
La startup Cooking

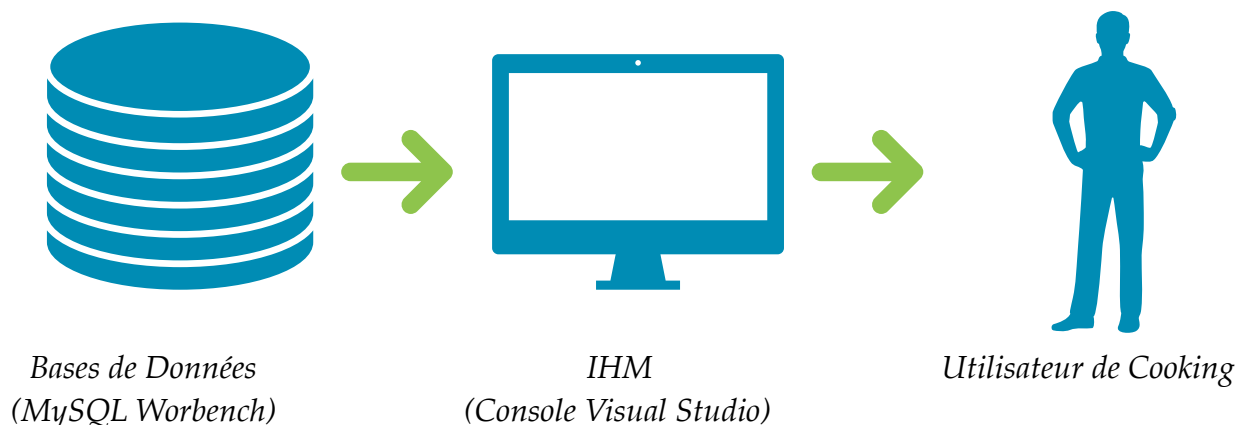
Bases de Données et Interopérabilité

Alexandra Nebout & Stephan Adjarian - 13/05/2020



Introduction

Dans le cadre du lancement du site internet « Notre petite Cuisine », nous avons dû gérer intégralement la gestion et les données de la StartUp Cooking. Nous avons utilisé MySQL Workbench, logiciel de gestion et d'administration de bases de données MySQL et VisualStudio. Ce projet consistait donc à connecter une base de données à un logiciel de code en C#.



Chaine de traitement de l'information

Notre application permet de s'inscrire en tant que client ou créateur de recettes, de passer commande, de publier de nouvelles recettes, gérer les flux d'argent et les stocks de Cooking. Notre projet comporte également une partie de gestion de l'entreprise, permettant d'accéder aux statistiques de l'entreprise ou encore de supprimer des employées (cuisiniers) ou des créateurs de recettes.

Ce rapport décrit la majorité des fonctions de notre application, avec quelques illustrations de son bon fonctionnement. Nous y détaillons des explications ne figurant pas en commentaires dans le code.

Remarque importante : Nous avons fait le choix de ne pas créer de classes en c# associées à nos tables car nous avons considéré que se projet a pour but de manipuler des données MySQL via un code C#, ce n'est donc pas un problème de programmation orienté objet.

Quelques mots sur nos Clients

Un Client doit toujours s'identifier avant d'effectuer une action sur notre application. Si il n'existe pas dans notre base de données, nous lui proposons automatiquement de créer un compte (*Creer_Un_Compte_Client*), ensuite on lui propose de passer commande immédiatement (en suivant la chaine d'appel suivante).

Les sous-programmes sont appelés dans cet ordre : Choix_Repas —> Payer_une_Commande —> *automatiquement* :

Inserer_Table_Commande —> *Inserer_Table_Ayant* —> *Incrementer_Nb_Commande*

—> *Modifier_Solde_Client* —> (Si nombre commandes >= 10 : *Modifier_Prix_Commande*) —>

(Si nombre commandes >= 50 : *Modifier_Prix_Commande* —> *Modif_Solde_Cdr*)

—> *Decrementer_Stock_Ingredient* —> *Modifier_Date_Derniere_Utilisation* ->

Modifier_solde_cuisinier -> *Modifier_solde_cdr*.

L'entité Commande comporte 3 attributs : une référence (clé primaire), une date et l'ID du client passant commande (clé étrangère). Cette table permet de sauvegarder toute les commandes passées afin de connaître l'activité de l'entreprise sur une période donnée, par exemple.

L'entité Ayant comporte 2 attributs : la référence de la commande en cours et le nom d'une recette commandée; ces deux attributs forment sa clé primaire. Cette table permet d'associer une recette à une commande afin de faire une étude statistique des recettes les plus commandées, par exemple.

Quelques mots sur nos Créateurs de Recettes

Créateur de Recette (Cdr) est automatiquement un *Client* de Cooking. Les 2 entités sont reliées par une relation de cardinalité 1:1.

A tout moment, un Cdr peut connaître son solde, consulter les recettes qu'il a créées, le nombre de commandes passées pour celles-ci ou encore créer une nouvelle recette.

L'entité Recette est relié à *Cdr*, à *Commande* et à *Produit*. Ces deux dernières relations étant de cardinalité n:n, les associations deviennent des tables (*ayant* et *composée_de*).

Le *Cdr* peut créer des nouvelles recettes. Pour cela il se rend dans le sous-programme *Cdr* ('E' dans le menu du main). Ensuite, il s'identifie, si son id existe alors on arrive dans le sous-programme *Creer_Une_Recette* puis *Decrire_Composition_Recette*. Si son ID n'existe pas alors on lui propose directement de créer un compte de Cdc (*Creer_compte_cdr*) puis il est directement redirigé vers la création d'une recette.

Quelques mots sur les saisies

Toutes les saisies sont vérifiées dans le code et on ne peut pas entrer n'importe quoi. Ceci permet de gérer notre application de façon professionnelle. Ces vérifications se font par exemple par les sous-programmes : *Recuperer_les_Objets* et *Test_Entree_Valide*. Ces sous-programmes sont codés de façon à ce qu'il soit utilisable pour tester la validité d'un nom de recette ou d'un ID de *Cdr* (on envoie en paramètres l'attribut et la table en question), par exemple.

Quelques mots sur la partie Gestionnaires

Dans les références de chaque commande, on inclut systématiquement le rang de la commande que le client est en train de passer. La référence d'une commande est composée de l'ID du client, de la date du jour et du rang de la commande. Ceci permet au gestionnaire de retrouver facilement quelles recettes ont été commandées récemment.

Dans le modèle logique des données ci-joint, on lit facilement que pour supprimer une recette, il faut d'abord supprimer le tuple de la table *composee_de* et ayant correspondant à cette *recette*, afin qu'il n'y ait pas de problème de consistance dans la base de données. De même lorsqu'on souhaite supprimer un cuisinier.

La difficulté de cette partie réside surtout dans la recherche des bonnes requêtes SQL.

