

Projet E-shop

Projet de remise à niveau MBDS

Spécification de l’analyse de conception

MOMAS Lisa

TAMANINI Nicolas

Table des matières

[1 Description du sujet 3](#_Toc50974307)

[2 Description textuelle des requêtes de mise à jour 3](#_Toc50974308)

[3 Description textuelle des requêtes de suppression 3](#_Toc50974309)

[4 Description textuelle des requêtes de consultation 3](#_Toc50974310)

[5 Dictionnaire de données MERISE 3](#_Toc50974311)

[6 Description textuelle des associations 3](#_Toc50974312)

[7 Définition du Modèle Entité-Association MERISE 3](#_Toc50974313)

[8 Définition du Modèle Logique de Données 4](#_Toc50974314)

[9 Spécification des traitements avec des packages PLSQL 4](#_Toc50974315)

[10 Spécification des triggers 4](#_Toc50974316)

[11 Maquettes d’écran 4](#_Toc50974317)

# Description du sujet

Pour le projet de remise à niveau, nous avons décidé de choisir le sujet de la gestion d’une boutique de vêtements en ligne. L’application sera orientée sur la partie « arrière-boutique » (back office) d’une boutique de vêtement et doit intégrer les fonctionnalités suivantes qui permettront aux clients du site de commander des vêtements et de se faire livrer la commande :

* Visualisation d’une liste de vêtements et filtrage par catégorie, taille, couleur, etc.
* Création, modification, suppression de vêtements
* Gestion des stocks de vêtements
* Gestion des commandes
* Gestion des comptes utilisateur

# Description textuelle des requêtes de mise à jour

# Description textuelle des requêtes de suppression

# Description textuelle des requêtes de consultation

# Dictionnaire de données MERISE

# Description textuelle des associations

Nous avons 6 tables dans notre SGBD e-shop.

* Nous avons la table UTILISATEUR qui contient les données nécessaires pour se créer un compte utilisateur (id, nom, prénom, adresse, email). Cet utilisateur choisi ses vêtements en fonction des déclinaisons disponibles, nous avons donc une liaison entre UTILISATEUR et DECLINAISON\_VETEMENT. Il peut n’y avoir aucun ou plusieurs utilisateurs qui choisissent aucune ou plusieurs déclinaisons. 0 à n déclinaisons peuvent être choisi par 0 à n utilisateurs.

Après avoir choisis ses vêtements, l’utilisateur peut passer 0 ou plusieurs commandes comprenant ses déclinaisons de vêtement. Aucune ou plusieurs déclinaisons de vêtements de la table DECLINAISON\_VETEMENT peuvent donc être commandés par 0 ou plusieurs COMMANDE.

* Nous avons la table VETEMENT qui contient les attributs id, nom, description. Cette table peut comprendre un « type » de vêtement. Cela peut être un t-shirt avec une écriture propre dessus. Ce vêtement fait partie d’une CATEGORIE, dans notre cas, le t-shirt avec écriture ferait partie de la catégorie t-shirt. Nous avons donc un vêtement qui appartient à une catégorie.

1 vêtement appartient donc à 1 catégorie.

Cependant, une catégorie peut n’avoir aucun ou plusieurs vêtements dans son sein. Nous avons donc 0 ou n vêtements dans une catégorie.

* Nous avons la table DECLINAISON\_VETEMENT. Elle comprend les données des différentes déclinaisons d’un vêtement. En effet, nous pouvons avoir un vêtement de différentes tailles ou couleurs, c’est pour cela que cette table est liée aux tables COULEUR et TAILLE. En fonction de sa taille ou couleur, le prix ou la quantité d’un vêtement peut changer d’où un attribut prix et quantité. Cependant, ce vêtement à un nom et une description propre qu’importe sa déclinaison et fait partie d’une catégorie. C’est pour cela que la table est également liée à la table vêtement.

1 déclinaison de vêtement peut avoir que 1 taille assignée ainsi que 1 couleur. Cette déclinaison appartient qu’à 1 « type » de vêtement mais ce type de vêtement peut avoir n déclinaison.

# Définition du Modèle Entité-Association MERISE

0 à n UTILISATEUR peut passer 0 à n COMMANDE comprenant des déclinaisons de vêtement. 1 COMMANDE peut être passée par 1 UTILISATEUR.

0 à n DECLINAISON\_VETEMENT peuvent avoir 0 à n COMMANDE.

0 à n COMMANDES peuvent être passées par 0 à n UTILISATEUR et avoir 1 à n DECLINAISON\_VETEMENT.

Un UTILISATEUR peut choisir 0 à n DECLINAISON\_VETEMENT, et 0 à n DECLINAISON\_VETEMENT peuvent être choisit par 0 à n UTILISATEUR.

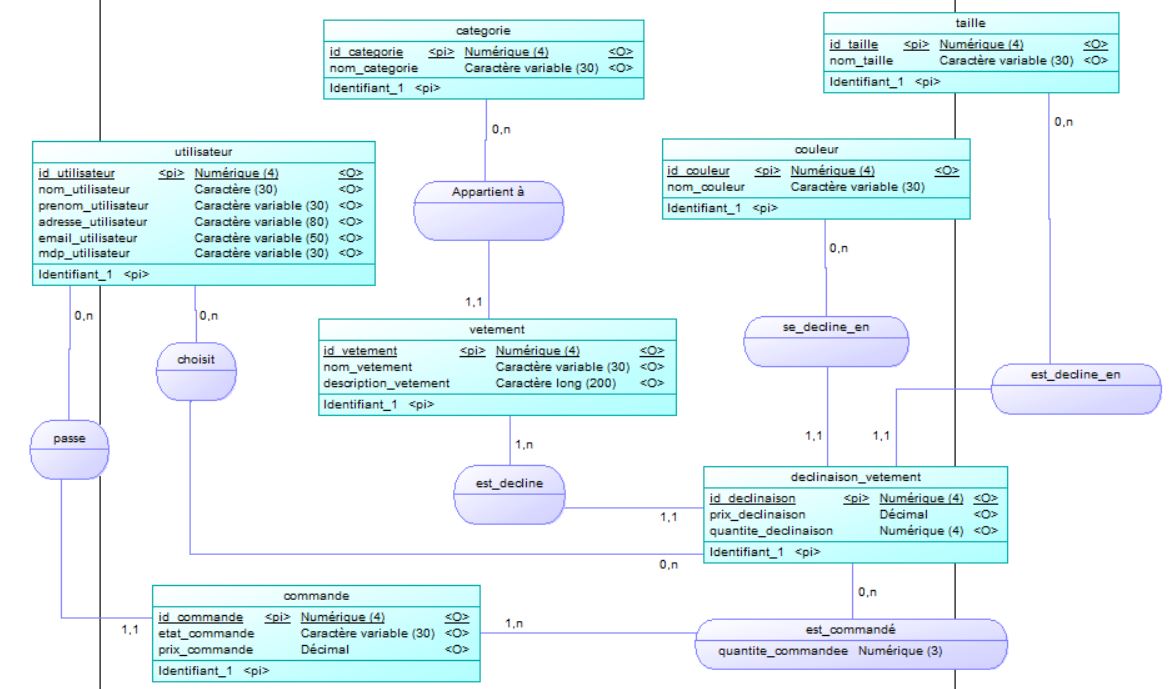
1 VETEMENT appartient à 1 CATEGORIE. Cependant, 1 CATEGORIE peut avoir 0 à N VETEMENT dans son sein.

1 DECLINAISON\_VETEMENT peut avoir que 1 TAILLE assignée ainsi que 1 COULEUR. 1 TAILLE ou COULEUR peut avoir 0 à n DECLINAISON\_VETEMENT.

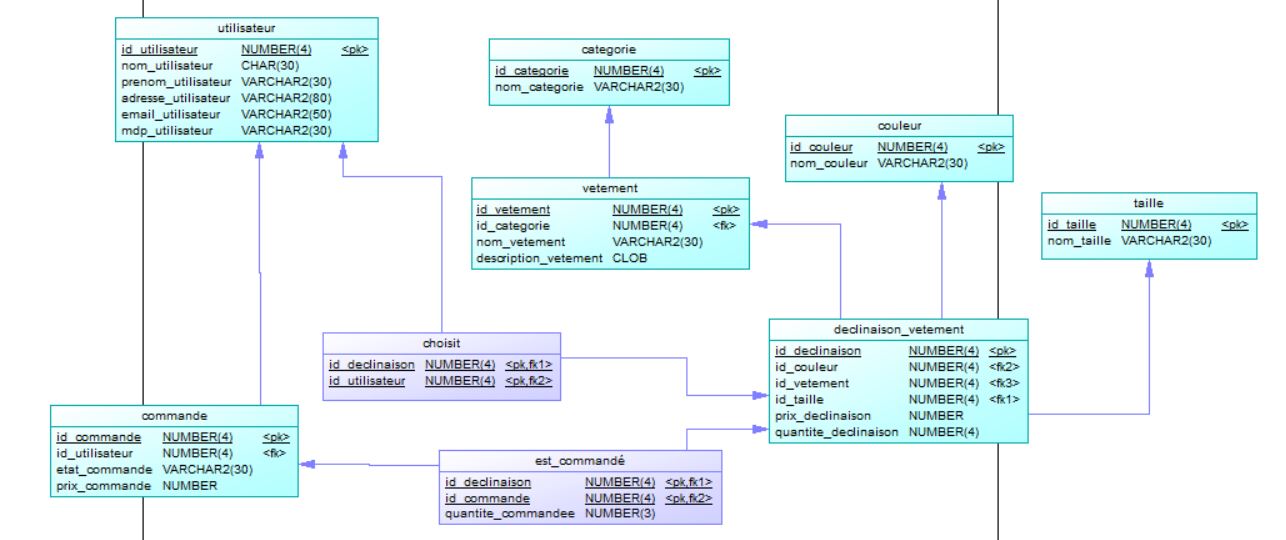
1 VETEMENT peut avoir 0 à n DECLINAISON\_VETEMENT ET 1 DECLINAISON\_VETEMENT peut avoir 1 à n VETEMENT.

# Définition du Modèle Logique de Données

Représentation du MCD réalisé sur PowerAMC



Représentation du MLD réalisé sur PowerAMC



# Spécification des traitements avec des packages PLSQL

Nous avons choisi de faire les fonctions et procédures PLSQL sur les tables UTILISATEUR (table A) et COMMANDE (table B).

Nous y avons réalisé des fonctions insérées permettant d’insérer dans la table une ligne avec tous ses attributs.

Les fonctions supprimer permet de supprimer une ligne du tableau correspondant à l’ID mis en paramètre de la fonction.

Pour la table UTILISATEUR, les fonctions modifiant les informations permettent de modifier un mot de passe (valeur nouvelle étant celle passée en paramètre) en fonction de l’id de l’utilisateur passé également en paramètre. Une autre permet de modifier les nom et prénoms par de nouvelles valeurs définis en paramètre en fonction de l’id rentrée en paramètre.

Pour la table COMMANDE, nous avons une fonction modifiant le prix d’une commande ainsi qu’une autre modifiant l’état de la commande (ex : nous passons d’un statut de commande « en cours » à un statue « expédié »).

Les fonctions lister permet d’afficher les informations du tableau de la forme suivante :

userId - userLastName| - userFirstName - userAddress - userEmail - userPassword. Elles listent les informations des utilisateurs.

# Spécification des triggers

Nous avons réalisé deux triggers.

Le premier se nomme USERDELETE\_TRIGGER et empêche la suppression des données de l’utilisateur lorsque celui-ci à déjà réalisé une commande. Cela permet de garder un suivi sur toutes les commandes réalisées au sein du e-shop.

Le deuxième trigger se nomme INSERTUSERTRIGGER et permet d’afficher à chaque insertion d’un utilisateur l’id, le nom et prénom du nouvel utilisateur.

Ces deux triggers sont spécifiés après un delete ou un insert.

# Maquettes d’écran

Nous avons fait des maquettes de notre site web sur AdobeXD. Notre logiciel devait idéalement remplir ces fonctionnalités et interfaces :

* Un écran de connexion

Cette page permet à l’utilisateur de se connecter.

L’adresse mail et le mot de passe de l’utilisateur doivent être intégrés dans la base de données Oracle. Dans le cas contraire :

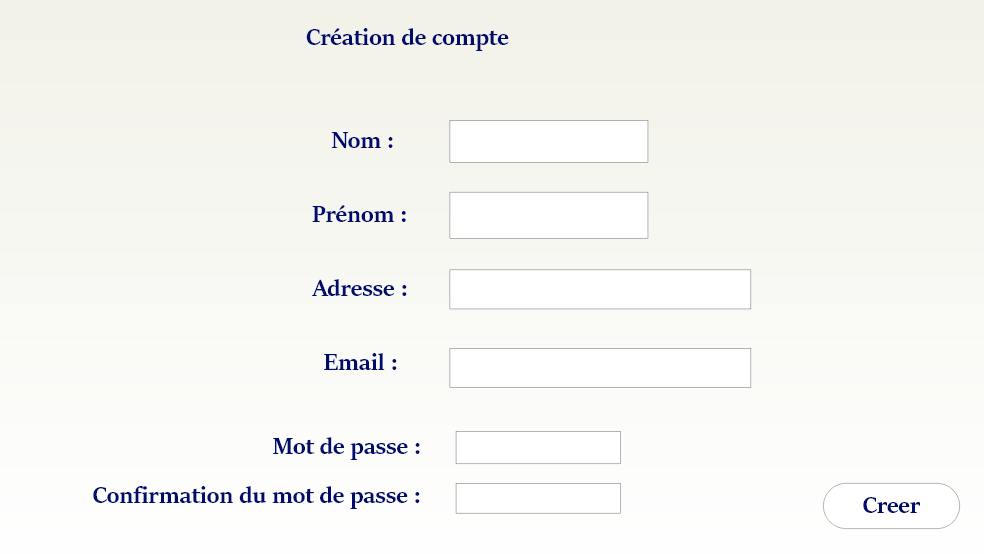
- L’utilisateur doit se créer de nouveaux identifiants en cliquant sur le lien hypertexte illustré sur le label « Se créer un compte ».

- L’utilisateur a oublié son mot de passe, il rentre l’adresse mail dans le champs dédiée à cela et cliquer sur le lien hypertexte « mot de passe oublié ».



* Ecran création de compte

Cette page permet à l’utilisateur de se créer un compte. Cela ajoute les informations à la base de données Oracle. Cette page est l’équivalent de notre page surprise.



* Un écran avec moteur de recherche (gérer la liste des données d’une table)

Dans cette écran, l’utilisateur peut réaliser plusieurs filtres. Il peut choisir de visualiser une liste de vêtement correspondant à la taille choisie et/ou à la couleur choisie et/ou la catégorie. Plus il y aura de filtre, moins il y aura de vêtements affichés dans la liste illustré dans le tableau situé au bas de la page. Les filtres sont cumulables et envoie un résultat instantané sur la liste.



* Ecran de gestion de vêtements

Cette page permet à l’interlocuteur de choisir d’ajouter ou supprimer un vêtement.

* La suppression d’un vêtement permet de supprimer toutes les déclinaisons liées à un vêtement. Les déclinaisons d’un vêtement choisis sont la taille, la couleur et la catégorie. L’utilisateur devra sélectionner le nom du vêtement qu’il veut supprimer.
* L’ajout d’un vêtement se fait d’abord par l’entrée du nom du vêtement et de sa description. L’utilisateur doit ensuite cliquer sur le bouton ajouter et cela renverra automatiquement à la page « écran de gestion de déclinaison ».



* Ecran de gestion de déclinaison

Cet écran permet à l’utilisateur d’ajoute ou supprimer des déclinaisons de vêtements.

Le remplissage de tous les champs est obligatoire lorsque l’utilisateur à auparavant cliqué sur le bouton « ajouter » de l’écran « gestion de vêtement ».

Dans le cas contraire, l’utilisateur peut ajouter ou supprimer une quantité d’une déclinaison de vêtement en sélectionnant dans un premier temps le nom du vêtement (comboBox situé en haut de l’écran), la couleur, la taille et la catégorie du vêtement. La démarche est la même pour supprimer une quantité d’une déclinaison du vêtement. Il faudra pour cela cliquer sur le bouton supprimer.

