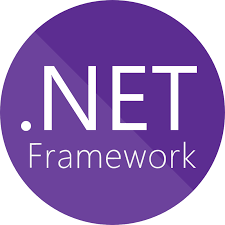
PERSISTANCE

L’application est réalisée en C#, WPF et XAML et en utilisant MVVM

Il y a deux bases de données :

* La base de données centrale qui est en Postgresql
* La base de données locale en SQLite





Le système de synchronisation est le suivant :

La synchro se fait en plusieurs étapes :

* Lorsque l’utilisateur appuie sur le bouton « Synchroniser » l’application vérifie que l’ordinateur est bien sur un réseau et qu’on arrive à pinger la machine avec le serveur
* Ajout des visites de la base locale dans la base centrale : pour chaque visite de la base locale l’appli regarde si la base centrale possède cette visite grâce à un Guid unique et si aucune occurrence n’est trouvé la visite est ajoutée dans la base centrale.
* Suppression des visites de la base locale dans la base centrale : Lors de la suppression d’une visite par un utilisateur, le champ IsDeleted de la visite correspondante dans la base de données locale est taggué à true (la ligne n’est pas vraiment supprimée) donc lors de la synchro, après avoir fait l’ajout expliqué juste avant, l’application récupère les visites taggué a true et modifie les visites de la base centrale avec celles-ci. Un système de purge a été mis en place pour contrôle la volumétrie, lorsqu’il y a plus de 500 visites déclarées comme supprimés dans la base elles sont supprimées définitivement.
* Modification des visites de la base locale dans la base centrale : Lors de la modification d’une visite par un utilisateur, le champ DateUpdate de la visite correspondante dans la base de données locale est remplie avec la date actuelle ainsi lors de la synchro l’application compare les dates de modification et la plus récente est celle qui sera pris en compte donc si une modification d’une visite de la base locale est plus récente que la même visite de la base centrale alors c’est la visite de la base locale qui est pris en compte.
* Rapatriement de la base centrale dans la base locale : Pour finir toutes les visites de la base locale sont supprimées et remplacées par les visites de la base centrale.

De plus à chaque synchro un système d’archivage se lance, ainsi si un problème a lieu la base de données aura été copié avant dans un autre dossier et pourra être réutilisé.

Pour les tests :

Notes temporaire :

Tagger une ligne quand elle est modifié ou alors rejouer toutes les requêtes

Tagger les données supprimer, elle reste en base mais elle ne s'affiche pas sur l'ihm grâce aux tags puis gérer suppression définitive (en parler)

Versionning des visites

On peut mettre un système de purge pour contrôler la volumétrie pour supprimer les lignes de plus

Faire une clé pas auto génér

Vu que c'est une petite base donc je reprends tout parce qu'on estime que même avec 10 commerciaux (continuer)....si il y avait une volumétrie importante une autre méthode serait utiliser (expliquer) importer que le delta ce qui implique de savoir quel modif a été faite