Une image contenant art, croquis, dessin, conception

Description générée automatiquement

|  |
| --- |
| Documentation technique Paramétrage et sécurisation de SQL serveur Standard |

Table des matières

[1 Analyse préliminaire 4](#_Toc198297606)

[1.1 MCD et MLD 4](#_Toc198297607)

[1.2 Backup scheduler sur Express 5](#_Toc198297608)

[1.2.1 Création de script 5](#_Toc198297609)

[1.3 Express VS Standard 6](#_Toc198297610)

[1.4 6](#_Toc198297611)

[1.5 Stratégie de test 6](#_Toc198297612)

[1.6 Risques techniques 7](#_Toc198297613)

[2 Réalisation 7](#_Toc198297614)

[2.1 Dossier de réalisation 7](#_Toc198297615)

[2.2 Description des tests effectués 8](#_Toc198297616)

[2.2.1 Backup DB 8](#_Toc198297617)

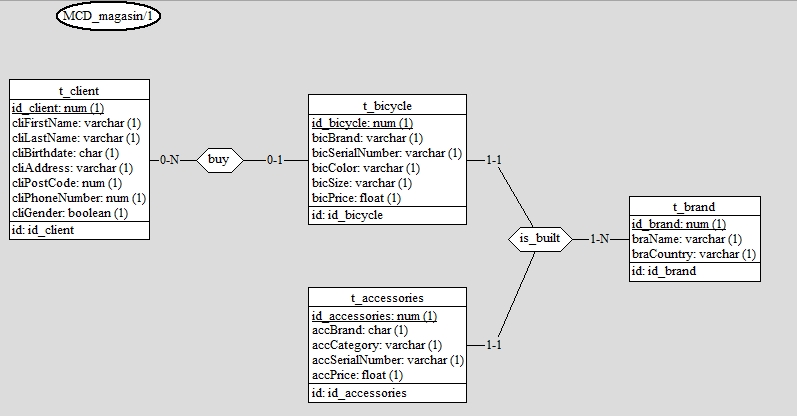
[2.3 Erreurs restantes 9](#_Toc198297618)

# Analyse préliminaire

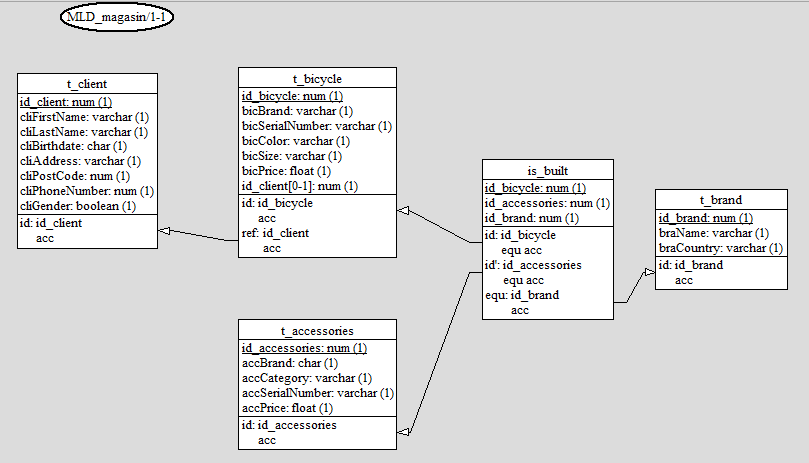
## MCD et MLD

DB magasin :

Model conceptuel des données :



Model logique des données :



Requête SQL pour générer la table :

DB-main à générer un premier code pour créer les tables mais cela n’était pas adapté pour Microsoft SQL Server pour cela que j’ai utilisé chat-gpt pour le convertir et que j’ai corrigé les quelques erreurs qui restait.

Script disponible sur mon repos git-hub (<https://github.com/luca7274/TPI>).

## Backup scheduler sur Express

### Création de script

Expliquer pourquoi avoir utiliser SQLCMD et pas SMO

Pour la réalisation des diferents scrips, deux chois s’offret à moi :

SQLCMD ou SMO (SQL server management Objects). Le premier est un outil de ligne de commande qui permet d’interagir avec un server sql et le second est une biblioteque .NET qui s’apparente plus à de la programation orienté objet. J’ai opter pour SQLCMD car il n’a pas besoin d’être installer à contrario de SMO qui a une installation compliquée et sont utilisation est assé simple et va parfaitement pour le type de script que j’ai réaliser car il correspond directement au commande que l’on peut utiliser dans SSMS.

Premièrement il faut créer un script permettant de réaliser la backup (disponible sur github).

-adapter le nom du server avec son instance.

-choisir la DB à backup.

-choisir ou la back-up va être stocker

Une image contenant texte, Police, capture d’écran, ligne

Description générée automatiquement

## Express VS Standard

La version express ne vient pas avec SQL agent et ne possède pas de de dossier maintenance afin de planifier l’exécution d’une backup par exemple.

|  |  |  |
| --- | --- | --- |
| Critère de comparaison | SQL Server Express 2022 | SQL Server Standard 2022 |
| Limites Hardware | - CPU : 1 socket / 4 cœurs  - RAM : 1410 Mo | - CPU : 4 sockets / 24 cœurs  - RAM : 128 Go |
| Taille maximale d’une base de données relationnelle | DB de maximum 10 Go | DB de maximum 524 Po |
| Fonctionnalité | SQL Agent : non  SQL Profiler : non  SQLCMD : oui  Database Tuning Advisor: non | SQL Agent : non  SQL Profiler : non  SQLCMD : oui  Database Tuning Advisor : Oui |

## 

*Par exemple :*

* *Multimédia: carte de site, maquettes papier, story board préliminaire, …*
* *Bases de données: interfaces graphiques, modèle conceptuel.*
* *Programmation: interfaces graphiques, maquettes, analyse fonctionnelle…*
* *…*

## Stratégie de test

*Décrire la stratégie globale de test:*

* *types de des tests et ordre dans lequel ils seront effectués.*
* *les moyens à mettre en œuvre.*
* *couverture des tests (tests exhaustifs ou non, si non, pourquoi ?).*
* *données de test à prévoir (données réelles ?).*
* *les testeurs extérieurs éventuels.*

## Risques techniques

* *risques techniques (complexité, manque de compétences, …).*

*Décrire aussi quelles solutions ont été appliquées pour réduire les risques (priorités, formation, actions, …).*

# Réalisation

## Performance dashboard

SSMS propose nativement une solution pour génère des graphiques de performance.

Pour ce faire, il faut faire clique droite sur l’instance puis report et après standard reports.

*Décrire la réalisation "physique" de votre projet*

* *les répertoires où le logiciel est installé*
* *la liste de tous les fichiers et une rapide description de leur contenu (des noms qui parlent !)*
* *les versions des systèmes d'exploitation et des outils logiciels*
* *la description exacte du matériel*
* *le numéro de version de votre produit !*
* *programmation et scripts: librairies externes, dictionnaire des données, reconstruction du logiciel - cible à partir des sources.*

*NOTE : Evitez d’inclure les listings des sources, à moins que vous ne désiriez en expliquer une partie vous paraissant importante. Dans ce cas n’incluez que cette partie…*

**Faire une sorte de bibliographie avec tous les termes techniques  
 (ex : SSMS : outil de gestion de base de données)**

## Description des tests effectués

### Backup DB

Pour vérifier que le script de back-up fonctionne parfaitement, il va falloir vérifier plusieurs étapes de son fonctionnement :

1. Le script de back-up :

Pour ce faire, après avoir établi directement dans le script à partir de quelle DB (la DB doit contenir des données pour que le test soit correctement réalisé) et dans quel dossier nous voulons effectuer la back-up.

Il va falloir ensuite exécuter le script, puis se rendre dans le dossier de backup pour vérifier qu’il y a bien un nouveau fichier en .bak avec la date de création dans le nom et qui possède une taille suffisamment grande afin que nous puissions déterminer que les données se soient bien exportées.

1. L’exécution hebdomadaire :

Premièrement, il va falloir lancer la tâche hebdomadaire manuellement afin de constater s’il arrive bien à l’exécuter sans problème (ex : ouvre simplement le script dans un bloc-notes au lieu de l’exécuter) et vérifier que le fichier existe bien (voir point 1).

Ensuite, paramétrer la tâche pour qu’elle se lance automatiquement à une heure proche (ex : 2 minutes après l’heure actuelle) et vérifier que le fichier existe bien (voir point 1).

1. La restauration des données vers une autre DB :

Au préalable, mettez deux fichiers de back-up dans votre dossier. Le premier avec une date plus ancienne et moins de données et un autre avec plus récent et avec plus de données.

À l’aide d’un second script et après avoir défini directement dans le script, à partir de quel dossier de backup et vers quelle DB restaurer les données (pour la phase de test, il est important que la DB soit vide). Il va falloir exécuter le script puis une fois celui-ci terminé, il faut se connecter à la DB via SSMS et vérifier que les données les plus récentes se soient bien exportées vers la DB.

*Pour chaque partie testée de votre projet, il faut décrire:*

* *les conditions exactes de chaque test*
* *les preuves de test (papier ou fichier)*
* *tests sans preuve: fournir au moins une description*

## Erreurs restantes

*S'il reste encore des erreurs:*

* *Description détaillée*
* *Conséquences sur l'utilisation du produit*
* *Actions envisagées ou possibles*