Nom du prototype : Mariadb

Par :

# Étape 1 : Faire la documentation : ​

* Faire des wireframes si nécessaire​
* Aller chercher les dessins d’architecture de la solution, de ce que vous voulez mettre en place (si disponible) ​
* Comment les interfaces sont reliées? ​**Aller cherchez les bouts de codes pertinents dans la documentation.**
* Aller chercher la documentation et les tutoriels sur vos produits.  ​

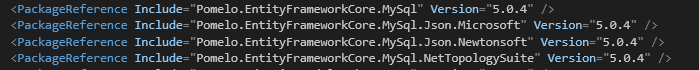
Le résultat voulu : On aimerait utiliser MariaDB aulieu de MS SQL avec backup automatisé.

Environnement: Netcore avec la librairie Pomelo.EntityFramework.MySql, MariaDB et HeidiSQL

Documentation :

* <https://github.com/PomeloFoundation/Pomelo.EntityFrameworkCore.MySql>
* <https://blog.georgekosmidis.net/using-mariadb-in-an-aspnet-core-api-with-entity-framework-core.html>
* <https://medium.com/@BMatt92656920/building-a-web-application-with-asp-net-core-mvc-entity-framework-core-mariadb-bootstrap-a2bf0927d20e>
* <https://sqlbackupandftp.com/blog/mariadb-backup-and-restore-overview#:~:text=Create%20a%20MariaDB%20database%20backup,just%20press%20%E2%80%9CRun%20Now%E2%80%9D>.

L’ajout des packages nécessaires pour l’utilisation de MariaDB



Connecter le contexte des catalogues au serveur MariaDB

# Étape 2 : Planifier

* Quels sont les fonctions que je vais tester. Le plus intéressant pour un prototype est d’utiliser la/les fonctions la plus complexe.
* Est-ce que vous devez faire attention à certains points?
* Donnez-vous un temps limite!​

## Fonctions à tester, hypothèses, contraintes :

Fonctions à tester.

1. J’ai besoin d’une application de base utilisant une autre base de donnée que MariaDB
2. J’ai besoin d’un serveur MariaDB
3. J’ai besoin d’un système de Backup automatique (SQLBackupAndFTP dans mon cas)
4. J’ai besoin d’un seed de base qui me permettra de tester ma base de données
5. Fonctions complexes : dans le contexte de eShopOnWeb, je vais tester les différentes fonctionnalités de l’application tel qu’Ajouter, Supprimer un article au panier. Tester l’interaction de l’application et de la base de données.
6. Je vais tester le système de backup automatique avec MariaDB, en automatisant un backup à chaque heure.
7. Je vais voir s’il y a des différences avec l’utilisation de MariaDB aulieu de MS SQL.

Hypothèse :

Je suppose que MariaDB, étant un système de base de données relationnel plus récent que SQL, pourrait être une option plus envisageable considérant qu’elle possède d’avantage fonctionnalités et une vitesse considérablement plus élevée que celle de MS SQL.

Craintes : Je n’ai aucune crainte autre que le projet eShop pourrait reconnaitre des problèmes d’incompatibilité puisqu’il a été construit autour de MS SQL.

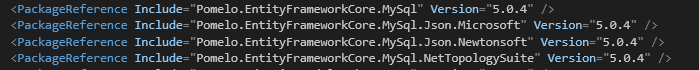
## Durée limite

Je me donne 3h00 pour faire fonctionner MariaDB avec un backup automatisé.

# Étape 3 : Expérimentez

Notez ce que vous faites

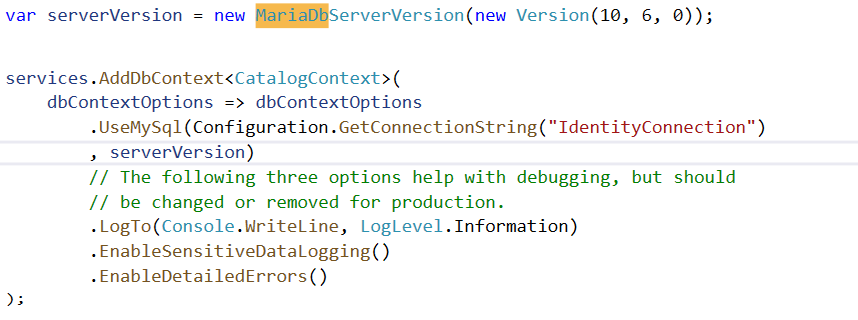
J’installe les packages nécessaires relier le projet à une BD MariaDB

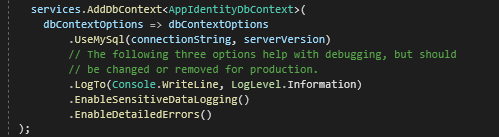


Je crée la connection string de la base de donnée

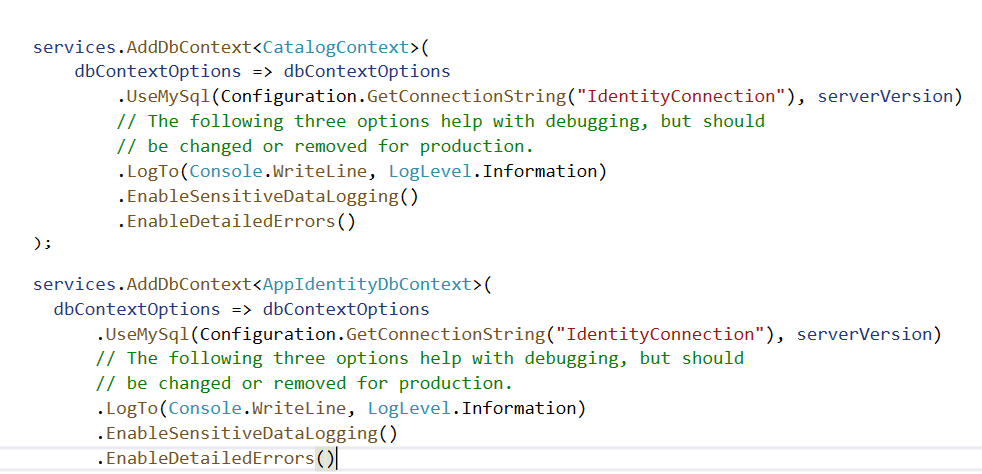
 ​

J’ajoute le DbContexte des catalogues et de identity en spécifiant l’utilisation de MySql avec une version de MariaDb compatible.

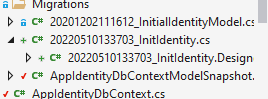




Je précise partout que je vais utiliser MySql (MariaDB) aulieu de SqlServer



Je supprime les migrations existantes puisqu’ils n’ont pas été fait avec MySQL(MariaDB)

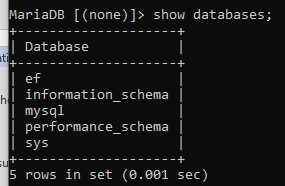


Je re crée une migration Initiale pour le CatalogContext et met à jour la DB

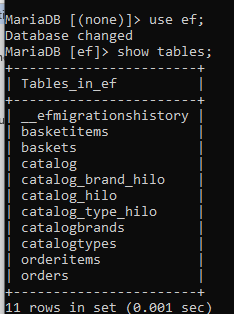




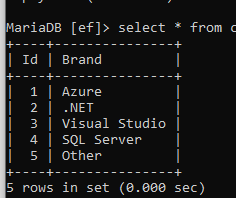
La DB est apparu, ça l’a fonctionner.

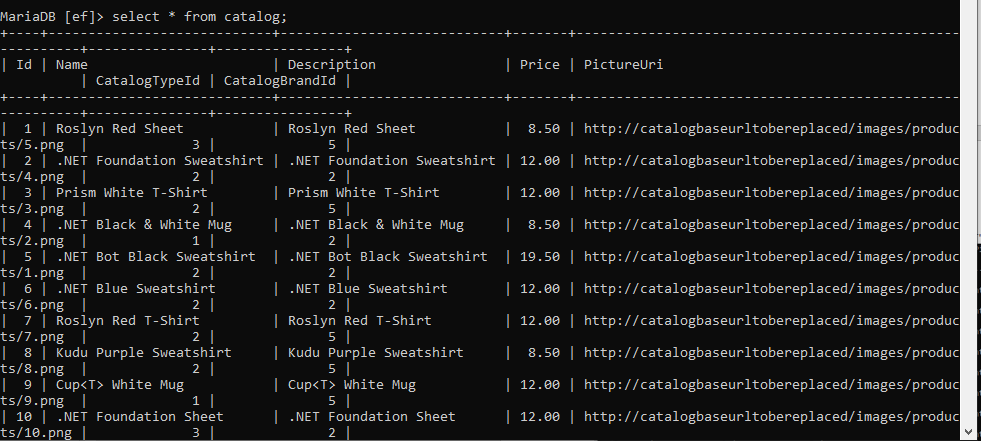


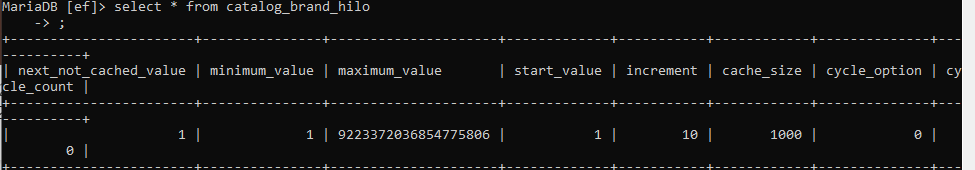
Les tables sont présentes



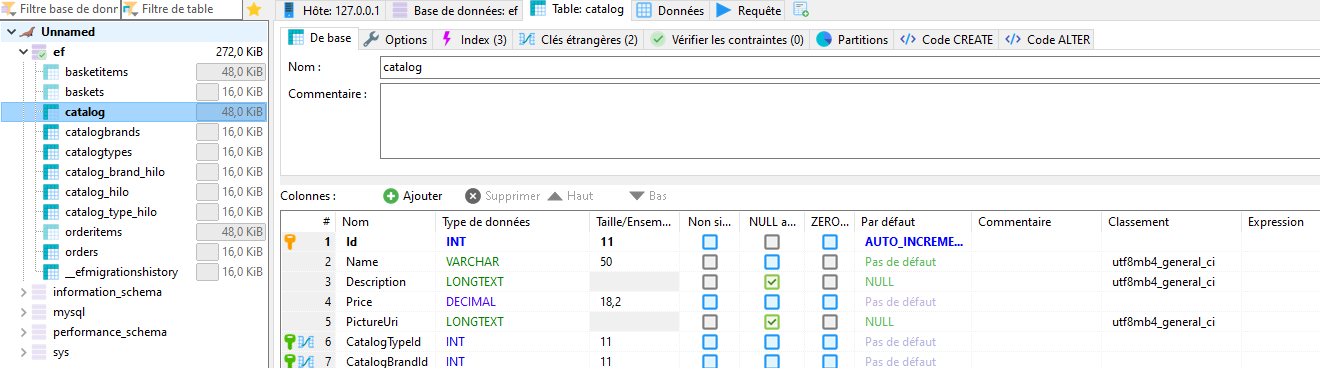
Les tables sont remplies





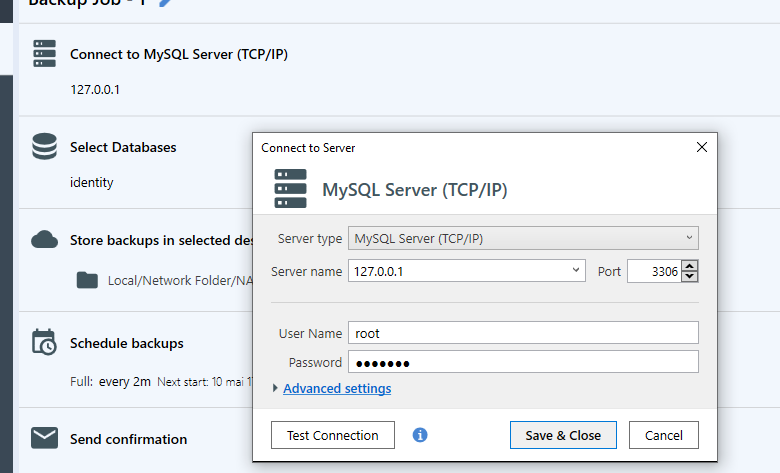


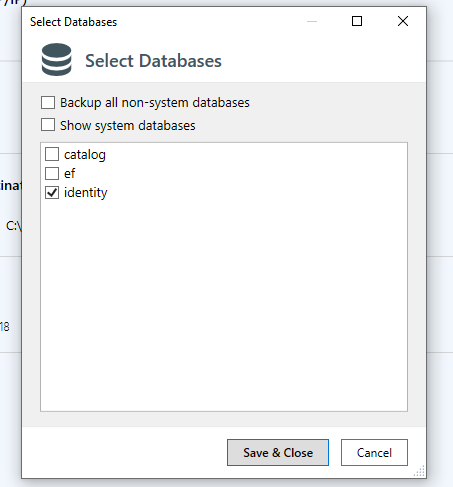
Toutes les tables du catalogue sont remplis.

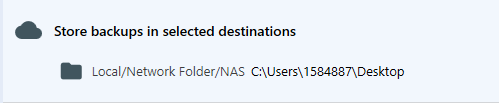


J’ajoute le système de backup automatisé à l’aide de SQLBackupAndFTP :

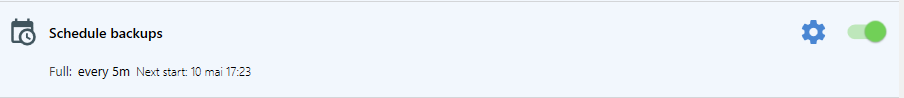
Je me connecte à ma base de donnée :



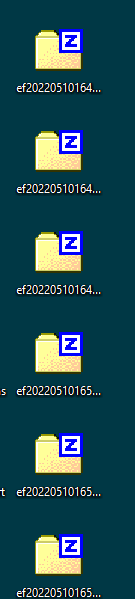
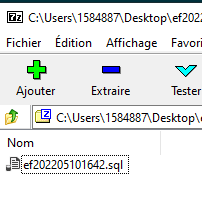
Je selectionne ma base de donnée : 

Je définie l’emplacement du fichier backup :

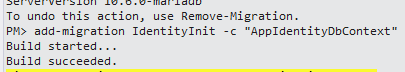
Je définie le temps d’intervalle auquel mes backups seront créée :



Les backups se crée avec un intervalle de 5 minutes, par example.

Je crée une migration pour le contexte d’identity et Update la base de donnée



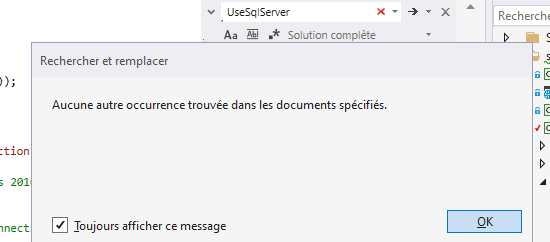


Pour le AppIdentityContext, la base de donnée n’a pas voulu se mettre à jour, c’est probablement qu’elle utilise encore SqlServer aulieu de MySQL



Cela fait maintenant 2h40 que je suis en train d’implémenter mariaDB, il ne me reste donc que 20 minutes pour régler cette erreur de lecture.

Je recherche donc si j’ai oublié de préciser l’utilisation de MySQL pour le context Identity



Il me semble avoir tout changé, je ne sais donc pas pourquoi la requête SQL n’a pas été traduite en requête lisible par MariaDB.

# Étape 4 : Documenter les résultats.​

* Est-ce que je suis confortable à continuer avec cette solution?​
  + Non, bien que MariaDB me semblait être une approche assez simple et efficace, elle m’a causé plusieurs problèmes à l’implémenter sur l’application WEB, déjà construite, qui est déjà relié à un autre système de base de données.

J’ai réussi à migrer le CatalogDbContext mais je n’ai pas eu autant de succès pour l’IdentityDbContext.

* Est-ce que la solution comme tester va répondre aux besoins de mon client?​
* Est-ce que c’est ce que je m’attendais? Est-ce que mes hypothèses étaient les bonnes. ​