**Par Sandya Soma Soundaram et Fatou Sow**

**PROJET TUTORÉ S3**

**Description de l’analyse, la conception et l’implémentation de la base de données**

Table des matières

[Introduction 3](#_Toc499503084)

[Diagramme de cas d’utilisation 3](#_Toc499503085)

[Dictionnaire des données 4](#_Toc499503086)

[Modèle conceptuel de données 4](#_Toc499503087)

[Schéma relationnel de données 4](#_Toc499503088)

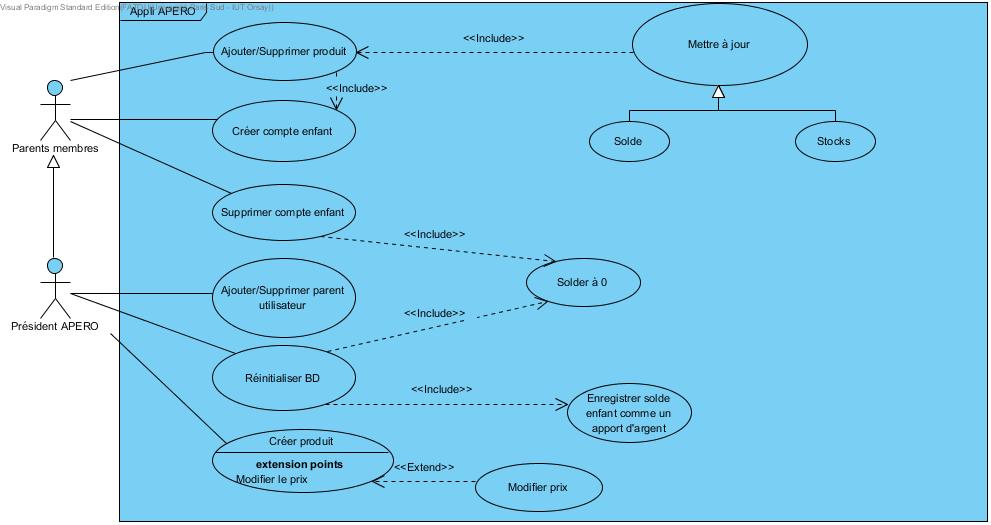
[Conclusion 4](#_Toc499503089)

## Introduction

L’application à concevoir pour APERO doit permettre aux parents de l’association de s’occuper des goûters de leurs enfants après leurs entraînements de rugby. Elle remplacera la gestion manuelle peu pratique car elle engendrait une perte de temps et un problème d’organisation. Grâce à la future application, les utilisateurs seront en mesure de créer un compte pour chaque enfant afin de contrôler son solde en fonction de ses achats mais aussi de gérer le stock de produits nécessaires aux goûters (ajout, suppression et notifications lorsque les produits sont sur le point de manquer). Tout ceci sera sécurisé et l’application sera facile d’utilisation et accueillante.

Les acteurs de l’application seront les parents membres de l’association qui seront en charge de la distribution de goûters ainsi que le président de l’association.

## Diagramme de cas d’utilisation



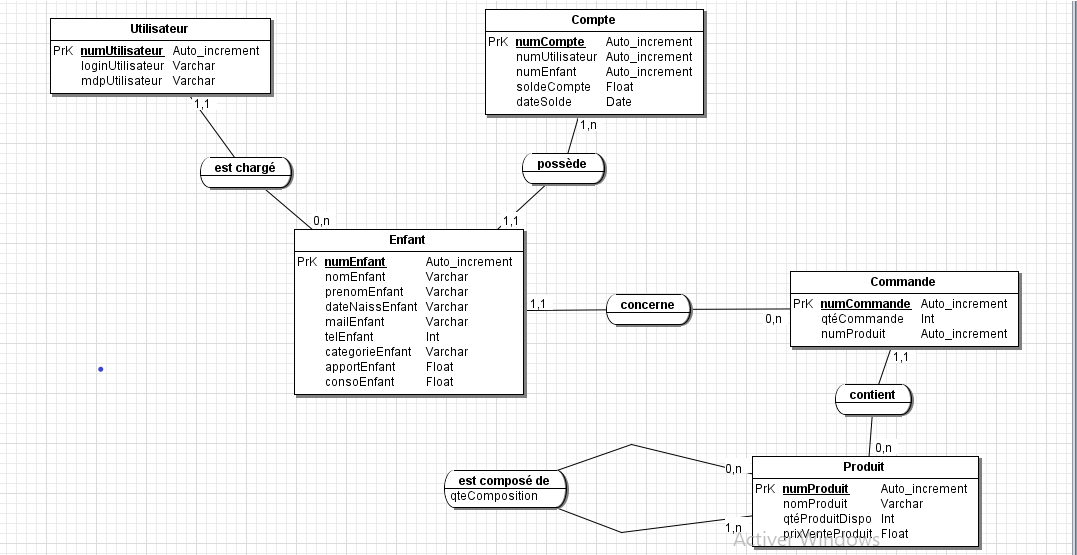
Les parents membres chargés de la distribution des goûters peuvent créer un compte pour un enfant. Ils peuvent aussi ajouter les produits consommés ou les supprimer si nécessaire. Cette action entraîne une mise à jour de la base de données, au niveau du solde de l’enfant et du stock de produits. Par ailleurs, l’ajout ou la suppression des produits n’est possible que si un compte est créé. En outre, ils sont en mesure de supprimer le compte d’un enfant mais doivent au préalable mettre son solde à zéro.

Le président d’APERO qui est le gestionnaire, est une spécialisation d’un parent membre. Ainsi, il peut exécuter les mêmes actions d’un membre mais de plus, peut ajouter ou supprimer des parents utilisateurs. Il est aussi capable de réinitialiser la base de données après avoir mis les soldes à zéro et enregistré le solde restant comme un apport d’argent. Il peut aussi créer un produit et éventuellement modifier son prix

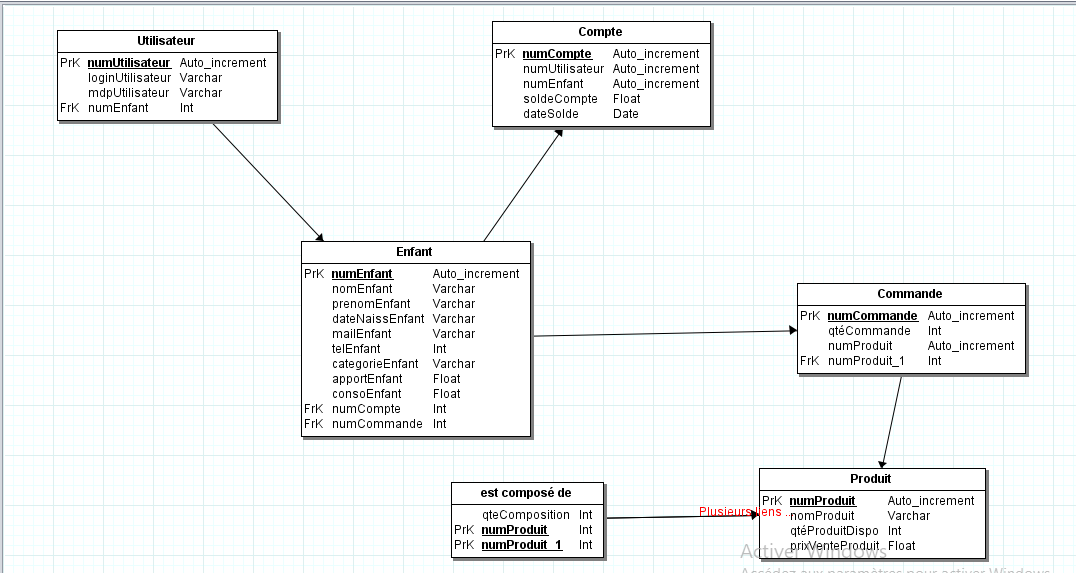
## Dictionnaire des données

|  |  |  |
| --- | --- | --- |
| **NOM RUBRIQUE** | **TYPE** | **DESCRIPTION** |
| numUtilisateur | A | Numéro de l’utilisateur inscrit |
| loginUtilisateur | A | Login propre à un utilisateur |
| mdpUtilisateur | A | Mot de passe propre à un utilisateur |
| numEnfant | A | Numéro de l’enfant inscrit |
| nomEnfant | A | Nom de l’enfant inscrit |
| prenomEnfant | A | Prénom de l’enfant inscrit |
| dateNaissEnfant | A | Date de naissance de l’enfant inscrit |
| mailEnfant | A | Adresse mail de l’enfant inscrit |
| telEnfant | A | Numéro de téléphone d’un des parents de l’enfant |
| categorieEnfant | A | Catégorie de l’enfant selon son âge |
| numCompte | A | Nom de compte de l’enfant inscrit |
| soldeCompte | C | = apportEnfant - consoEnfant |
| consoEnfant | A | Dépenses de l’enfant pour chaque goûter |
| dateSolde | A | Dates du dernier solde |
| apportEnfant | A | Ajout d’argent pour le solde de l’enfant |
| numProduit | A | Numéro du produit |
| nomProduit | A | Nom du produit |
| prixVenteProduit | A | Prix de vente du produit |
| numCommande | A | Numéro de commande |
| qtéCommande | A | Quantité demandée à la commande |
| qtéProduitDispo | C | =qtéProduitDipso - qtéCommande |
| qteComposition | A | Proportion nécessaire à la réalisaton du produit |

## Modèle conceptuel de données



## Schéma relationnel de données



## 

## Conclusion