

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
2	Description textuelle des requêtes de mise à jour	3
3	Description textuelle des requêtes de suppression	4
4	Description textuelle des requêtes de consultation	4
5	Dictionnaire de données MERISE	6
6	Description textuelle des associations.....	7
7	Définition du Modèle Entité-Association MERISE	8
8	Définition du Modèle Logique de Données.....	9
9	Spécification des traitements avec des packages PLSQL	10
10	Spécification des triggers	10
11	Maquettes d'écran	10

1 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

2 Description textuelle des requêtes de mise à jour

REQUETES DE MISE A JOUR

-- 2 requêtes impliquant 1 table :

-- Réduction de 10% sur tous les vêtements

```
UPDATE declinaison_vetement SET prix_declinaison = prix_declinaison * 0.90;
```

-- Réduction de 20% sur tous les vêtements dont le prix est supérieur à 80€

```
UPDATE declinaison_vetement SET prix_declinaison = prix_declinaison * 0.80 WHERE  
prix_declinaison >= 80;
```

-- 2 requêtes impliquant 2 tables :

-- Reduction de 20% sur tous les vetements dont la taille est égale à XXL

```
UPDATE declinaison_vetement dv JOIN taille t ON dv.id_taille = t.id_taille SET  
dv.prix_declinaison = v.prix_declinaison * 0.80 WHERE t.nom_taille = 'XXL' ; --  
FONCTIONNE PAS
```

-- Augmentation de 5% sur les vetements rouges

```
UPDATE declinaison_vetement dv JOIN couleur c ON dv.id_couleur= c.id_couleur SET  
dv.prix_declinaison = dv.prix_declinaison * 1.05 WHERE dv.nom_couleur = 'Rouge' ; --  
FONCTIONNE PAS
```

-- 2 requêtes impliquant plus de 2 tables :

-- Augmentation de 5% sur les vetements rouges et de taille

```
UPDATE declinaison_vetement dv JOIN couleur c ON dv.id_couleur= c.id_couleur JOIN
taille t ON dv.id_taille = t.id_taille SET dv.prix_declinaison = v.prix_declinaison * 1.05
WHERE c.nom_couleur = 'Rouge' AND t.nom_taille = 'S' ; -- FONCTIONNE PAS
```

```
-- Réduction de 5€ pour les vêtements rouges de taille M avec un prix supérieur à 9€--
UPDATE declinaison_vetement dv JOIN couleur c ON dv.id_couleur= c.id_couleur JOIN
taille t ON dv.id_taille = t.id_taille SET dv.prix_declinaison = dv.prix_declinaison - 5 WHERE
dv.prix_declinaison > 9 AND c.nom_couleur = 'Rouge' AND t.nom_taille = 'M' ; --
FONCTIONNE PAS
```

3 Description textuelle des requêtes de suppression

REQUETES DE SUPPRESSION

```
-- 2 requêtes impliquant 1 table :
```

```
-- Supprimer une ligne de la table declinaison_vetement (suppression d'un vêtement)
DELETE FROM declinaison_vetement WHERE id_declinaison = 1;
```

```
-- Supprimer tous les vêtements dont le prix est supérieur à 50€
DELETE FROM declinaison_vetement WHERE prix_declinaison > 50;
```

```
-- 2 requêtes impliquant 2 tables :
```

```
-- Supprimer toutes les commandes d'un utilisateur
DELETE (SELECT * FROM commande JOIN utilisateur ON utilisateur.id_utilisateur =
commande.id_utilisateur WHERE utilisateur.id_utilisateur = 5);
```

```
-- Supprimer tous les vêtements dont la taille est 'M'
DELETE (SELECT * FROM declinaison_vetement JOIN taille ON
declinaison_vetement.id_taille = taille.id_taille WHERE taille.nom_taille = 'M');
```

```
-- 2 requêtes impliquant plus de 2 tables :
```

```
-- Supprimer tous les vêtements dont la catégorie est "T-shirt"
DELETE (SELECT * FROM declinaison_vetement JOIN vetement ON
declinaison_vetement.id_vetement = vetement.id_vetement JOIN categorie ON
vetement.id_categorie = categorie.id_categorie WHERE categorie.nom_categorie = 'T-shirt');
```

```
--Supprimer les commandes contenant un vêtement dans la catégorie 'Pantalon'
DELETE (SELECT * FROM commande JOIN est_commande ON commande.id_commande
= est_commande.id_commande JOIN declinaison_vetement ON
est_commande.id_declinaison = declinaison_vetement.id_declinaison JOIN vetement ON
declinaison_vetement.id_vetement = vetement.id_vetement JOIN categorie ON
vetement.id_categorie = categorie.id_categorie WHERE categorie.nom_categorie =
'Pantalon');
```

4 Description textuelle des requêtes de consultation

DESCRIPTION TEXTUELLES DES REQUETES DE CONSULTATION

```

-- 5 requêtes impliquant 1 table dont 1 avec un group By et une avec un Order By :

-- Trier les vêtements par ordre croissant du prix
SELECT id_declinaison, prix_declinaison FROM declinaison_vetement ORDER BY
prix_declinaison;

-- Regrouper les vêtements de quantité disponible par couleur
SELECT id_couleur, COUNT(*) FROM declinaison_vetement GROUP BY id_couleur
HAVING quantite_declinaison > 0; -- FONCTIONNE PAS

-- Filtrer les vêtements par gamme de prix. Ici c'est entre 20 et 40€.
SELECT * FROM declinaison_vetement WHERE prix_declinaison BETWEEN 20 AND 40;

-- Lister les vêtements dont le prix est inférieur à 30€.
SELECT * FROM declinaison_vetement WHERE prix_declinaison < 31;

-- Lister les vêtements dont le prix est égal à 30€.
SELECT * FROM declinaison_vetement WHERE prix_declinaison = 30;

-- 5 requêtes impliquant 2 tables avec jointures internes dont 1 externe + 1 group by + 1 tri :

-- Donner le prix des commandes regroupées par utilisateur (GROUP BY)
SELECT utilisateur.id_utilisateur, SUM(commande.prix_commande) FROM utilisateur JOIN
commande ON utilisateur.id_utilisateur = commande.id_utilisateur GROUP BY
utilisateur.id_utilisateur;

-- Donner la liste des utilisateurs avec les commandes qui leurs sont associées. Si un utilisateur
n'a jamais commandé, on affiche quand même son nom
SELECT prenom_utilisateur, nom_utilisateur, id_commande FROM utilisateur LEFT OUTER
JOIN commande ON utilisateur.id_utilisateur = commande.id_utilisateur;

-- Donner la liste des déclinaisons de vêtements de taille 'M'
SELECT id_declinaison, nom_taille FROM declinaison_vetement JOIN taille ON
declinaison_vetement.id_taille = taille.id_taille WHERE taille.nom_taille = 'M';

-- Donner la liste des déclinaisons de vêtements de couleur rouge
SELECT id_declinaison, nom_couleur FROM declinaison_vetement JOIN couleur ON
declinaison_vetement.id_couleur = couleur.id_couleur WHERE couleur.nom_couleur =
'Rouge';

-- Donner la liste des vêtements de catégorie 'Pantalon'
SELECT nom_vetement, nom_categorie FROM vetement JOIN categorie ON
vetement.id_categorie = categorie.id_categorie WHERE categorie.nom_categorie = 'Pantalon';

--Trier les vêtements par catégorie

-- 5 requêtes impliquant plus de 2 tables avec jointures internes dont 1 externe + 1 group by +
1 tri :

```

-- Donner la liste des utilisateurs et de la ou des commande(s) contenant ayant vêtement de taille S que ces utilisateurs ont passé.

```
SELECT prenom_utilisateur, nom_utilisateur, commande.id_commande FROM utilisateur
JOIN commande ON utilisateur.id_utilisateur = commande.id_utilisateur JOIN est_commande
ON commande.id_commande = est_commande.id_commande JOIN declinaison_vetement ON
est_commande.id_declinaison = declinaison_vetement.id_declinaison JOIN taille ON
declinaison_vetement.id_taille = taille.id_taille WHERE taille.nom_taille = 'S';
```

-- Donner la liste des déclinaisons de vêtements dont la catégorie est 'T-Shirt' triés par prix décroissants

```
SELECT id_declinaison, nom_vetement, prix_declinaison, quantite_declinaison FROM
declinaison_vetement JOIN vetement ON declinaison_vetement.id_vetement =
vetement.id_vetement JOIN categorie ON vetement.id_categorie = categorie.id_categorie
WHERE nom_categorie='T-Shirt' ORDER BY prix_declinaison DESC;
```

-- Donner la liste des déclinaisons de vêtements dont la taille est S et la couleur est noir

```
SELECT id_declinaison, nom_taille, nom_couleur FROM declinaison_vetement JOIN taille
ON declinaison_vetement.id_taille = taille.id_taille JOIN couleur ON
declinaison_vetement.id_couleur = couleur.id_couleur WHERE nom_taille = 'S' AND
nom_couleur='Noir';
```

-- Donner le nom et la catégorie des vêtements étant associés à des déclinaisons. Si un vêtement n'est associé à aucune déclinaison, le nom et la catégorie de ce vêtement sont quand même affichés.

```
SELECT nom_categorie, nom_vetement, id_declinaison FROM vetement LEFT OUTER JOIN
categorie ON vetement.id_categorie = categorie.id_categorie LEFT OUTER JOIN
declinaison_vetement ON vetement.id_vetement = declinaison_vetement.id_vetement;
```

-- Donner la valeur marchande du stock de vêtements par catégorie trié par valeur marchande décroissante

```
SELECT categorie.nom_categorie, SUM(declinaison_vetement.prix_declinaison)"Valeur
marchande du stock par catégorie" FROM vetement JOIN categorie ON vetement.id_categorie
= categorie.id_categorie JOIN declinaison_vetement ON vetement.id_vetement =
declinaison_vetement.id_vetement GROUP BY categorie.nom_categorie ORDER BY
SUM(declinaison_vetement.prix_declinaison) DESC;
```

5 Dictionnaire de données MERISE

Vous trouverez ci-joint le dictionnaire de donnée MERISE de chaque entité.

Entité VETEMENT

Entité : vetement						
NOM CODE	LIBELLE	TYPE	FORMAT	Identifiant ou Unique	Obligatoire	CONTRAINTE
id_vetement	ID_VETEMENT	Numérique	999999	OUI	oui	>=1
nom_vetement	NOM_VETEMENT	Alphanumérique	A30	NON	oui	Première Lettre Majuscule
description_vetement	DESCRIPTION_VETEMENT	Alphanumérique	A200	NON	oui	Première Lettre Majuscule

Entité DECLINAISON_VETEMENT

Entité : declinaison_vetement						
NOM CODE	LIBELLE	TYPE	FORMAT	Identifiant ou Unique	Obligatoire	CONTRAINTE
id_declinaison	ID_DECLINAISON	Numérique	999999	OUI	oui	>=1
prix_declinaison	PRIX_DECLINAISON	Numérique	999999.99	NON	oui	>=2 et <=300
quantite_declinaison	QUANTITE_DECLINAISON	Numérique	999999	NON	oui	>=0

Entité UTILISATEUR

Entité : utilisateur						
NOM CODE	LIBELLE	TYPE	FORMAT	Identifiant ou Unique	Obligatoire	CONTRAINTE
id_utilisateur	ID_UTILISATEUR	Numérique	999999	OUI	oui	>=1
nom_utilisateur	NOM_UTILISATEUR	Alphanumérique	A30	NON	oui	Première Lettre Majuscule
prenom_utilisateur	PRENOM_UTILISATEUR	Alphanumérique	A30	NON	oui	Première Lettre Majuscule
adresse_utilisateur	ADRESSE_UTILISATEUR	Alphanumérique	A80	NON	oui	
email_utilisateur	EMAIL_UTILISATEUR	Alphanumérique	A50	NON	oui	
mdp_utilisateur	MDP_UTILISATEUR	Alphanumérique	A30	NON	oui	

Entité COMMANDE

Entité : commande						
NOM CODE	LIBELLE	TYPE	FORMAT	Identifiant ou Unique	Obligatoire	CONTRAINTE
id_commande	ID_COMMANDE	Numérique	999999	OUI	oui	>=1
etat_commande	ETAT_COMMANDE	Alphanumérique	A30	NON	oui	Première Lettre Majuscule
prix_commande	PRIX_COMMANDE	Numérique	99999.99	NON	oui	>=2 et <=2000

Entité CATEGORIE

Entité : categorie						
NOM CODE	LIBELLE	TYPE	FORMAT	Identifiant ou Unique	Obligatoire	CONTRAINTE
id_categorie	ID_CATEGORIE	Numérique	999999	OUI	oui	>=1
nom_categorie	NOM_CATEGORIE	Alphanumérique	A30	NON	oui	Première Lettre Majuscule

Entité TAILLE

Entité : taille						
NOM CODE	LIBELLE	TYPE	FORMAT	Identifiant ou Unique	Obligatoire	CONTRAINTE
id_taille	ID_TAILLE	Numérique	999999	OUI	oui	>=1
nom_taille	NOM_TAILLE	Alphanumérique	A30	NON	oui	Première Lettre Majuscule

Entité COULEUR

Entité : couleur						
NOM CODE	LIBELLE	TYPE	FORMAT	Identifiant ou Unique	Obligatoire	CONTRAINTE
id_couleur	ID_COULEUR	Numérique	999999	OUI	oui	>=1
nom_couleur	NOM_COULEUR	Alphanumérique	A30	NON	oui	Première Lettre Majuscule

6 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.

7 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 choisis par 0 à n UTILISATEUR.

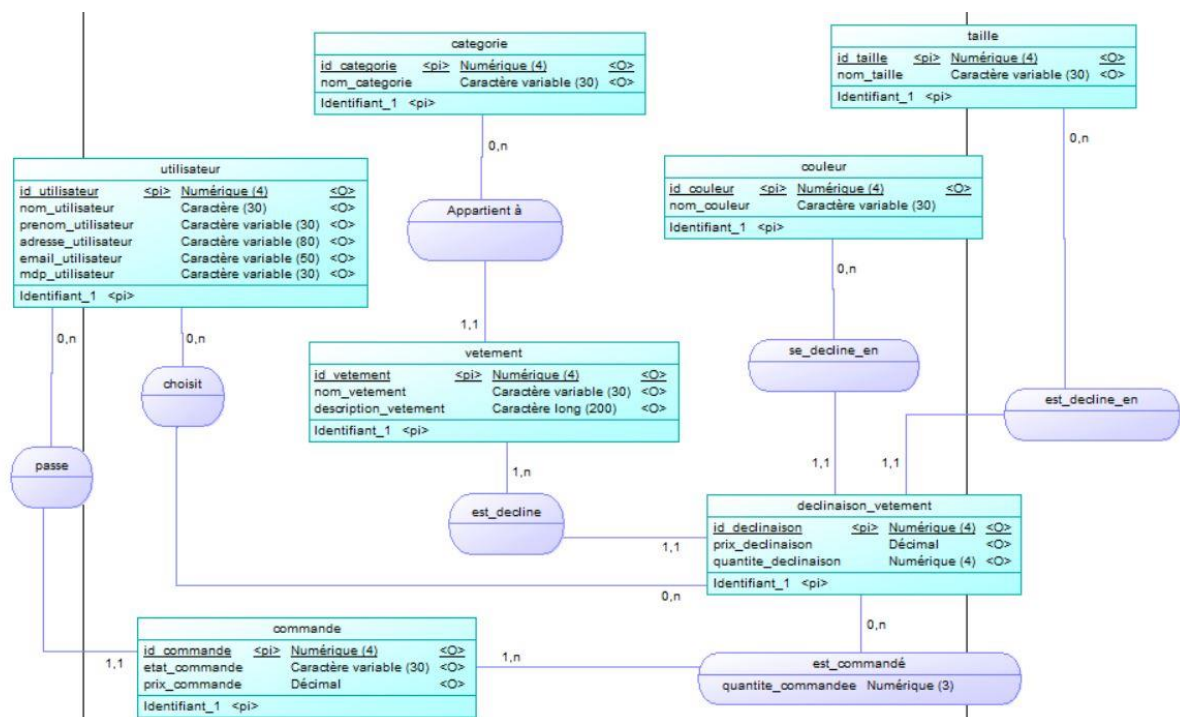
1 VETEMENT appartient à 1 CATEGORIE. Cependant, 1 CATEGORIE peut avoir 0 à N VETEMENT dans son sein.

1 DECLINAISON_VETEMENT peut avoir que 1 TAILLE assignée ainsi que 1 COULEUR. 1 TAILLE ou COULEUR peut avoir 0 à n DECLINAISON_VETEMENT.

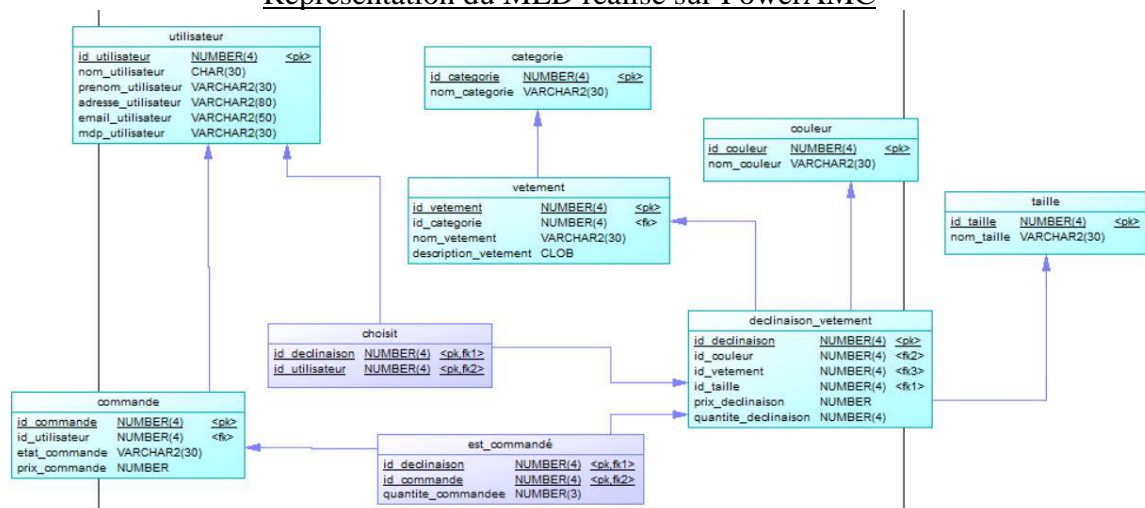
1 VETEMENT peut avoir 0 à n DECLINAISON_VETEMENT ET 1 DECLINAISON_VETEMENT peut avoir 1 à n VETEMENT.

8 Définition du Modèle Logique de Données

Représentation du MCD réalisé sur PowerAMC



Représentation du MLD réalisé sur PowerAMC



9 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.

10 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.

11 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 ».



The login form is titled "Bienvenue" in orange. It includes a logo for "TAMANINI MOMAS" in the top left. There are two input fields: "Adresse Mail" and "Mot de passe". Below the password field, there is a link "Mot de passe oublié" and a text line "Vous n'avez toujours pas de compte? - Se créer un compte".

TAMANINI
MOMAS

Bienvenue

Adresse Mail

Mot de passe

[Mot de passe oublié](#)

Vous n'avez toujours pas de compte? - [Se créer un compte](#)

- 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.



The account creation form is titled "Création de compte" in blue. It contains several input fields for user information: "Nom", "Prénom", "Adresse", "Email", "Mot de passe", and "Confirmation du mot de passe". A "Creer" button is located at the bottom right.

Création de compte

Nom :

Prénom :

Adresse :

Email :

Mot de passe :

Confirmation du mot de passe :

[Creer](#)

- 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.

The screenshot shows a web interface titled "Moteur de recherche" (Search Engine) in blue text. Below the title, there are three filter buttons: "Catégories" (Categories), "Couleurs" (Colors), and "Tailles" (Sizes). Each button has a small downward-pointing triangle icon to its right, indicating a dropdown menu. Below these buttons, the text "Liste des vêtements correspondants :" (List of corresponding clothing items :) is displayed in blue. Underneath this text is a large, empty rectangular box, which is intended to display the search results.

- 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 ».

Ajout ou suppression d'un vêtement

Nom du vêtement :

Ajouter

Description du vêtement :

Entrer le vêtement à supprimer :

Supprimer

- Ecran de gestion de déclinaison

Cet écran permet à l'utilisateur d'ajouter ou supprimer des déclinaisons de vêtements.

Le remplissage de tous les champs est obligatoire lorsque l'utilisateur a 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.

The screenshot shows a web form titled "Ajout ou suppression d'une déclinaison". At the top, there is a dropdown menu. Below it, the form is organized into two rows. The first row contains three labels: "Taille :", "Couleur :", and "Quantité :". Each label is followed by a corresponding input field: a dropdown for "Taille", a dropdown for "Couleur", and a text input for "Quantité". The second row contains the label "Ajout d'une couleur :" followed by a text input field. To the right of this input is a button labeled "Ajouter couleur". At the bottom of the form, there are two large, rounded buttons: "Ajouter" on the left and "Supprimer" on the right.