CAHIER DES CHARGES

**Table des matières**

[1. Expressions de besoin : 1](#_Toc511287171)

[2. Analyse: 2](#_Toc511287172)

[2.1 Les rôles 2](#_Toc511287173)

[2.2 Schémas des cas d’utilisations UML 2](#_Toc511287174)

[2.3 Schémas des cas d’utilisations détaillée 2](#_Toc511287175)

[2.4 ECRANS 3](#_Toc511287176)

[2.5 MCD 3](#_Toc511287177)

[3. Conception**:** 4](#_Toc511287178)

[3.2 Utilisation de la structuration MVC 4](#_Toc511287179)

[3.3 Conception détaillée 4](#_Toc511287180)

[**Se connecter** (les scripts) 4](#_Toc511287181)

# Expressions de besoin :

3 rôles :

Contraintes : Sécurisation du site 🡪 Interdire l’accès aux pages de admin

# Analyse:

## 2.1 Les rôles

* 1. Administrateurs : gérer client, gérer fleurs …
  2. Client : lister fleurs, rechercher fleurs, gestion panier, commandes …
  3. Internaute : lister fleurs, rechercher fleurs

## 2.2 Schémas des cas d’utilisations UML



## 2.3 Schémas des cas d’utilisations détaillée

**Action Modifier**

1. Le système affiche le menu
2. L’utilisateur sélectionne l’action Modifier dans le menu
3. Le système affiche un formulaire avec [ref, des, prix, image, cat]
4. L’utilisateur saisit les données
5. Le système enregistre les modifications, et affiche un message de réussite.

Exceptions :

5.1 Si la référence saisie n’existe pas, le système affiche un message d’erreur

Améliorations :

3. Le système affiche un formulaire avec [reference]

4. L’utilisateur saisit la référence souhaitée

5. Le système affiche la référence avec [designation, prix, image, categorie]

6. L’utilisateur choisit et modifier les valeurs souhaitées

7. Le système enregistre les modifications, et affiche un message de réussite.

**Action Rechercher**

1. Le système affiche le menu
2. L’utilisateur sélectionne l’action Rechercher dans le menu
3. Le système affiche un formulaire de recherche [designation]
4. L’utilisateur saisit la désignation.
5. Le système retourne le résultat de la recherche.

Exceptions :

* 1. Si la désignation n’existe pas, le système affiche un message d’échec.

**Action « Se connecter »**

1. Le système affiche le menu
2. L’utilisateur sélectionne « Se connecter »
3. Le système affiche un formulaire avec nom et mdp
4. L’utilisateur entre les données et valide
5. Le système reconnait l’utilisateur et affiche les services proposés dans le menu (et affiche les informations utilisateurs)

Exceptions :

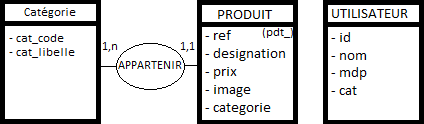
* 1. Si le système ne reconnait pas l’utilisateur, un message d’erreur est affiché.

## 2.4 ECRANS



Arborescence des pages

## 2.5 MCD



# Conception**:**

## 3.2 Utilisation de la structuration MVC

M : Modèle (structuration des données et accès à la nase de données)

\_bdGestionDonnees.lib

V : Vue (affichage)

entete.php

C : Contrôleur

Traiter la logique algorithmique des interactions liées à un service

ajouter.php, lister.php, etc…

## 3.3 Conception détaillée

### Se connecter (les scripts)

- **Modèle**

- **Contrôleur** :

Etape 1 : (clic sur le menu)

Appeler la vue vConnecter

Etape 2 : (validation du formulaire)

Appeler la fonction du Modele : Rechercher ($nom, $mdp, $tabErreurs)

Si la recherche donne un résultat, ou initialise la session : affecterInfosConnecte ($nom, $mdp)

sinon on affiche un message d'erreurs

- **Vues**:

Vue vConnecter

Formulaire avec les champs identifiant et mot de passe