Projet informatique intégré

# Présentation du domaine d’application

Dans l’idée de créer un bar à cafés, une borne sera utilisée pour permettre aux clients d’effectuer leurs commandes. Notre application aura comme fonctionnalités les suivantes :

* la commande de cafés (avec des ajouts possibles, nommés *toppings,* pour permettre un choix un peu plus personnel),
* un système de carte de fidélité reprenant certains avantages (permettant d’avoir un formulaire d’ajout/modification/suppression du client),
* un système de gestion des stocks pour le café, les toppings ainsi que les différents accompagnements,



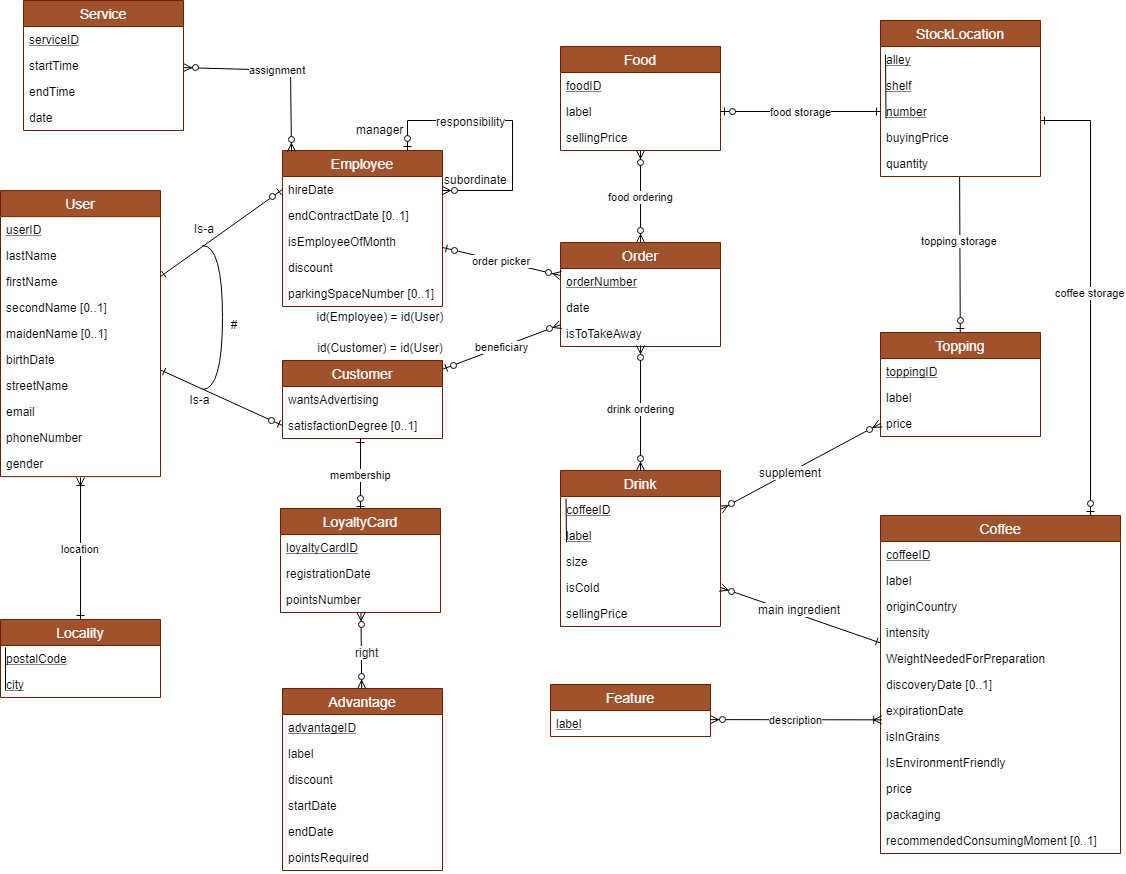
* une gestion des employés.

Les commandes sont sauvegardées dans la base de données pour permettre aux clients de repasser une commande faite précédemment. Si le client décide de supprimer son compte, ses commandes sont tout de même gardées pour pouvoir effectuer des statistiques chez le manager.

Lorsque le stock de certains produits est vide, ceux-ci ne sont plus disponibles comme choix de commande ; ce système permet de faire le calcul de prix en tenant compte des avantages de la carte de fidélité. Le système de gestion des stocks permet d’ajouter des nouveaux cafés, d’en supprimer et d’en mettre à jour.

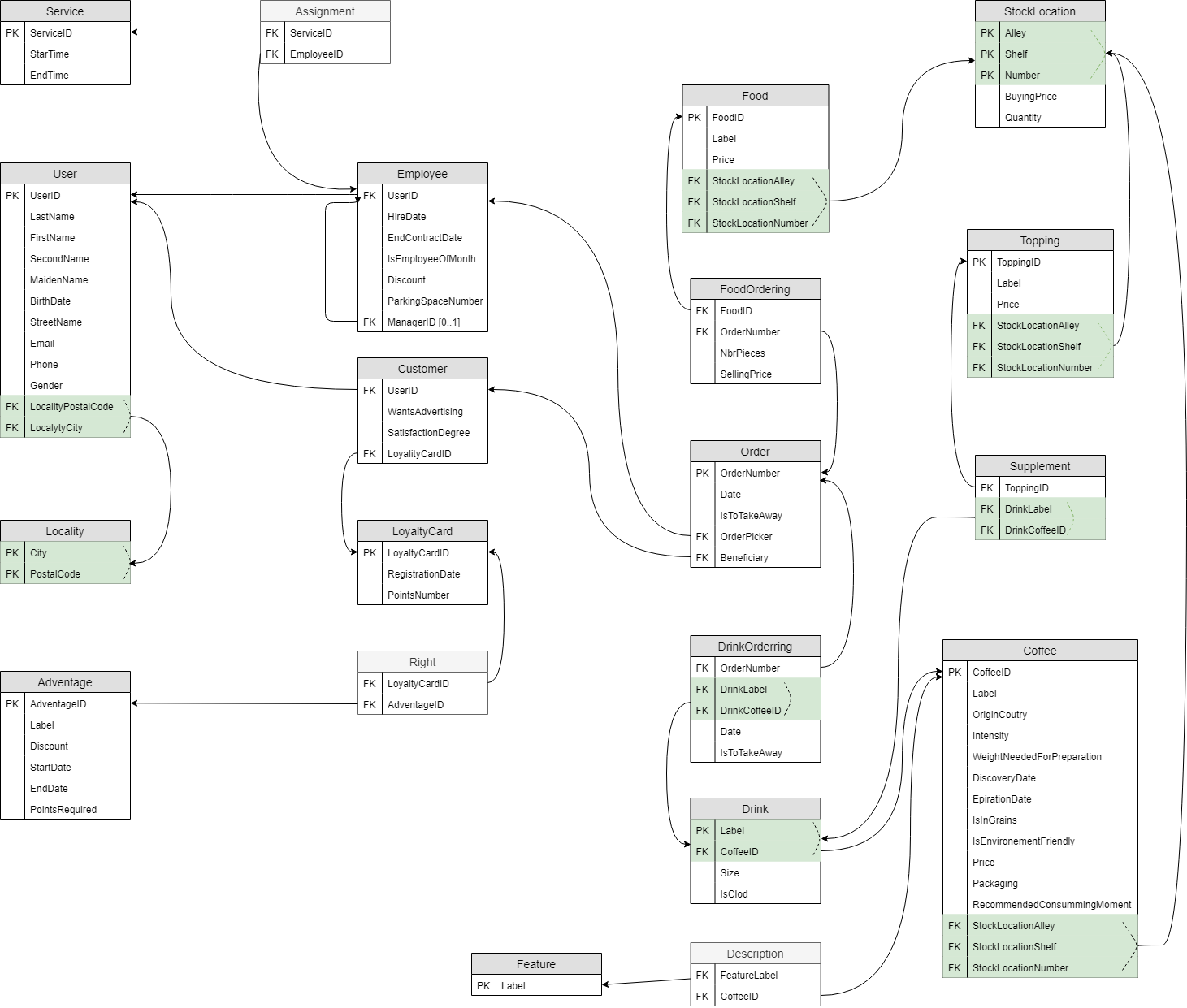
Les employés peuvent gérer les stocks, les clients, ainsi que les horaires de leurs subalternes s’ils sont manager. Un formulaire de gestion des employés sera créé pour les managers afin d’effectuer les tâches de création, de suppression et de modification des employés.

# Schéma entités-associations



# Schéma entités-relations

# Schéma logique des tables



# Spécification des tables

## User

Superclasse de client et employé.

|  |  |  |  |
| --- | --- | --- | --- |
| Nom | Type | Nullable ? | Description |
| UserID | Integer | / | Clé primaire de la table. |
| LastName | Varchar | / | Nom de famille de la personne. |
| FirstName | Varchar | / | Prénom de la personne. |
| SecondName | Varchar | Oui | Deuxième prénom de la personne s’il en a un. |
| MaidenName | Varchar | Oui | Nom de jeune fille de l’utilisatrice si elle veut le renseigner. |
| BirthDate | Date | / | Date de naissance de la personne. |
| StreetName | Varchar | / | Numéro et rue de la personne. |
| Email | Varchar | / | Adresse mail de la personne. |
| PhoneNumber | Integer | / | Numéro de téléphone de la personne. |
| Gender | Varchar | / | Genre de la personne. |
| LocalityPostalCode | Integer | / | Clé étrangère composée vers localité |
| LocalityCity | Varchar | / |

## Locality

|  |  |  |  |
| --- | --- | --- | --- |
| Nom | Type | Nullable ? | Description |
| PostalCode | Integer | / | Clé Code postal à 4 chiffres (on se limite à la Belgique). |
| City | Varchar | / | Nom de la ville de résidence. |

## Service

|  |  |  |  |
| --- | --- | --- | --- |
| Nom | Type | Nullable ? | Description |
| ServiceID | Integer | / | Clé primaire. Identifiant de la tranche horaire de travail. |
| StartTime | Timestamp | / | Date et heure de début du service. |
| EndTime | Timestamp | / | Date et heure de fin du service. |

## Assignment

|  |  |  |  |
| --- | --- | --- | --- |
| Nom | Type | Nullable ? | Description |
| EmployeeID | Integer | / | Clé étrangère. |
| ServiceID | Integer | / | Clé étrangère. |

## Employee

La table employé permet d’avoir un formulaire. On pourra ajouter un nouvel employé dans la base de données lorsque celui-ci commence à travailler au Coff-IG. Il sera possible de modifier son adresse, la date de fin de contrat s’il est prolongé par exemple. Lorsque le contrat est terminé, on supprime l’employé de la base de données. Tous les services qu’il a effectué seront également supprimés de la base de données après confirmation du manager.

|  |  |  |  |
| --- | --- | --- | --- |
| Nom | Type | Nullable ? | Description |
| UserID | Integer | / | Clé étrangère, obligatoirement égale à la clé primaire de User. Clé primaire de la table. |
| HireDate | Date | / | Date d’engagement de l’employé. |
| EndContractDate | Date | Oui | Date de résiliation de contrat (prévue pour les CDD, donc facultative). |
| IsEmployeeOfTheMonth | Bit | / | Booléen définissant l’employé comme employé du mois (permet un éventuel pourcentage de réduction supplémentaire). |
| Discount | Double | / | Réduction allouée à chaque employé sur une commande personnelle de café(peut varier en fonction de la responsabilité de l’employé). |
| ParkingSpaceNumber | Integer | Oui | Numéro de parking éventuel réservé à l’employé (s’il vient en voiture). |
| ManagerID | Integer | Oui | Clé étrangère vers le manager dans le cas où l’employé est un subordonné. |

## Customer

La table client permet d’avoir un formulaire. On pourra ajouter un nouveau client qui désire avoir une carte de fidélité. On pourra modifier les informations du client telles que son adresse ou son numéro de téléphone. Si le client désire ne plus être enregistré, on pourra le supprimer de la base de données tout en gardant en mémoire les commandes qu’il a passées. Sa carte de fidélité sera également supprimée ainsi que les droits qui y étaient liés. Les avantages sont, eux, sauvegardés car ils sont liés à plusieurs cartes de fidélité. Le client pourra tout de même commander une fois la suppression effectuée.

|  |  |  |  |
| --- | --- | --- | --- |
| Nom | Type | Nullable ? | Description |
| UserID | Integer | / | Clé étrangère. Est obligatoirement égal à celui de User. |
| WantsAdvertising | Bit | / | Booléen représentant le choix de recevoir des publicités ou non. |
| SatisfactionDegree | Integer | Oui | Potentiel degré de satisfaction du client sur 5 (1/5 : pas satisfait ; 5/5 très satisfait). |
| LoyaltyCardID | Integer | Oui | Clé étrangère. Numéro de carte de fidélité de la personne si elle décide d’en avoir une. Un client peut ne pas utiliser sa carte quand il effectue une commande. Un client peut aussi effectuer des commandes sans être renseigné comme client et sans avoir de carte. |

## LoyaltyCard

|  |  |  |  |
| --- | --- | --- | --- |
| Nom | Type | Nullable ? | Description |
| LoyaltyCardID | Integer | / | Clé primaire. Numéro de carte de fidélité. |
| RegistrationDate | Date | / | Date de délivrance de la carte au client. |
| PointsNumber | Integer | / | Nombre de points collectés sur la carte de fidélité. |

## Right

|  |  |  |  |
| --- | --- | --- | --- |
| Nom | Type | Nullable ? | Description |
| AdvantageID | Integer | / | Clé étrangère vers l’avantage. |
| LoyaltyCardID | Integer | / | Clé étrangère vers la carte de fidélité. |

## Advantage

|  |  |  |  |
| --- | --- | --- | --- |
| Nom | Type | Nullable ? | Description |
| AdvantageID | Integer | / | Clé primaire. Identifiant de la réduction accordée. |
| Label | Varchar | / | Description de la réduction. |
| Discount | Double | / | Pourcentage de réduction lié à l’action. |
| StartDate | Date | / | Date de début de l’action |
| EndDate | Date | / | Date de fin de l’action |
| PointsRequired | Integer | / | Nombre de points requis pour la réduction. |

## Order

On mémorise les commandes de tous les clients, enregistrés ou non.

|  |  |  |  |
| --- | --- | --- | --- |
| Nom | Type | Nullable ? | Description |
| OrderNumber | Integer | / | Clé primaire de la table. |
| Date | Date | / | Date à laquelle est passée la commande. |
| IsToTakeAway | Bit | / | Si la commande est à emporter ou non. |
| Beneficiary | Integer | Oui | Clé étrangère vers l’identifiant du client qui commande.  Peut être null si le client se supprime de la base de données. On garde cependant ses commandes (pour calculer des statistiques. Ex : boisson la plus souvent commandée, …). |
| OrderPicker | Integer | Oui | Clé étrangère vers l’identifiant de l’employé qui prépare la commande. Si on supprime l’employé de la base de données, on garde les commandes qu’il a préparées. |

## Drink

|  |  |  |  |
| --- | --- | --- | --- |
| Nom | Type | Nullable ? | Description |
| CoffeeID | Integer | / | Identifiant composé et clé étrangère vers le type de café utilisé dans la boisson. |
| Label | Varchar | / | Identifiant composé. |
| Size | Varchar | / | Taille de la tasse de café (Small, Medium, Large). |
| IsCold | Bit | / | Café froid ou chaud. |

## DrinkOrdering

|  |  |  |  |
| --- | --- | --- | --- |
| Nom | Type | Nullable ? | Description |
| OrderNumber | Integer | / | Identifiant composé et clé étrangère vers la commande à laquelle appartient la ligne de commande. |
| DrinkLabel | Varchar | / | Identifiant composé et clé étrangère vers la boisson. |
| DrinkID | Integer | / |
| NbrDrinks | Integer | / | Nombre d’occurrences de la même boisson dans la ligne de commande. |
| SellingPrice | Double | / | Prix de vente de la boisson au moment de la commande, il peut être différent du prix du café et des toppings qui évoluent dans le temps. |

## Feature

|  |  |  |  |
| --- | --- | --- | --- |
| Nom | Type | Nullable ? | Description |
| Label | Varchar | / | Mot clé qui décrit le café pour le descriptif textuel. |

## Description

|  |  |  |  |
| --- | --- | --- | --- |
| Nom | Type | Nullable ? | Description |
| FeatureLabel | Varchar | / | Clé étrangère. |
| CoffeeID | Integer | / | Clé étrangère. |

## Coffee

La table café permet d’avoir un formulaire. On pourra ajouter une nouvelle variété de café dans la base de données pour la proposer à la vente. Le prix et le conditionnement pourront être modifiés (ainsi que les autres attributs). Si on désire ne plus vendre un type de café, on pourra le supprimer. Comme le prix est sauvegardé dans la commande, on pourra toujours garder une trace des ventes passées.

|  |  |  |  |
| --- | --- | --- | --- |
| Nom | Type | Nullable ? | Description |
| CoffeeID | Integer | / | Clé primaire de la table. |
| Label | Varchar | / | Nom du type de café utilisé dans la boisson. |
| OriginCountry | Varchar | / | Nom du pays d’origine pour le descriptif textuel. |
| Intensity | Integer | / | Intensité du café sur 5 (1 grain/5 : café doux ; 5 grains/5 : café fort). |
| WeightNeededForPreparation | Double | / | Nombre de grammes de grains de café pour préparer la boisson. Intervient dans la gestion du stock pour calculer la quantité stockée. |
| DiscoveryDate | Date | Oui | Date de découverte pour le descriptif textuel. |
| ExpirationDate | Date | / | Date de péremption du café. |
| IsInGrains | Bit | / | Pour savoir quel type de machine utiliser pour préparer la boisson. |
| IsEnvironmentFriendly | Bit | / | Café bio ou non. |
| Price | Double | / | Prix de vente du café (peut être différent d’une commande à la suivante). |
| Packaging | Double | / | Conditionnement des grains de café. |
| RecommendedConsummingMoment | Varchar | Oui | Moment de consommation préféré pour le descriptif textuel. |
| StockLocationAlley | Integer | / | Clé étrangère vers l’allée où est stocké le café. |
| StockLocationShelf | Integer | / | Clé étrangère vers l’étagère dans l’allée où est stocké le café. |
| StockLocationNumber | Integer | / | Clé étrangère vers le numéro sur l’étagère dans l’allée où est stocké le café. |

## Supplement

|  |  |  |  |
| --- | --- | --- | --- |
| Nom | Type | Nullable ? | Description |
| DrinkID | Integer | / | Clé étrangère vers l’identifiant de la boisson (type de café). |
| DrinkLabel | Varchar | / |
| ToppingID | Integer | / | Clé étrangère vers l’identifiant du topping. |

## Topping

|  |  |  |  |
| --- | --- | --- | --- |
| Nom | Type | Nullable ? | Description |
| ToppingID | Integer | / | Clé primaire de la table. |
| Label | Varchar | / | Nom du topping (ex : lait, sucre, caramel, …). |
| Price | Double | / | Prix de vente du topping (peut être différent d’une commande à la suivante). |
| StockLocationAlley | Integer | / | Clé étrangère vers l’allée où est stocké le topping. |
| StockLocationShelf | Integer | / | Clé étrangère vers l’étagère dans l’allée où est stocké le topping. |
| StockLocationNumber | Integer | / | Clé étrangère vers le numéro sur l’étagère dans l’allée où est stocké le topping. |

## FoodOrdering

|  |  |  |  |
| --- | --- | --- | --- |
| Nom | Type | Nullable ? | Description |
| OrderNumber | Integer | / | Identifiant composé et clé étrangère vers la commande à laquelle appartient la ligne de commande. |
| FoodID | Integer | / | Identifiant composé et clé étrangère vers la nourriture. |
| NbrPieces | Integer | / | Nombre d’aliments identiques dans une ligne de commande. |
| SellingPrice | Double | / | Prix de vente de l’aliment au moment de la commande, il peut être différent du prix du café et des topping qui évoluent dans le temps. |

## Food

|  |  |  |  |
| --- | --- | --- | --- |
| Nom | Type | Nullable ? | Description |
| FoodID | Integer | / | Clé primaire de la table. |
| Label | Varchar | / | Nom de la nourriture (ex : cookies, tarte aux pommes, …). |
| Price | Double | / | Prix de vente de la nourriture (peut être différent d’une commande à la suivante). |
| StockLocationAlley | Integer | / | Clé étrangère vers l’allée où est stockée la nourriture. |
| StockLocationShelf | Integer | / | Clé étrangère vers l’étagère dans l’allée où est stockée la nourriture. |
| StockLocationNumber | Integer | / | Clé étrangère vers le numéro sur l’étagère dans l’allée où est stockée la nourriture. |

## StockLocation

Les quantités en stock sont calculables à partir des cardinalités qui relient Food, Topping et Coffee. On pourra vérifier si les aliments sont toujours disponibles au moment de passer une commande en comparant les aliments en stock avec les quantités déjà consommées lors des commandes précédentes. S’il n’y a plus de stock pour la commande, le champ sera grisé.

|  |  |  |  |
| --- | --- | --- | --- |
| Nom | Type | Nullable ? | Description |
| Alley | Integer | / | Clés primaires de la table. |
| Shelf | Integer | / |
| Number | Integer | / |
| BuyingPrice | Double | / | Prix d’achat de la marchandise au fournisseur. La différence entre ce prix et le prix de vente permettra de calculer les bénéfices. |
| Quantity | Integer | / | Quantité en stock d’une marchandise. |