Projet ADO.Net

# Présentation

Le client « Grand Hôtel » souhaite que vous développiez une application pour la gestion de son établissement.

Il possède déjà une base de données SQL Server, dont il vous fournit une sauvegarde, ainsi que le modèle physique de données :



Ci-dessous la description des tables et règles de gestions principales :

|  |  |
| --- | --- |
| **Table** | **Description** |
| Client | Un client ne peut avoir qu’une seule adresse, mais plusieurs numéros de téléphones et plusieurs adresses email. |
| Adresse | Adresse unique du client. Certains clients n’ont pas d’adresse.  Lorsqu’elle existe l’adresse est identifiée par l’id du client. |
| Téléphone | Donne la liste des N° de tel. De chaque client. Le champ CodeType vaut M pour un N° de mobile et F pour un N° fixe.  Pro indique s’il s’agit d’un N° professionnel |
| Facture | La date de paiement de la facture est nulle tant que la facture n’est pas payée |
| LigneFacture | Détaille les prix et quantités des services facturés (chambres, petits déjeuner…etc.) pour chaque facture. Une ligne est identifiée par l’id de la facture et un numéro. |
| Réservation | Contient une ligne pour chaque jour et chambre occupée.  Le champ HeureArrivee indique l’heure d’arrivée approximative prévue par le client. La précision est à l’heure près.  Le champ Travail indique si le client réserve pour son travail  Une chambre peut être réservée par plusieurs personnes (ex : un couple avec enfants) à la même date. Mais la réservation correspondante ne peut être reliée qu’à un seul client (qui représente les personnes). |
| Calendrier | Contient tous les jours de chaque année d’exploitation de l’hôtel. Elle est utile pour déterminer facilement le nombre de jours dans chaque mois, ainsi que les jours durant lesquels certaines chambres ne sont pas occupées. |
| Tarif | Contient l’évolution des tarifs des différentes prestations (chambres de types 1 et 2, petit déjeuner…).  Les tarifs sont révisés au 1er janvier de chaque année. |
| TarifChambre | Permet d’associer les chambres aux tarifs. |

Créer une application console nommée GrandHotel avec les modules fonctionnels décrits ci-dessous et utilisant :

* Le projet TConsole pour créer des pages et menus de navigation, et afficher des listes sous forme de tableaux
* Des SqlCommand ou Entity Framework 6 dans l’approche Code First

# Fonctionnalités de l’application

## Gestion des clients

Ce module doit permettre de :

* Afficher la liste des clients
* Afficher les coordonnées d’un client (adresse postale, listes de ses N° de téléphone et adresses email), en ne demandant qu’une fois l’identifiant du client
* Saisir un nouveau client et son adresse
* Ajouter un N° de téléphone ou une adresse email à un client
* Supprimer un client et ses informations liées (adresse, téléphone, email) si ce client n’est associé à aucune facture ni occupation de chambre.
* Exporter la liste des clients au format xml

## Gestion des factures

Ce module doit permettre de :

* Afficher la liste des factures d’un client à partir d’une date donnée (par défaut sur un an glissant)
* Afficher les lignes d’une facture identifiée par son Id
* Saisir une facture
* Saisir les lignes de facture d’une facture donnée
* Mettre à jour la date et le mode de paiement d’une facture
* Exporter les factures d’un client donné au format xml, sans le détail des lignes, mais avec le montant total de la facture.

## Résultats de l’hôtel

Ce module doit permettre d’afficher :

* Les clients pour lesquels on n’a pas de numéro de portable (id, nom)
* Le taux moyen de réservation de l’hôtel par mois-année (2015-01, 2015-02…), c'est à dire la moyenne sur les chambres du ratio (nombre de jours de réservation dans le mois / nombre de jours du mois)
* Le nombre quotidien moyen de clients présents dans l’hôtel pour chaque mois de l’année 2017, en tenant compte du nombre de personnes dans les chambres
* Le chiffre d’affaire de l’hôtel par trimestre de chaque année
* Le nombre de clients dans chaque tranche de 1000 € de chiffre d’affaire total généré. La première tranche est < 5000 €, et la dernière >= 8000 €

# Livraison

Vous devez fournir :

* L’ensemble des sources sur GitHub
* Les fichiers binaires permettant d’exécuter directement l’application (exe, dll, fichier de configuration)
* Un fichier readme.docx fournissant toutes les explications que vous jugerez nécessaires pour lancer et utiliser l’application.

# Critères d’évaluation

|  |  |
| --- | --- |
| **Livraison** | **Fonctionnel et ergonomie** |
| L'ensemble des éléments demandés est fourni (source, exe, fichier de config, doc)  Le code compile et s'exécute correctement avec la base fournie  Quelques tests unitaires sur les méthodes de DAL sont fournis | Les fonctionnalités implémentées sont stables (pas de bugs en exécution)  L'ergonomie des pages de l’appli est simple et compréhensible  Nombre de fonctionnalités implémentées |
| **Architecture technique** | **Qualité du code** |
| Les principes d'architecture sont respectés (séparation UI et DAL) | La sémantique est pertinente Les conventions de codage sont respectées  Le code est commenté et facile à lire |

**Important : Privilégier la qualité à la quantité**

Ne pas chercher à implémenter l’ensemble des fonctionnalités, mais plutôt à faire du code de bonne qualité, et une application stable et bien testée.