrer dans le vif du sujet, avec des permissions assez importantes que ce soit pour les patient ou même pour les staffs.

Bien entendu, le projet n'est pas parfait, nous sommes encore des "débutants", il y a des axes d'améliorations, avec des permissions peut être plus complète, une gestion plus complète et même une gestion des pays des hôpitaux, que l'on a pas utilisé.

- **Diagramme de présentation**