



Projet de Fin d'Etudes
Licence Sciences et Techniques Génie Informatique

Développement d'un système
Comparateur de prix



Lieu de stage : Sofeweb - Oujda

Réalisé par :

Samir MAFTAH
Taoufik ZELMATI

Encadré par :

Pr. Ahlame BEGDOURI
Pr. Azeddine ZAHI

Soutenu le 16/06/2011 devant le jury composé de :

Pr. Rachid BENABBOU (Président)
Pr. Said NAJAH (Rapporteur)
Pr. Ahlame BEGDOURI (Encadrant)

Année Universitaire 2010-2011

REMERCIEMENTS

Nous tenons à exprimer nos sentiments de reconnaissances à toutes les personnes qui, par leur aide et leurs encouragements, nous ont permis de réaliser ce travail dans les meilleures conditions.

En préambule, nous adressons nos vifs remerciements à M. Yassine MASSOUATH le directeur adjoint de « Sofeweb », pour l'opportunité qu'il nous a octroyée en acceptant de nous prendre en stage au sein de l'association et aussi pour son encadrement à Sofeweb.

Nous présentons nos sincères gratitudees à Mme. Ahlame BEGDOURI pour son encadrement et aussi pour ses prestigieux conseils, ses remarques pertinentes et sa grande disponibilité tout au long de cette période.

Un merci du fond du cœur à nos parents, nos sœurs, nos frères et toute la famille qui ont toujours été à nos côtés.

Enfin, nous souhaitons remercier tout le corps professoral et administratif de la Faculté des Sciences et Techniques de Fès, qui fournit d'énormes efforts pour ses étudiants, afin qu'ils puissent jouir d'une formation complète, dans un climat de confiance et de convivialité.

Tables des matières

Introduction	4
Chapitre I : Cadre général du projet	5
1- Organisme d'accueil : Sofeweb.....	6
1.1- Présentation de Sofeweb	6
1.2- Organisation de Sofeweb	6
2- Description informelle du projet.....	7
2.1- Objectifs du projet.....	7
2.2- Fiche annuaire	7
Chapitre II : Dossier de conception.....	9
1. Démarche suivie : Processus 2TUP.....	10
2. Cahier de charge	11
3. Modélisation du contexte	13
3.1- Identification des acteurs.....	13
3.2- Identifications des messages.....	14
4. Capture des besoins fonctionnelles	15
4.1- Identification des cas d'utilisation.....	15
4.2- Description détaillée des cas d'utilisation	19
5. Analyse.....	22
5.1- Modèle statique	22
5.2- Modèle dynamique	23
Chapitre III : Étude technique du projet.....	27
1. Architecture et structure de l'application.....	28
2. Outils technologiques	29
Chapitre IV : Présentation de l'application.....	32
1. Ecran d'accueil.....	34
2. Fonctionnalités de l'internaute	34
3. Fonctionnalités du membre.....	35
4. Fonctionnalités du marchand.....	36
5. Fonctionnalités de l'administrateur	39
Conclusion	41
Annexe.....	43

Introduction

Au terme de notre formation en filière « Licence Science et Technique – Génie Informatique » au sein de la Faculté des Sciences et Techniques de Fès, nous sommes amenés à effectuer un stage de fin d'études de huit semaines afin d'appliquer les notions théoriques et pratiques acquises durant cette année.

Notre choix s'est porté sur la société « Sofeweb » spécialisée dans les systèmes d'informations et des hautes technologies. L'excellent dévouement que porte cette société au service de l'évolution des systèmes d'information nous a convaincu pour être la société adéquate où nous pouvions effectuer notre stage.

Nous présentons à travers ce rapport quatre parties qui traiteront tous les points importants ainsi que toutes les étapes.

La première partie définit le cadre général de notre projet, il présente en premier lieu l'organisme d'accueil, puis une description du projet.

La deuxième partie présente le dossier de conception du projet, elle décrit le cahier de charges à réaliser, le contexte du système à développer, l'étude fonctionnelle et l'analyse du projet.

La troisième partie donne une description de l'étude technique, elle présente l'architecture de notre système ainsi que les outils de développement utilisés.

La dernière partie est dédiée à la présentation du système que nous avons réalisé.

Chapitre I

Cadre général du projet

*La première partie aura pour objectif la
présentation de l'organisme d'accueil
« Sofeweb » et une description générale du projet*

1- Organisme d'accueil : Sofeweb

1.1- Présentation de Sofeweb

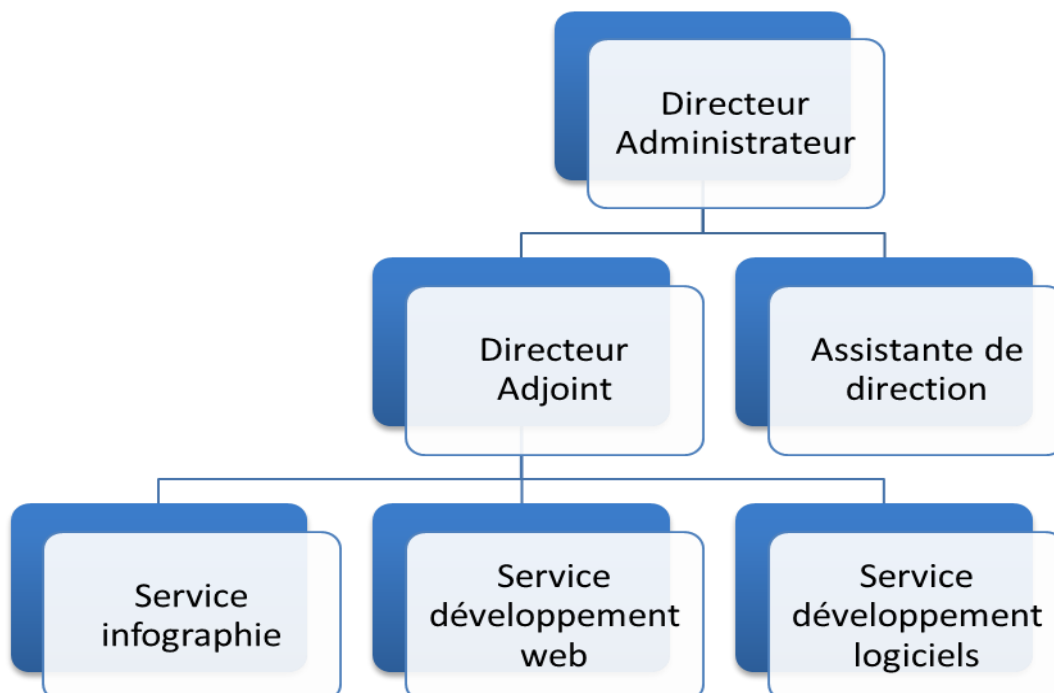
La société SOFEWEB est une société à responsabilité limitée, fondée il y a plus de deux ans autour de l'idée : conception et développement des sites web, mais très vite, son activité a été élargie par l'arrivée d'autres actionnaires à l'édition et le développement de logiciels standards et sur mesure.

La demande accrue des prestations de services en termes de technologie moderne dans le domaine de l'enseignement primaire, non seulement en termes de scolarité mais aussi éducation et pédagogie, a induit l'idée de la mise en place et le développement d'un service d'Animation parascolaire.

Les services de SOFEWEB sont :

- Edition et développement de logiciel (standard et sur mesure).
- Conception et développement de sites web.
- Infographie et animation.
- Animation parascolaire.

1.2- Organisation de Sofeweb



2- Description informelle du projet

2.1- Objectifs du projet

Les comparateurs de prix sont des sites qui permettent à leurs utilisateurs de comparer les offres de sociétés en ligne selon plusieurs critères, dont le prix. Si l'utilisateur est intéressé par une des offres, il clique et se trouve dirigé vers le site de la société.

Pour une société, avoir son catalogue produit référencé dans ces sites est un puissant moyen d'obtenir du trafic qualifié et c'est un bon outil de veille concurrentielle.

Lors de l'inscription, le marchand, qui est la personne qui représente la société, saisit les différentes informations qui permettra par la suite à l'administrateur de juger la légitimité de cette société.

2.2- Fiche annuaire

Les sociétés nouvellement inscrites dans le système comparateur de prix ne pourront pas bénéficier de toutes les fonctionnalités du système tant que leur compte n'est pas validé par Sofeweb. Cette validation consiste à vérifier principalement la validité des informations concernant la société en question.

Une fiche annuaire doit regrouper les différentes informations de la société, saisies par le marchand et qui sont nécessaires à la validation ou non de la société.

Une fois la société validée, son représentant aura la possibilité d'ajouter des offres et les envoyer pour qu'elles soient confirmées par l'administrateur (voir Figure.1).

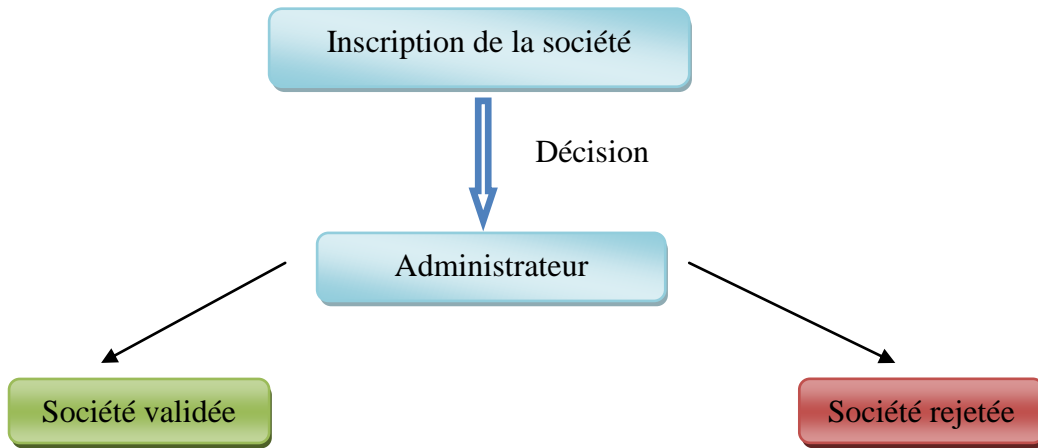


Figure.1 : Les étapes de validation de la société

Les offres ajoutées par le marchand doivent être validées par l'administrateur pour qu'elles soient visibles sur le comparateur de prix. Cette validation consiste à vérifier le lien de l'offre sur le site du marchand (Figure.2).

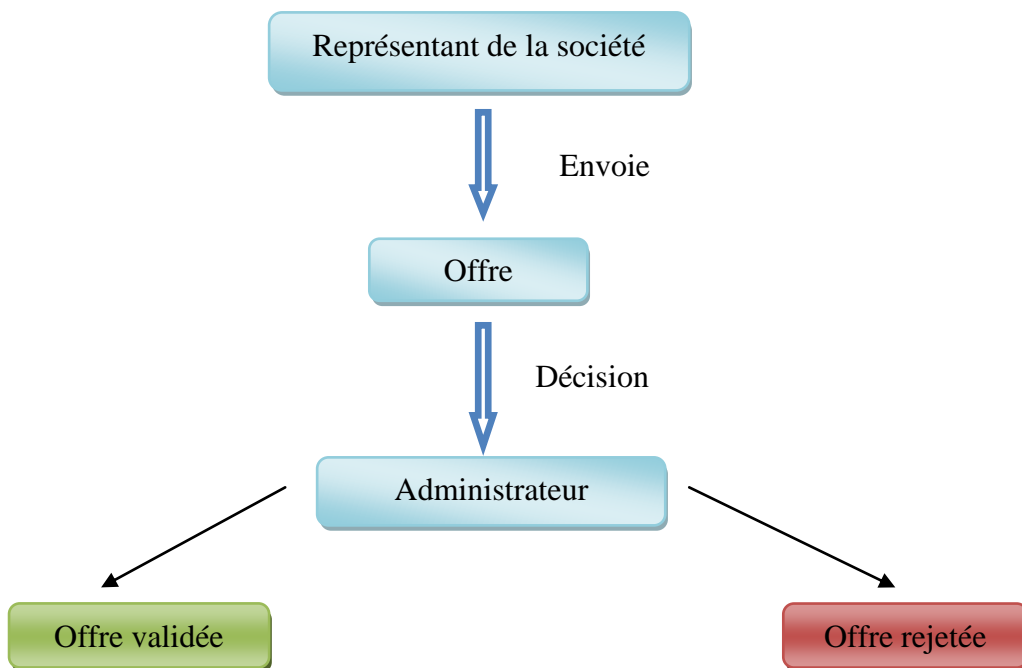


Figure.2 : Les étapes de validation d'une offre

Chapitre II

Dossier de conception

Lors de cette étude nous allons présenter le cahier de charge, la modélisation du contexte de notre système ainsi que l'étude fonctionnelle du projet

1. Démarche suivie : Processus 2TUP

Pour la réalisation de notre projet, nous avons utilisé le processus 2TUP.

Ce processus dissocie les aspects fonctionnels des aspects techniques permettant ainsi de répondre aux contraintes de changement continuels imposées au système. Il s'articule sur 3 branches :

- *La branche gauche (fonctionnelle)* : Les principales étapes de la branche fonctionnelle se présentent comme suit :
 - L'étape capture des besoins fonctionnels : produit le modèle des besoins focalisé sur le métier des utilisateurs. Elle qualifie, au plus tôt le risque de produire un système inadapté aux utilisateurs.
 - L'étape d'analyse consiste à étudier précisément les spécifications fonctionnelles de manière à obtenir une idée de ce que va réaliser le système en terme de métier.
- *La branche droite (technique)* : capitalise un savoir-faire technique pour le système indépendamment des fonctions à réaliser. Cette branche comporte les étapes suivantes :
 - La capture des besoins techniques qui spécifie les contraintes, les besoins non fonctionnels et les choix conditionnant la conception du système.
 - La conception générique consiste à construire l'architecture technique du système qui doit être la moins dépendante possible des aspects fonctionnels. Cela revient à construire le squelette du système en décrivant les composants nécessaires et leurs interactions. Cette étape se concrétise par la production d'un prototype.
- *La branche du milieu* : à l'issue des évolutions du modèle fonctionnel et de l'architecture technique, la réalisation du système consiste à fusionner les résultats des deux branches. Cette branche comporte les étapes suivantes :
 - La conception préliminaire qui consiste à intégrer le modèle d'analyse dans l'architecture technique.

- La conception détaillée qui définit comment réaliser chaque composant.
- Le codage et les tests des composants.

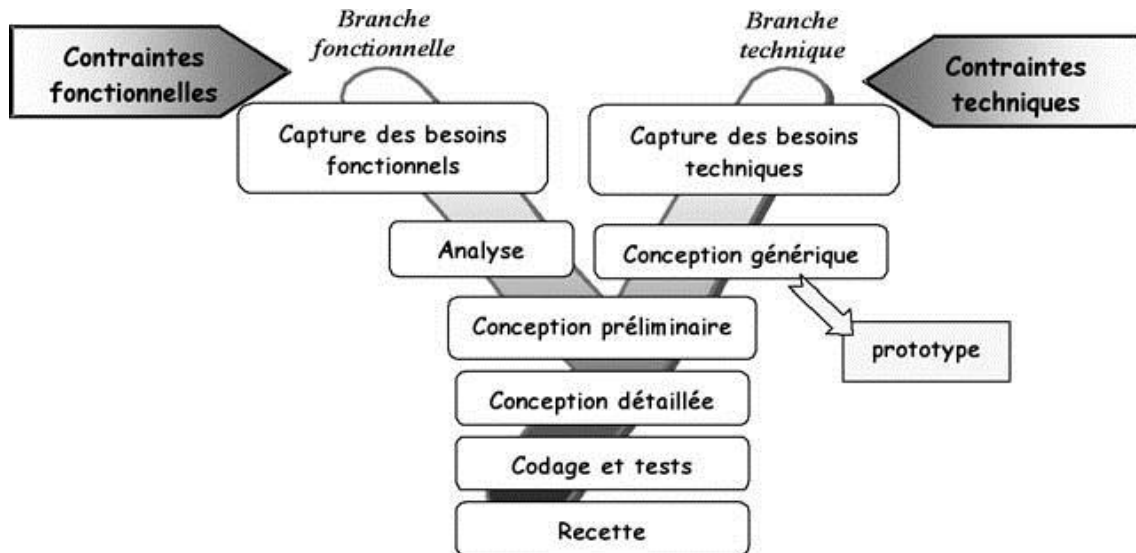


Figure.3 : Processus 2TUP

2. Cahier de charge

Après un premier tour d'horizon des besoins exprimés par la société nous avons établi le cahier des charges ci-après qui consiste en la réalisation d'une application web qui permet l'implémentation des tâches suivante :

- Gestion des offres : Le système doit permettre
 - Au marchand
 - D'ajouter les offres : depuis son espace personnelle le marchand peut ajouter des nouvelles offres. Les offres ajoutées ne sont pas automatiquement validées.
 - Éditer les informations d'une offre.
 - Supprimer une offre.
 - Valider les offres : envoyer les offres pour qu'elles soient approuvées par l'administrateur et par suite visibles dans le site.

- A l'administrateur
 - Approuver les offres : pour qu'une offre soit visible dans le site, l'administrateur doit l'approuver et ceci après la vérification de ses informations.
 - Désapprouver les offres : si une offre contient un détail erroné ou mal saisi, l'administrateur rejettera cette offre.
- Gestion des sociétés : le système doit permettre à l'administrateur :
 - Valider l'inscription d'une société. En effet, les sociétés nouvellement inscrites sont mises en attente jusqu'à ce que l'administrateur les valide.
 - Rejeter une société : si une société n'existe pas ou elle n'est pas inscrite dans le registre commercial, l'administrateur la rejette.
- Gestion des catégories : le système doit permettre à l'administrateur de
 - Ajouter des nouvelles catégories et des sous-catégories : les différents produits sont regroupés dans des catégories
 - Modifier l'intitulé d'une catégorie ou d'une sous-catégorie.
- Gestion des membres : un membre inscrit doit pouvoir :
 - Ajouter une offre à sa liste des offres préférées.
 - Supprimer une offre de sa liste des offres préférées.
 - Recevoir les dernières actualités du site.
 - Modifier son profil.

3. Modélisation du contexte

3.1- Identification des acteurs

Un acteur représente l'abstraction d'un rôle joué par des entités externes (utilisateur, dispositif matériel ou autre système) qui interagissent directement avec le système étudié.

Un acteur peut consulter et/ou modifier directement l'état du système, en émettant et/ou en recevant des messages éventuellement porteurs de données.

Notre système est représenté par 4 acteurs principaux :

- **L'internaute** : c'est n'importe quelle personne qui peut accéder à notre application. Les tâches qu'un internaute peut effectuer se résument dans la recherche par différents critères, l'inscription dans le site et la consultation des produits disponibles.
- **Le membre** : un membre est un internaute inscrit dans le système. En plus des tâches d'un internaute, un membre peut ajouter une offre à sa liste des offres préférées et par suite gérer cette liste. Il peut aussi modifier les informations reliées de son compte.
- **Le marchand** : est une personne qui représente la société inscrite dans le comparateur de prix et qui veut bénéficier des services offerts par le système. Le marchand a pour mission l'ajout, la modification et la suppression des offres, comme il peut aussi gérer son compte.
- **L'administrateur** : S'occupe de la gestion des nouvelles inscriptions des sociétés ainsi que de la gestion des offres.

3.2- Identifications des messages

Un message représente la spécification d'une communication unidirectionnelle entre objets qui transportent de l'information avec l'intention de déclencher une activité chez le récepteur.

Un message est normalement associé à deux occurrences d'événements : un événement d'envoi et un événement de réception.

Cette notion de message est également tout à fait applicable pour décrire les interactions de plus haut niveau entre les acteurs et le système.

✓ **Message émis par le système :**

- Gestion des offres :
 - Informations et caractéristiques d'une offre.
 - Liste des offres disponibles.
 - Liste des offres non approuvées.
 - Liste des offres d'un marchand.

- Gestion des sociétés :
 - Informations d'une société.
 - Liste des sociétés en attente d'approbation.
 - Liste de toutes les sociétés.

- Gestion des membres :
 - Informations d'un membre.
 - Liste des membres inscrits.

✓ **Message reçus par le système :**

- Gestion des offres :
 - Ajout d'une offre.
 - Modification d'une offre.
 - Suppression d'une offre.
 - Approbation d'une offre.

- Gestion des sociétés :
 - Ajout d'une société
 - Modification d'une société.
 - Validation d'une société.
 - Suppression d'une société.

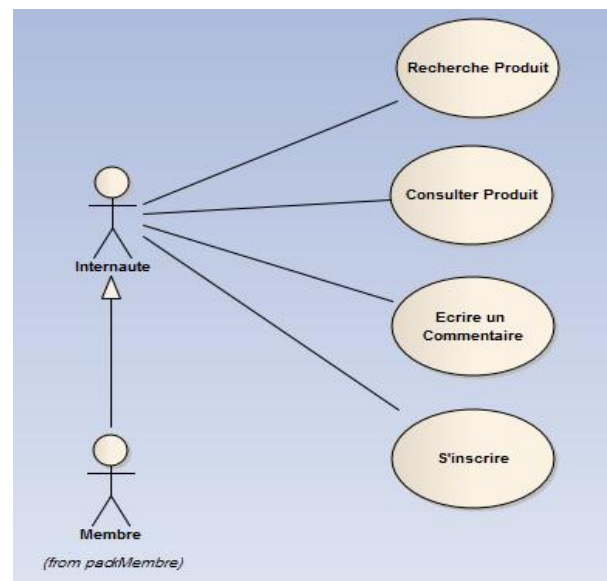
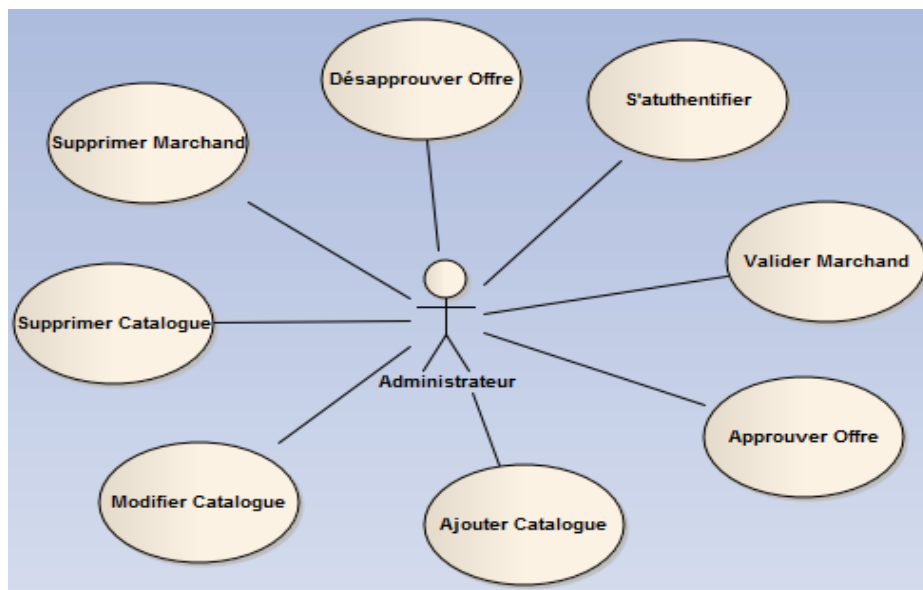
- Gestion des membres :
 - Ajout d'un membre
 - Suppression d'un membre

4. Capture des besoins fonctionnelles

4.1- Identification des cas d'utilisation

Un cas d'utilisation (use case) représente un ensemble de séquences d'actions réalisées par le système et produisant un résultat observable intéressant pour un acteur particulier.

Un cas d'utilisation modélise un service rendu par le système. Il exprime les interactions acteurs/système et apporte une valeur ajoutée « notable » à l'acteur concerné.



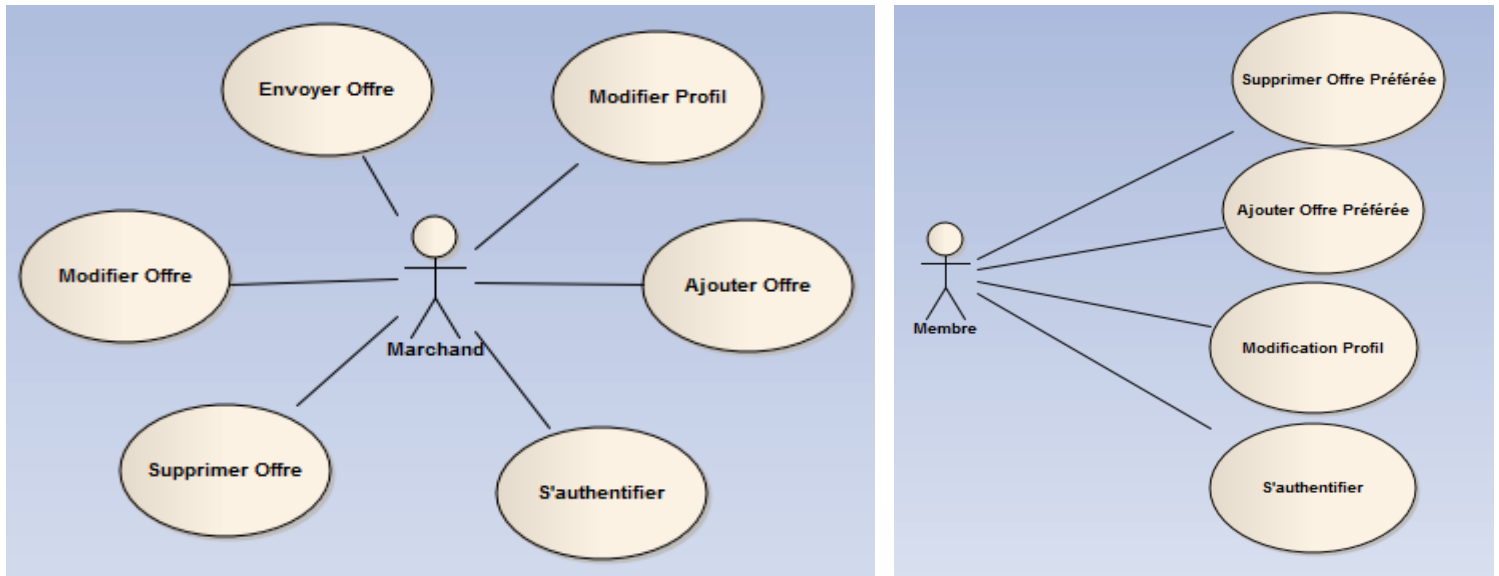


Figure.4 : Diagramme de cas d'utilisations

✓ Administrateur

Acteur principal	Nom du cas d'utilisation	Messages émis/reçus par le système
Administrateur	Valider Marchand	Reçus : - Numéro du marchand. Émis : - Liste des marchands.
	Supprimer Marchand	Reçus : - Numéro du marchand. Émis : - Liste des marchands.
	Modifier Marchand	Reçus : - Numéro du marchand. - Nouvelles informations. Émis : - Liste des marchands.
	Approuver Offre	Reçus : - Numéro de l'offre. Émis : - Liste des offres en attente de validation.

	Désapprouver Offre	Reçus : - Numéro de l'offre. Émis : - Liste des offres en attente de validation.
	Ajouter Catégorie	Reçus : - Nouvelles informations. Émis : - Liste des catégories.
	Modifier Catégorie	Reçus : - Numéro de la catégorie. - Nouvelles informations. Émis : - Liste des catégories.

✓ Marchand :

Acteur principal	Nom du cas d'utilisation	Messages émis/reçus par le système
Marchand	Ajouter Offre	Reçus : - Informations sur l'offre. Émis : - Liste des offres du marchand.
	Modifier Offre	Reçus : - Numéro de l'offre - Nouvelles informations. Émis : - Liste des offres du marchand.
	Supprimer Offre	Reçus : - Numéro de l'offre Émis : - Liste des offres du marchand.
	Envoyer Offre	Reçus : - Numéro de l'offre. Émis : - Liste des offres du marchand.

	Gérer Compte	Reçus : <ul style="list-style-type: none"> - Numéro du marchand - Nouvelles informations. Émis : <ul style="list-style-type: none"> - Les informations du marchand.
--	--------------	--

✓ Membre

Acteur principal	Nom du cas d'utilisation	Messages émis/reçus par le système
Membre	Ajouter offre préférée	Reçus : <ul style="list-style-type: none"> - Numéro de l'offre Émis : <ul style="list-style-type: none"> - Liste des offres préférées.
	Supprimer offre préférée	Reçus : <ul style="list-style-type: none"> - Numéro de l'offre Émis : <ul style="list-style-type: none"> Liste des offres préférées.
	Modifier profil	Reçus : <ul style="list-style-type: none"> - Numéro du membre. - Nouvelles informations. Émis : <ul style="list-style-type: none"> - Les informations du membre.

✓ Internaute

Acteur principal	Nom du cas d'utilisation	Messages émis/reçus par le système
Internaute	Rechercher Produit	Reçus : <ul style="list-style-type: none"> - Critère de recherche Émis : <ul style="list-style-type: none"> - liste des offres qui répondent aux critères.
	Consulter offre	Reçus : <ul style="list-style-type: none"> - Numéro de l'offre Émis : <ul style="list-style-type: none"> - Les informations de l'offre.

4.2- Description détaillée des cas d'utilisation

La description détaillée consiste à recenser toutes les interactions entre le système et les acteurs de façon textuelle. Le cas d'utilisation doit avoir un début et une fin clairement identifiés. Il doit préciser quand ont lieu les interactions entre acteurs et système, et quels sont les messages échangés. Il faut également préciser les variantes possibles, telles que les différents cas nominaux, les cas alternatifs, les cas d'erreurs, tout en essayant d'ordonner séquentiellement les descriptions, afin d'améliorer leur lisibilité. Chaque unité de description de séquences d'actions est appelée enchaînement. Un scénario représente une succession particulière d'enchaînements, qui s'exécute du début à la fin du cas d'utilisation.

Dans cette partie nous allons nous limiter au cas d'utilisation « Gestion des offres ». La description détaillée des autres cas d'utilisation sera abordé plus loin dans l'annexe.

Sommaire d'identification :

Titre : Ajouter Offre

But : Ajouter une offre dans le compte du marchand.

Résumé : Permettre au marchand d'ajouter une offre concernant un produit.

Acteur : Marchand

Description des enchaînements :

Pré-conditions : Le marchand est authentifié.

Post-conditions : L'offre est ajoutée.

Enchaînement nominal : Ce cas d'utilisation commence lorsque le marchand demande au système d'ajouter une offre.

- 1) Le marchand demande l'ajout d'une nouvelle offre.
- 2) Le système affiche le formulaire d'ajout.
- 3) Le marchand remplit tous les champs et valide.
- 4) Le système vérifie les champs s'ils sont corrects.
- 5) Le système enregistre la nouvelle offre.

Enchaînements alternatifs :

A1 : Champs invalides

L'enchaînement A1 démarre au point (4) du scénario nominal

- 5) Le système indique à l'utilisateur qu'un champ n'est pas valide

Le scénario nominal reprend au point (3).

Sommaire d'identification :

Titre : Modifier Offre

But : Modifier les détails d'une offre.

Résumé : Donner la possibilité au marchand de modifier les informations d'une offre.

Acteur : Marchand

Description des enchaînements :

Pré-conditions : Le marchand est authentifié.

Post-conditions : L'offre est modifiée.

Enchaînement nominal : Ce cas d'utilisation commence lorsque le marchand demande au système de modifier une offre.

- 1) Le marchand demande la modification d'une offre.
- 2) Le système affiche le formulaire de modification qui contient les informations de l'offre.
- 3) Le marchand modifie les informations désirées et valide.
- 4) Le système vérifie les champs s'ils sont corrects.
- 5) Le système enregistre les modifications.
- 6) Le système redirige le marchand vers la liste des offres.

Sommaire d'identification :

Titre : Supprimer Offre

But : Supprimer une offre.

Résumé : Donner la possibilité au marchand de supprimer une offre avant sa validation.

Acteur : Marchand

Description des enchaînements :

Pré-conditions : Le marchand est authentifié.

Post-conditions : L'offre est supprimée.

Enchaînement nominal : Ce cas d'utilisation commence lorsque le marchand demande au système de modifier une offre.

- 1) Le marchand demande la suppression d'une offre.
- 2) Le système affiche un message pour confirmer la suppression.

- 3) Le marchand valide la suppression.
- 4) Le système supprime l'offre.
- 5) Le système redirige le marchand vers la liste des offres.

Enchaînements d'exception :

E1 : Le marchand annule la suppression

L'enchaînement E1 démarre au point (2) du scénario nominal

- 3) Le système revient vers la liste des offres.

Sommaire d'identification :

Titre : Envoyer Offre

But : Envoyer une offre.

Résumé : L'offre n'est pas validée automatiquement après son ajout, et donc le système doit permettre à l'utilisateur de valider les offres choisies.

Acteur : Marchand

Description des enchaînements :

Pré-conditions : - Le marchand est authentifié.

- Au moins une offre existe dans le compte.

Post-conditions : L'offre est validée.

Enchaînement nominal : Ce cas d'utilisation commence lorsque le marchand demande au système d'envoyer une offre.

- 1) Le marchand choisit les offres à envoyer.
- 2) Le marchand demande l'envoi des offres.
- 3) Le système vérifie qu'au moins une offre est choisie.
- 4) Le système enregistre les offres validées.
- 5) Le système redirige le marchand vers la liste des offres.

Enchaînements alternatifs :

A1 : Aucune offre choisie

L'enchaînement A1 démarre au point (3) du scénario nominal

- 4) Le système indique à l'utilisateur qu'il doit choisir au moins une offre.

5. Analyse

L'analyse a pour objectif de détailler les spécifications réalisées lors de la capture des besoins fonctionnels. Elle consiste à modéliser ces spécifications selon les vues statique et dynamique.

5.1- Modèle statique

A partir de l'étude préliminaire et de la description détaillée des cas d'utilisation nous avons élaboré le diagramme de classes suivant :

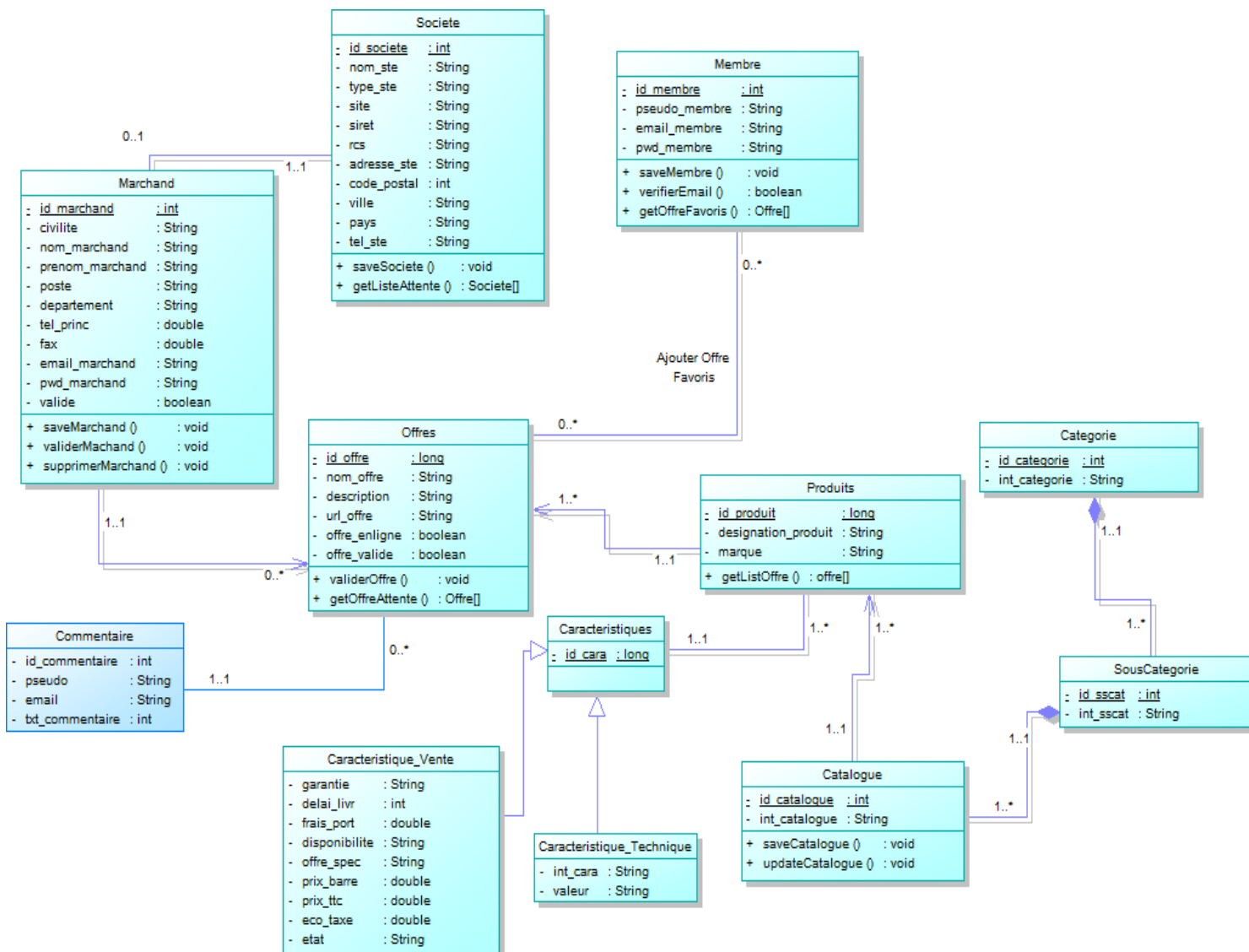


Figure.5 : Diagramme de classes

• **Règles de gestion :**

- Une société est représentée par un et un seul marchand.
- Une catégorie de produits contient une ou plusieurs sous-catégories de produits.
- Une sous-catégorie de produits contient plusieurs catalogues.
- Un produit appartient à un catalogue.
- Tout internaute peut commenter toute offre.

5.2- Modèle dynamique

Le modèle dynamique est une vision microscopique du fonctionnement du système. Il sert à mettre en évidence les interactions entre les objets du système. Il intervient après la définition du modèle statique.

Nous allons présenter le diagramme d'activités de la gestion des offres ainsi les diagrammes de séquence pour les cas d'utilisation : *Ajouter Offre*, *Modifier Offre*, *Supprimer Offre*

• **Diagramme d'activités :** Gestion des offres



Figure.6 : Diagramme d'activité du cas d'utilisation : Gestion Offre

Diagramme de séquence du cas d'utilisation Ajouter Offre

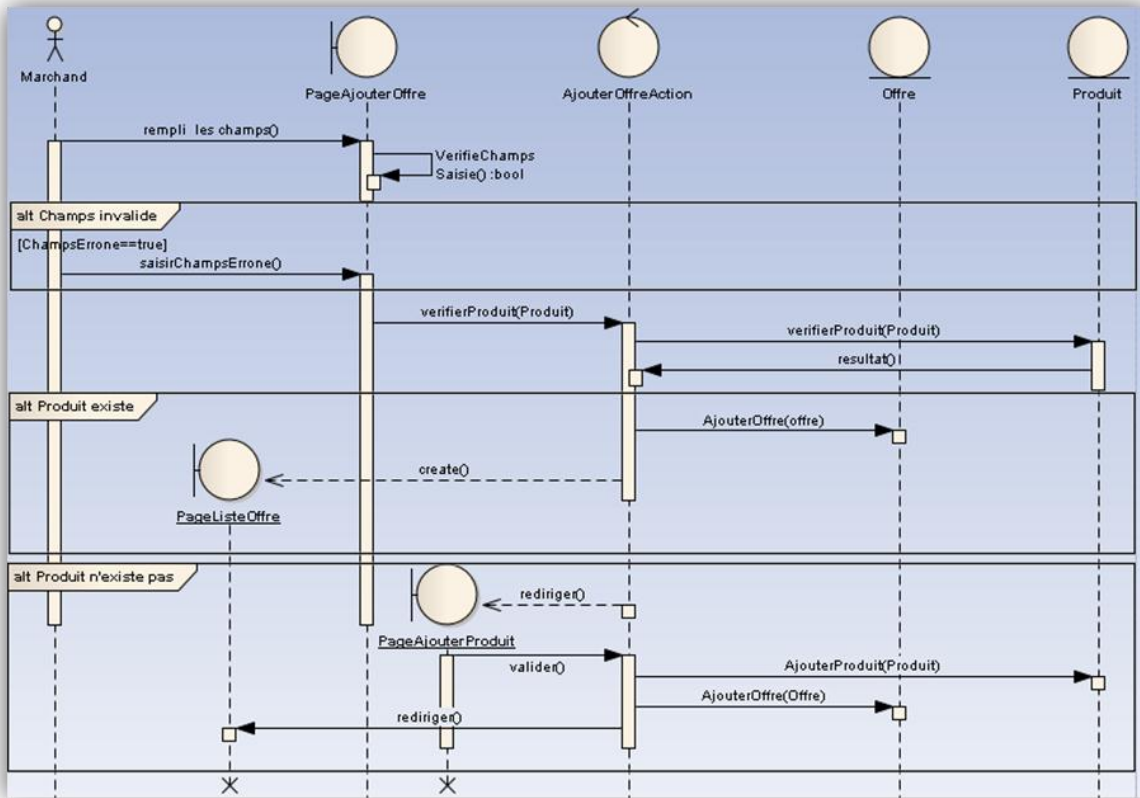


Figure.7 : Diagramme de séquence pour le cas d'utilisation Ajouter Offre

Diagramme de séquence du cas d'utilisation Modifier Offre

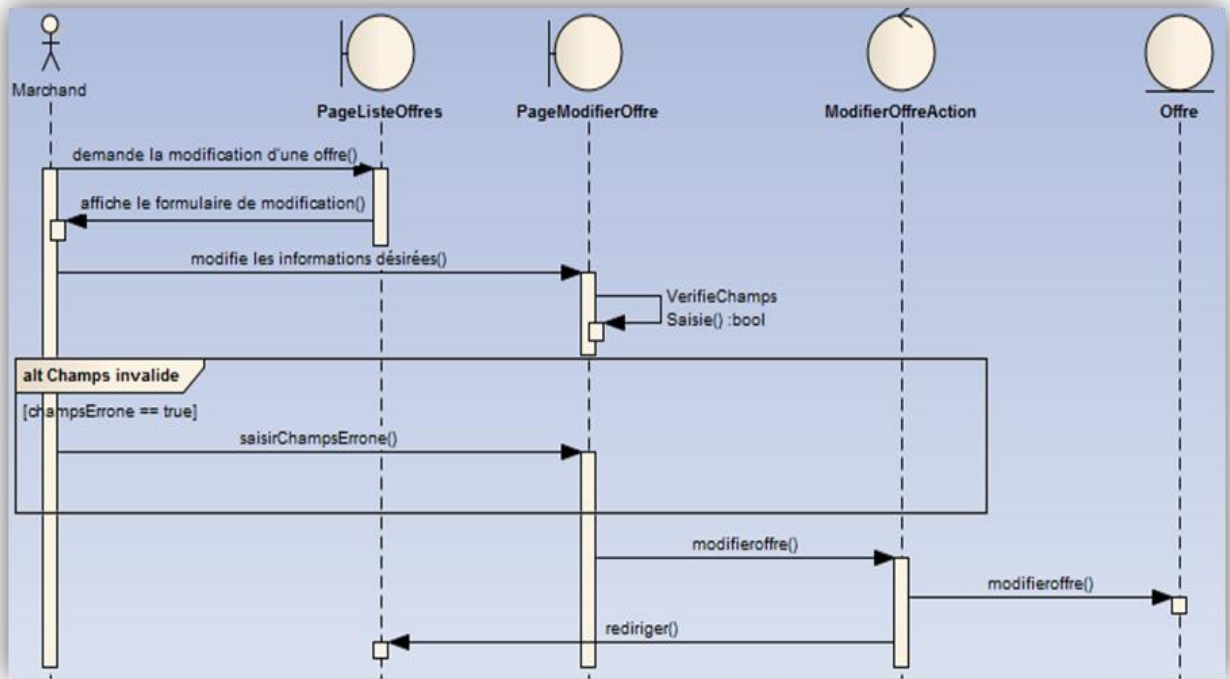


Figure.8 : Diagramme de séquence pour le cas d'utilisation Modifier Offre

Diagramme de séquence du cas d'utilisation Supprimer Offre

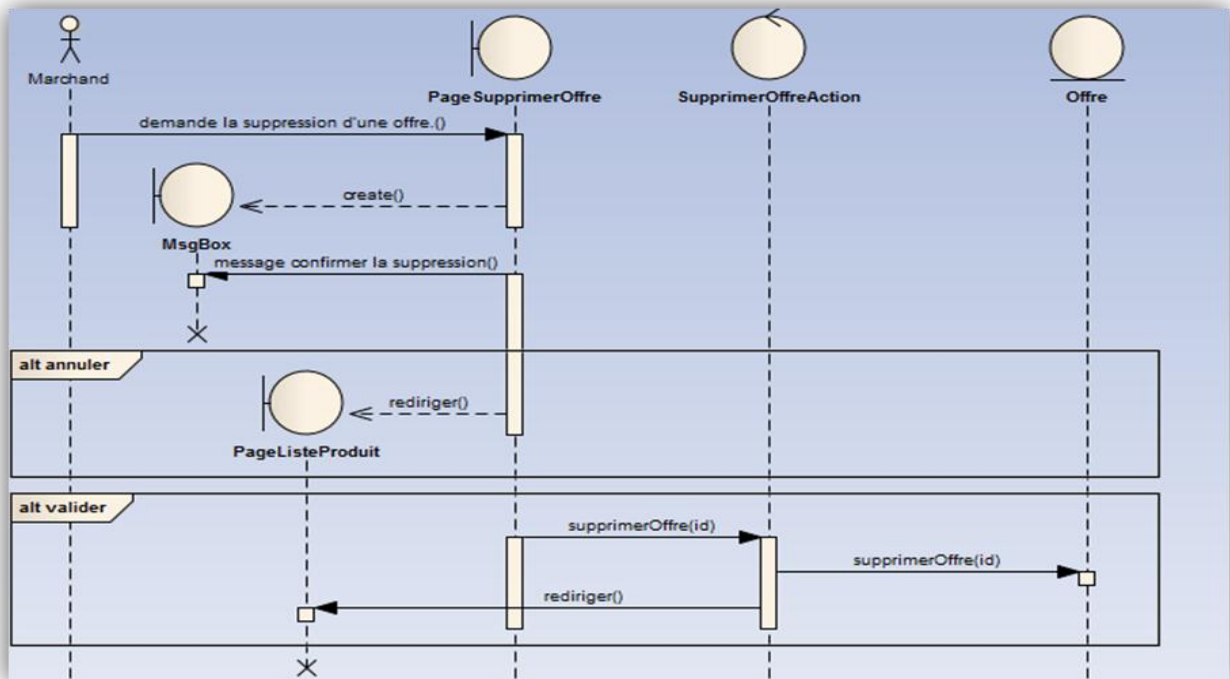


Figure.9 : Diagramme de séquence pour le cas d'utilisation Supprimer Offre

Diagramme de séquence du cas d'utilisation Valider Offre

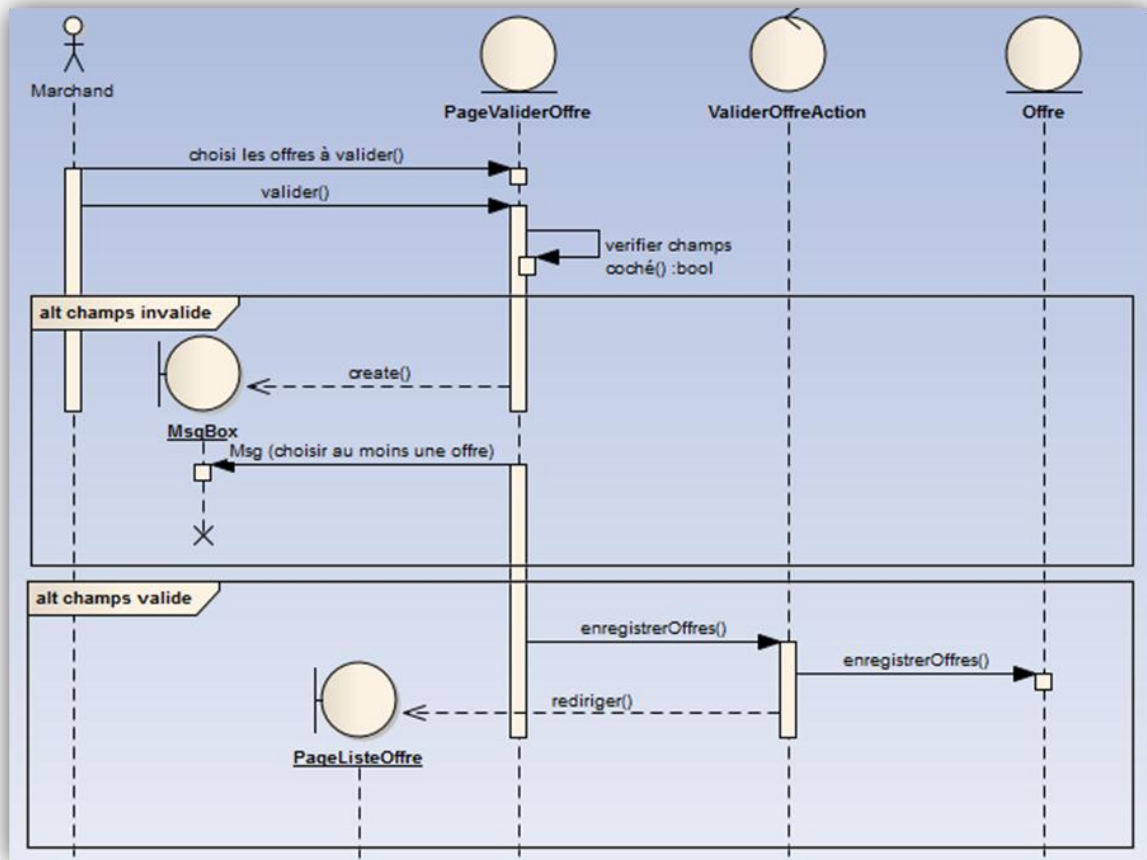


Figure.10 : Diagramme de séquence pour le cas d'utilisation Valider Offre

Chapitre III

Étude technique du projet

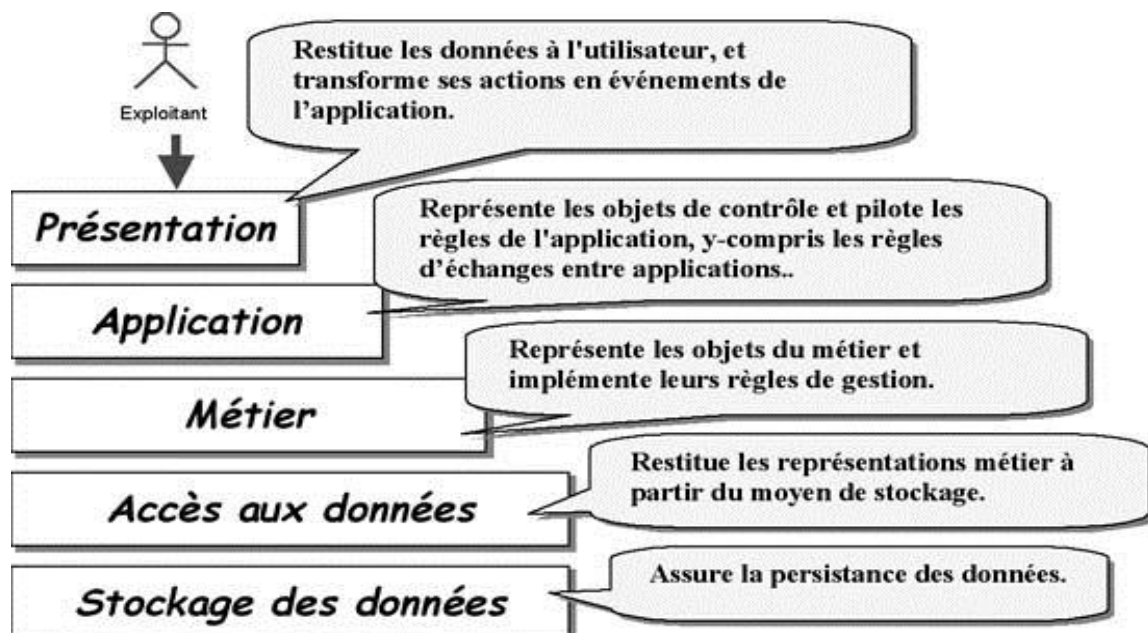
Dans cette partie nous allons présenter l'étude technique de notre projet. L'architecture de notre système sera présentée en premier lieu suivie des outils utilisés lors de la réalisation.

1. Architecture et structure de l'application

L'architecture d'une application définit la répartition des composants du système informatique à réaliser sur l'ensemble du matériel ainsi que leurs relations. Dans notre cas on a opté pour une architecture 3-tiers.

Pour avoir une application robuste et évolutive, nous avons utilisé le principe des couches où chaque couche représente un ensemble de spécifications ou de réalisations qui expriment un ensemble de responsabilités techniques et homogènes pour un système logiciel.

Structuration en 5 couches :



Dans notre projet :

- **Couche Présentation** : Représente l'interface de l'utilisateur, cette partie a été réalisée par le langage web HTML & CSS.
- **Couche Application** : Regroupe les classes des traitements et elle est réalisée à l'aide du composant *Zend_Controller_Action* de Zend Framework.
- **Couche Accès aux données** : couche d'accès à la base de données permettant l'interrogation de celle-ci. Réalisée à travers le composant *Zend_Db_Table_Abstract*.

2. Outils technologiques

✓ **PHP 5.3.5**

PHP (sigle de PHP: Hypertext Preprocessor) est un langage de scripts libre principalement utilisé pour produire des pages Web dynamiques via un serveur HTTP, mais pouvant également fonctionner comme n'importe quel langage interprété de façon locale, en exécutant les programmes en ligne de commande. PHP est un langage impératif disposant depuis la version 5 de fonctionnalités de modèle objet complètes. En raison de la richesse de sa bibliothèque, on désigne parfois PHP comme une plateforme plus qu'un simple langage.

✓ **Mysql 5.5.8**

MySQL est un système de gestion de base de données (SGBD). Selon le type d'application, sa licence est libre ou propriétaire. Il fait partie des logiciels de gestion de base de données les plus utilisés au monde, autant par le grand public (applications web principalement) que par des professionnels, en concurrence avec Oracle et Microsoft SQL Server.

✓ **jQuery**

jQuery est une bibliothèque JavaScript libre qui porte sur l'interaction entre JavaScript (comprenant AJAX) et HTML, et a pour but de simplifier des commandes communes de JavaScript.

Le framework contient les fonctionnalités suivantes :

- Parcours et modification du DOM (y compris le support des sélecteurs CSS 1 à 3 et un support basique de XPath).
- Événements.
- Effets et animations.
- Manipulations des feuilles de style en cascade (ajout/suppression des classes, d'attributs...).

✓ **Ajax**

En informatique, et plus particulièrement en architecture informatique, Ajax (acronyme de Asynchronous Javascript and XML) est une manière de construire des applications Web et des sites web dynamiques basés sur diverses technologies Web ajoutées aux navigateurs dans les années 1990.

- DOM et JavaScript sont utilisés pour modifier l'information présentée dans le navigateur par programmation.
- L'objet XMLHttpRequest est utilisé pour dialoguer de manière asynchrone avec le serveur Web.
- la notation XML est utilisée pour structurer les informations transmises entre le serveur Web et le navigateur.

✓ Zend Framework

Zend Framework (ZF) est un framework open-source destiné aux développements d'applications web et de services web avec PHP5. Le Zend Framework est construit en utilisant 100% de code orienté-objet. Il offre une robuste et performante implémentation du motif MVC, une abstraction de base de données simple d'utilisation, et un composant de formulaire qui implémente un rendu HTML, la validation et le filtrage des données, ainsi les développeurs peuvent consolider toutes ces opérations en utilisant une interface orienté-objet facile d'utilisation.

- **Configuration** : Le Zend Framework lui-même n'a pas besoin de configuration, mais l'application oui. La configuration par défaut est placée sous « *application/configs/application.ini* » et contient des directives de base pour régler l'environnement PHP (par exemple, activer ou désactiver le rapport d'erreurs).

```
1. ; application/configs/application.ini
2.
3. [production]
4. phpSettings.display_startup_errors = 0
5. phpSettings.display_errors = 0
6. includePaths.library = APPLICATION_PATH "../library"
7. bootstrap.path = APPLICATION_PATH "/Bootstrap.php"
8. bootstrap.class = "Bootstrap"
9. appnamespace = "Application"
10. resources.frontController.controllerDirectory = APPLICATION_PATH
    "/controllers"
11. resources.frontController.params.displayExceptions = 0
12.
13. [staging : production]
14.
15. [testing : production]
16. phpSettings.display_startup_errors = 1
17. phpSettings.display_errors = 1
18.
19. [development : production]
20. phpSettings.display_startup_errors = 1
21. phpSettings.display_errors = 1
```

- **Contrôleur d'action :**

Les contrôleurs d'action contiennent la logique 'utile' de l'application et font correspondre les requêtes aux bons modèles et aux bonnes vues.

Un contrôleur d'action devrait posséder une ou plusieurs méthodes se terminant par "Action"; ces méthodes sont interrogées via le web. Par défaut, les URLs de Zend Framework suivent le schéma « `/controller/action` », où "controller" correspond à la classe de contrôleur (sans le suffixe "Controller") et "action" correspond à la méthode d'action (sans le suffixe "Action").

- **Vue :**

Les vues dans Zend Framework sont des fragments écrites en PHP classique et qui sont intégrées dans des pages appelées « layout ». Les scripts de vues sont placés sous application « `/views/scripts/` », ils sont référencés plus tard dans les contrôleurs.

- **Le composant Zend_Db_Table :**

La classe Zend_Db_Table est une interface orientée objet vers les tables d'une base de données. Elle fournit des méthodes pour la gestion de la plupart des opérations concernant une table.

Zend_Db permet de s'affranchir du code SQL, mais les fonctionnalités de mapping relationnel objet (ORM*) sont encore plus utiles. Le Zend Framework propose cette l'ORM grâce à la classe Zend_Db_Table_Abstract.

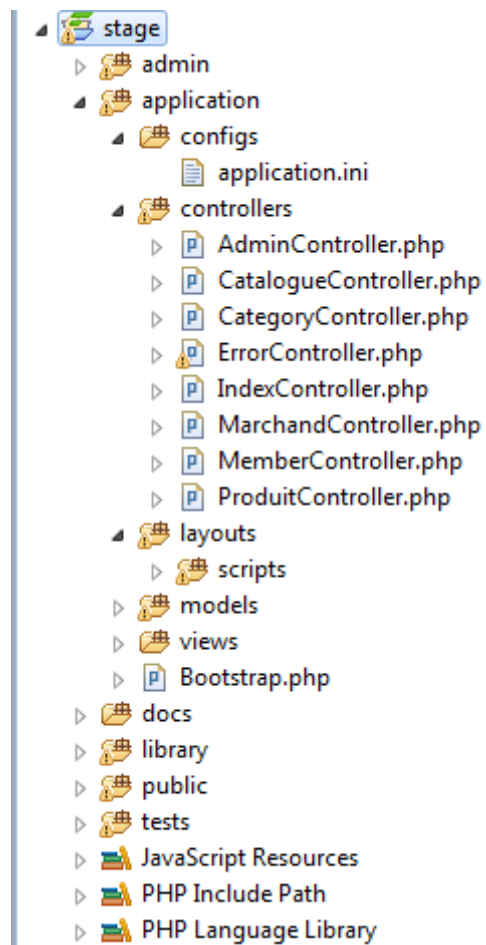
(*) : Un mapping objet-relationnel (en anglais object-relational mapping ou ORM) est une technique de programmation informatique qui crée l'illusion d'une base de données orientée objet à partir d'une base de données relationnelle en définissant des correspondances entre cette base de données et les objets du langage utilisé.

Chapitre IV

Présentation de l'application

Dans cette dernière partie nous allons présenter des captures d'écran de notre application et des classes réalisées

- **Hiérarchie de l'application :**



1. Ecran d'accueil

L'image ci-dessous représente la page d'accueil de notre application. A gauche se place un menu qui regroupe les différentes catégorie de produits à gérer par le comparateur de prix. Notre tâche était de développer la partie concernant la catégorie : informatique.



Ecran.1 : Page d'accueil du comparateur du prix

2. Fonctionnalités de l'internaute

Une fois on a cliqué sur la catégorie Informatique, une nouvelle page s'ouvre et nous affiche l'ensemble des sous-catégories et des catalogues que regroupe la catégorie choisie :

- Sous-catégorie : Ordinateur de bureau, Ordinateur portable, Périphérique...
- Catalogue : PC Bureau, PC Mac, Imprimante...



Ecran.2 : La liste des sous-catégories et des catalogues

Après avoir choisi un catalogue, la liste des offres que contient ce catalogue s'affiche ainsi que l'ensemble de ses informations

Vous avez choisi de comparer : iPad 2 16gb

2 marchands
150 € - 3999 €

Donnez votre avis sur ce produit !

Comparer Caractéristiques Avis

Affichage comprimé de 9,7 pouces, intégré 16 Go de mémoire, wi-fi, bluetooth, appareil photo à double écran tactile- ...

150€
Voir

Etat : Occasion
Disponibilité : non
Délais de livraison : 3 jours

Google + Livraison : 10 €

Affichage comprimé de 9,7 pouces, intégré 16 Go de mémoire, wi-fi, bluetooth, appareil photo à double écran tactile- ...

3999€
Voir

Etat : Neuf
Disponibilité : Non Disponible
Délais de livraison : 15 jours

Twitter + Livraison : 0 €

Ecran.3 : Liste des offres d'un produit

3. Fonctionnalités du membre

A travers cette interface le membre peut soit se connecter on utilisant l'email et le mot de passe fournis lors de son inscription ou bien créer un nouveau compte.

Créer un compte

Votre adresse email : *

Choisissez un pseudonyme : *
pour participer au Forum Conso

Choisissez un mot de passe : *

Confirmez votre mot de passe : *

Souhaitez-vous recevoir :

- Les newsletters ? Oui Non

*Champs obligatoires

Créer un compte

Déjà membre ? Connectez-vous

Email :

Mot de passe :

Connectez vous

Ecran.4 : Fenêtre de connexion et d'inscription d'un membre

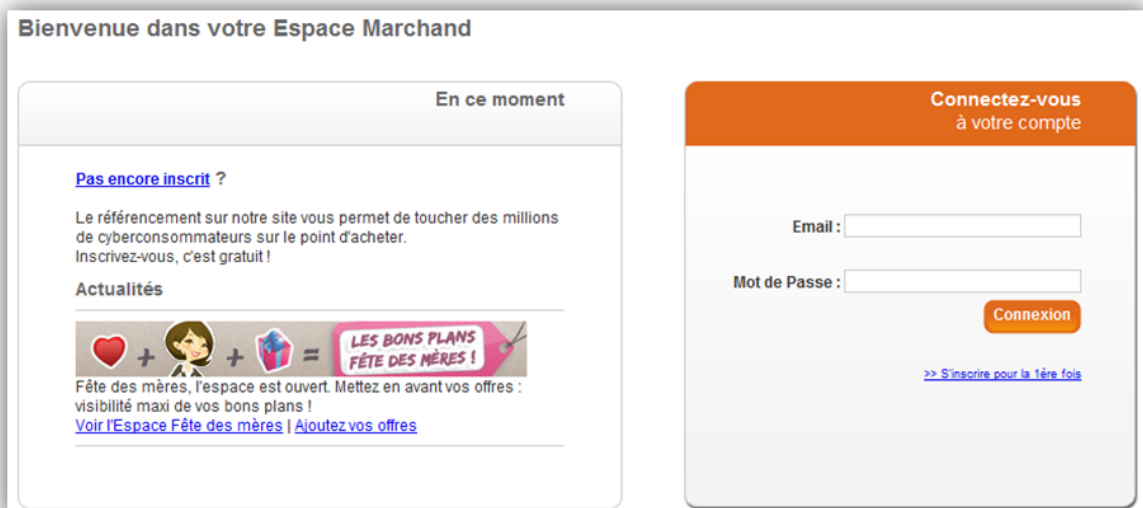
Une fois connecté, le membre peut visualiser sa liste des offres préférées.



Ecran.5 : Liste des offres préférées d'un membre

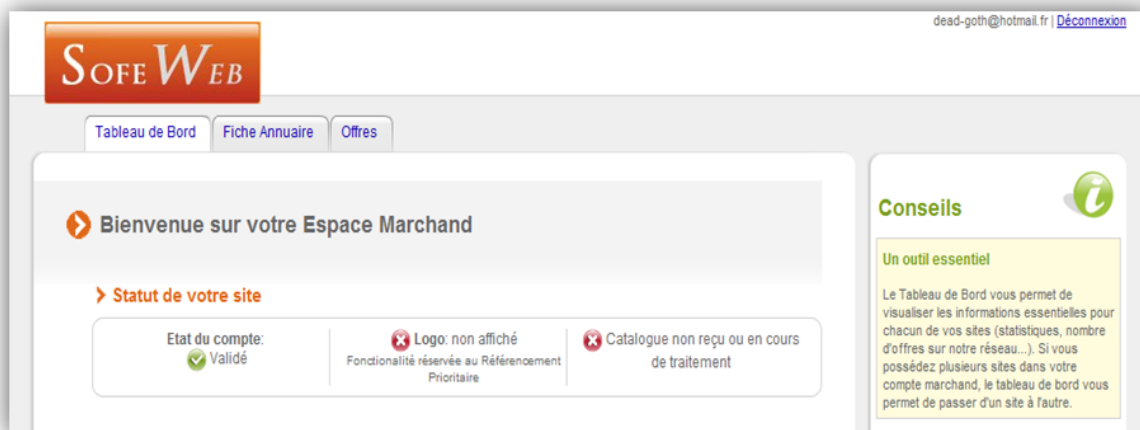
4. Fonctionnalités du marchand

L'interface qui permet au marchand de se connecter à son compte et par la suite gérer ses offres.



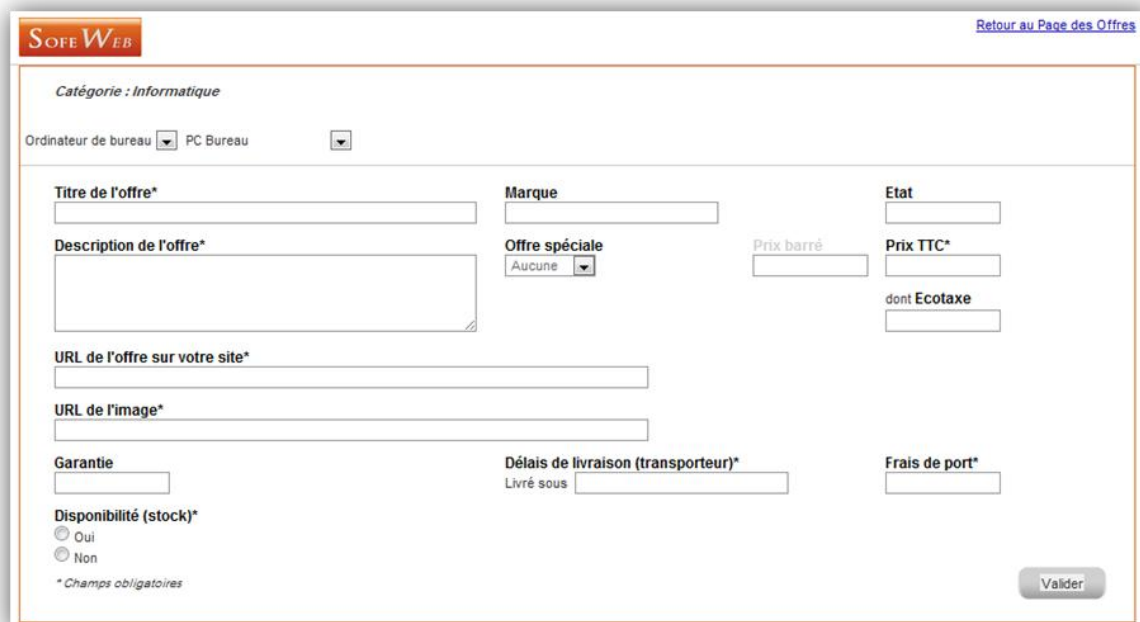
Ecran.6 : Fenêtre de connexion du marchand

Depuis son espace personnel, le marchand peut modifier les informations personnelles et celles de la société et aussi gérer la liste de ses offres



Ecran.7 : Espace personnel du marchand

L'interface qui permet au marchand d'ajouter une nouvelle offre, dans laquelle le marchand peut saisir toutes les informations concernant la nouvelle offre. En particulier les caractéristiques techniques et de vente du produit qui doivent être validées par la suite par l'administrateur.



Ecran.8 : Fenêtre d'ajout d'une offre

Si l'offre concerne un produit qui n'existe pas dans la base de données une nouvelle page s'affichera pour saisir les caractéristiques de ce produit. Ce dernier n'est pas pris en charge par le comparateur de prix jusqu'à sa validation par l'administrateur.

Ecran.9 : Formulaire d'ajout des caractéristiques d'un produit

La liste des offres d'un marchand dans laquelle on peut distinguer les offres validées et non encore validées.

Titre de l'offre	Catégorie	Prix barré	Prix	Marque	Etat	Edition	Sélection
iPad 2 16gb	Informatique	200	150	Apple		Editer Supprimer	<input type="checkbox"/>
iPad 2 16gb wi-fi	Informatique	6000	3999	Apple		Editer Supprimer	<input type="checkbox"/>
MacBook Pro 1	Informatique	200	150	Apple	Non Envoyée	Editer Supprimer	<input type="checkbox"/>
MacBook Pro 2	Informatique	200	150	Apple		Editer Supprimer	<input type="checkbox"/>

Ecran.10 : Liste des offres d'un marchand

5. Fonctionnalités de l'administrateur

L'interface qui permet à l'administrateur de s'authentifier



The image shows a login form titled "ControlPanelLogin". It features a dark background with white and blue text. The form includes two input fields: "Enter your username:" and "Enter your password:". Below the password field is a blue "Submit" button.

Ecran.11 : Formulaire de connexion de l'administrateur

La page d'accueil de l'administrateur. On distingue 3 menus correspondant à la gestion des sociétés, la gestion des offres et la gestion des produits.



Ecran.12 : Page d'accueil de l'administrateur

Cette capture affiche les sociétés qui sont en attente de validation. Il existe deux actions possibles : valider ou supprimer la société.

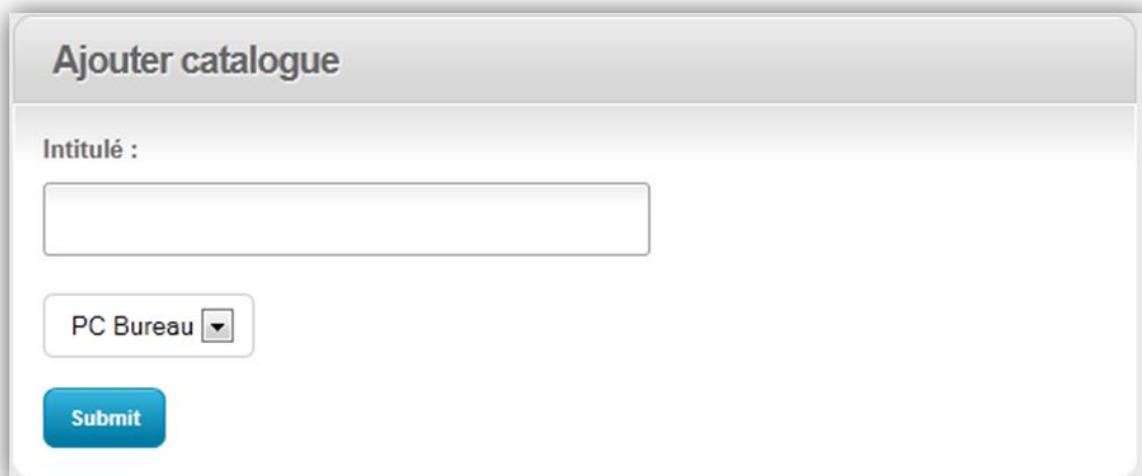


Représentant	Société	Actions
MAFTAH Samir Poste : Marketing / Publicité - Promotion Tél : 212666066606 Email : siori90@hotmail.com	Nom : Pixmania Type : SA SIRET : 72JD2 Tél : 05650505050 Adresse : 72, Bd Mohamed V 60000 Oujda Maroc Site web : www.google.com	 

 Approuver  Supprimer

Ecran.12 : Liste des offres non validées

Le formulaire qui permet à l'administrateur d'ajouter un nouveau catalogue



Ajouter catalogue

Intitulé :

PC Bureau ▼

Submit

Ecran.13 : Formulaire d'ajout d'un nouveau catalogue

Conclusion

Notre stage effectué au sein de Sofeweb Oujda nous a donné non seulement l'opportunité de mettre en pratique nos acquis théoriques, mais aussi il a contribué à l'enrichissement de notre expérience professionnelle et humaine.

En effet, cette expérience au sein du monde professionnel nous a permis de s'intégrer dans un milieu qui ne nous était pas familier jusqu'à maintenant et nous a également initié au sens de la responsabilité, le travail collectif, l'indépendance, la sociabilité et le respect.

Le travail a consisté, dans un premier temps, en l'étude du contexte du projet et l'analyse des processus des différentes tâches que notre système en assure l'exécution. Dans un second temps, on a conçu et développé ces différents processus en utilisant plusieurs technologies. Pour l'analyse la conception et le développement du projet on a adopté le formalisme UML, la démarche 2TUP et le framework : Zend.

Vu la taille importante du travail à faire, la limitation temporaire et la complexité des outils déployés, le système n'est pas encore finalisé mais le projet se prolongera sur la période des vacances pour aboutir à une première version.

REFERENCES

1. Bibliographie :

UML 2 en action De l'analyse des besoins à la conception 4e édition.
Pascal Roques & Franck Vallée
Zend Framework Bien développer en PHP. Julien Pauli & Guillaume
Ponçon (2009) .
Zend Framework in Action. Rob Allen & Nick Lo

2. Webographie :

<http://framework.zend.com/manual/fr/>

<http://akrobat.com/zend-framework-tutorial/>

Annexe

Description détaillée des cas d'utilisation

Description détaillée des cas d'utilisation

- **Rechercher Produit :**

Sommaire d'identification :

Titre : Rechercher un produit

But : Rechercher un produit dans la base de données.

Résumé : Effectuer une recherche selon différents critères saisis par l'internaute

Acteur : Internaute / Membre / Marchand

Description des enchaînements :

Pré-conditions : La base de données est alimentée par des produits.

Enchaînement nominal : Ce cas d'utilisation commence lorsque l'internaute demande au système de chercher un produit.

- 1) L'internaute saisie le mot-clé pour chercher le produit.
- 2) L'internaute valide la recherche.
- 3) Le système récupère les produits qui vérifient les critères.
- 4) Le système affiche les produits.

Enchaînements alternatifs :

A1 : Aucun résultat trouvé

L'enchaînement A1 commence au point (3) du scénario nominal

- 4) Le système indique à l'internaute que le mot-clé saisi ne correspond à aucun produit

Le scénario nominal reprend au point (1)

- **Consulter Offre :**

Sommaire d'identification :

Titre : Consulter une offre

But : Afficher les informations d'une offre

Résumé : Afficher les informations d'une offre que l'internaute à sélectionner depuis la liste des offres ou bien depuis le résultat d'une recherche

Acteur : Internaute / Membre / Marchand

Description des enchaînements :

Pré-conditions : L'offre existe.

Post-conditions : obtenir les informations de l'offre désirée.

Enchaînement nominal : Ce cas d'utilisation commence lorsque l'internaute demande au système d'afficher les informations d'une offre.

- 1) L'internaute sélectionne l'offre.
- 2) Le système récupère les informations relatives à l'offre.
- 3) Le système affiche les informations.

- **S'inscrire :**

Sommaire d'identification :

Titre : S'inscrire

But : Inscription dans le site.

Résumé : Enregistrer un internaute dans le site pour profiter de tous les fonctionnalités.

Acteur : Internaute.

Description des enchaînements :

Pré-conditions : Néant.

Post-conditions : Utilisateur inscrit.

Enchaînement nominal : Ce cas d'utilisation commence lorsque l'internaute demande au système de s'inscrire.

- 1) L'internaute envoie une demande d'inscription.
- 2) Le système affiche le formulaire d'inscription.
- 3) L'internaute remplit les champs et valide.
- 4) Le système vérifie si les champs sont valides.
- 5) Le système vérifie l'existence de l'utilisateur.
- 6) Le système enregistre l'internaute dans la base de données.
- 7) Le système affiche un message confirmant l'inscription.

Enchaînements alternatifs :

A1 : Champs invalides

L'enchaînement A1 démarre au point (4) du scénario nominal

- 5) Le système indique à l'utilisateur qu'un champ n'est pas valide

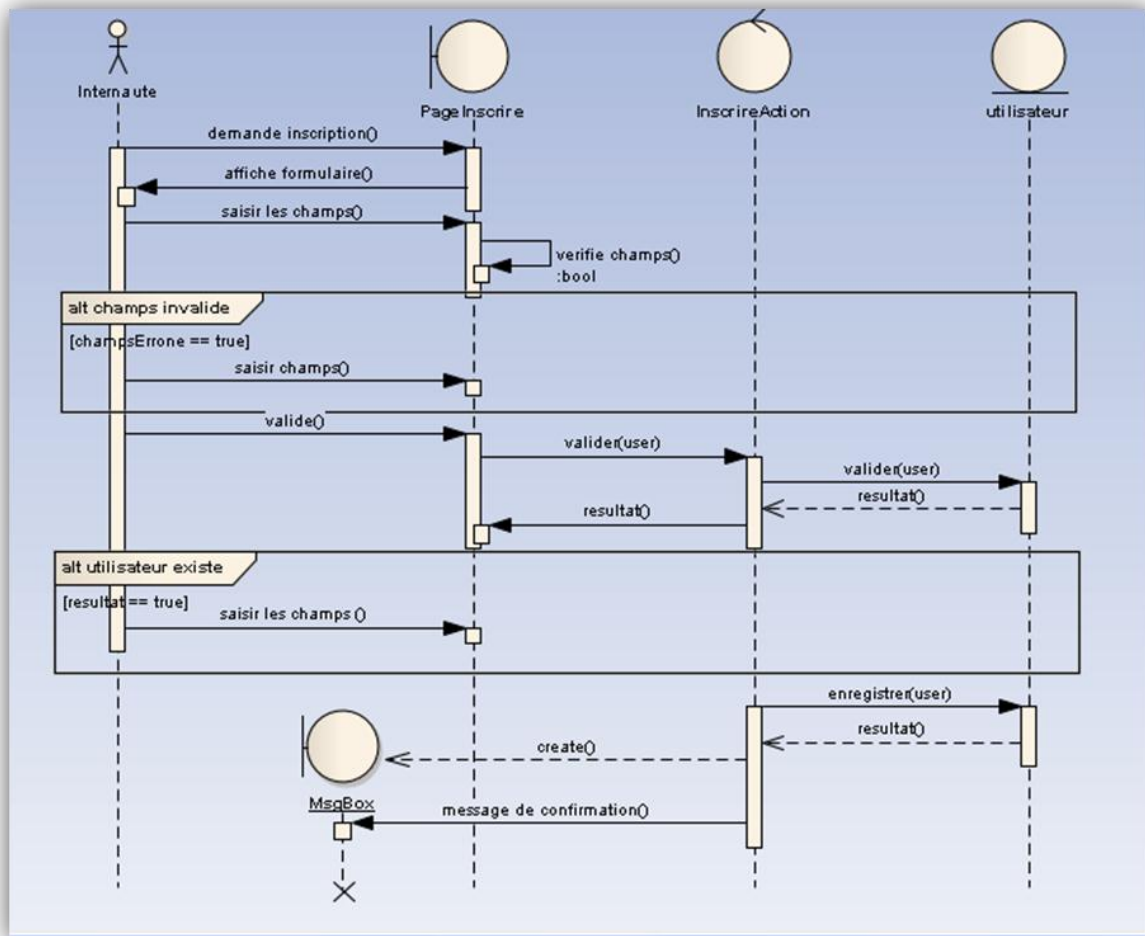
Le scénario nominal reprend au point (3).

A2 : Utilisateur existant

L'enchaînement A2 démarre au point (5) du scénario nominal

- 6) Le système indique à l'utilisateur que l'identifiant saisi existe déjà

Le scénario nominal reprend au point (3)



- **Modifier Profil :**

Sommaire d'identification :

Titre : Modifier Profil
But : Modifier les informations du membre
Résumé : Permettre aux membres de modifier leurs différentes informations.
Acteur : Membre

Description des enchaînements :

Pré-conditions : Le membre est authentifié.
Post-conditions : Profil modifié.

Enchaînement nominal : Ce cas d'utilisation commence lorsque le membre demande au système de gérer son compte.

- 1) Le membre demande la modification de son profil.
- 2) Le système affiche un formulaire qui contient les informations du membre.
- 3) Le membre modifie les informations désirées.
- 4) Le membre valide les modifications.

- 5) Le système vérifie si les champs sont valides.
- 6) Le système enregistre les modifications.

Enchaînements alternatifs :

A1 : Champs invalides

L'enchaînement A1 démarre au point (5) du scénario nominal

- 6) Le système indique à l'utilisateur qu'un champ n'est pas valide

Le scénario nominal reprend au point (3).

- **Authentification :**

Sommaire d'identification :

Titre : Authentification

But : S'authentifier dans le site

Résumé : Permettre aux membres et aux marchands de s'authentifier pour accéder à leurs espaces personnels.

Acteur : Membre / Marchand

Description des enchaînements :

Pré-conditions : L'utilisateur existe.

Post-conditions : L'utilisateur est authentifié.

Enchaînement nominal : Ce cas d'utilisation commence lorsque le membre demande au système de s'authentifier.

- 1) L'utilisateur demande la page d'authentification.
- 2) Le système affiche le formulaire d'authentification.
- 3) L'utilisateur remplit les champs et valide.
- 4) Le système vérifie la combinaison identifiant & mot de passe.
- 5) Le système enregistre l'utilisateur dans la session.

Enchaînements alternatifs :

A1 : Identifiants incorrectes

L'enchaînement A1 démarre au point (4) du scénario nominal

- 5) Le système indique à l'utilisateur que les identifiants saisis n'existent pas.

Le scénario nominal reprend au point (3).

- **Ajouter Offre Préférée :**

Sommaire d'identification :

Titre : Ajouter Offre Préférée

But : Ajouter une offre à la liste des offres préférées.

Résumé : Donner la possibilité aux membres d'enregistrer les offres qui les intéressent

Acteur : Membre

Description des enchaînements :

Pré-conditions : Le membre est authentifié.

Post-conditions : L'offre est ajoutée.

Enchaînement nominal : Ce cas d'utilisation commence lorsque le membre demande au système d'ajouter une offre à la liste des offres préférées.

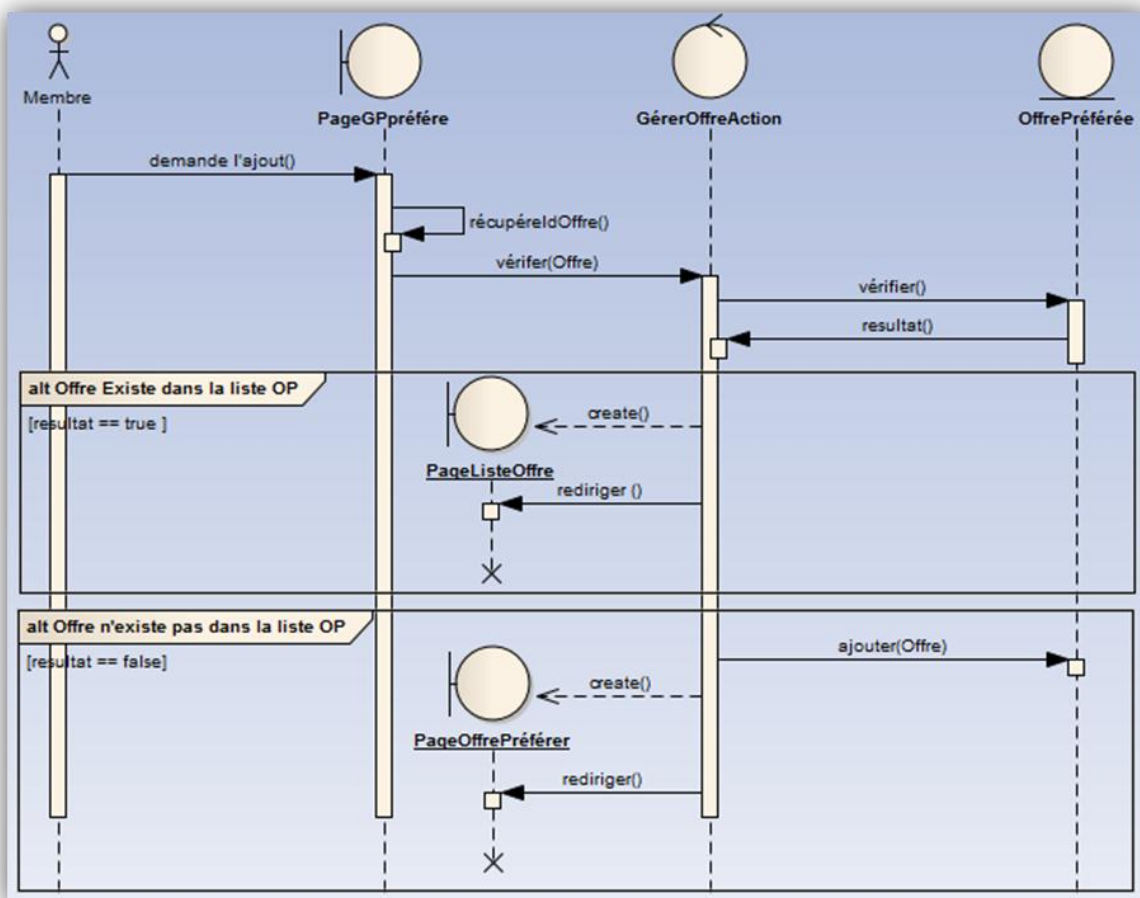
- 1) L'utilisateur demande l'ajout d'une offre.
- 2) Le système récupère le numéro de l'offre.
- 3) Le système vérifie si l'offre n'existe pas dans la liste.
- 4) Le système enregistre l'offre dans la liste des offres préférées du membre concerné.
- 5) Le système redirige le membre vers sa liste des offres préférées.

Enchaînements alternatifs :

A1 : L'offre existe déjà dans la liste

L'enchaînement A1 démarre au point (3) du scénario nominal

4) Le système ignore la demande et redirige le membre vers sa liste des offres.



- **Supprimer Offre Préféré :**

Sommaire d'identification :

Titre : Supprimer Offre Préféré

But : Supprimer une offre de la liste des offres préférées.

Résumé : Donner la possibilité aux membres de supprimer une offre de la liste des offres préférées.

Acteur : Membre

Description des enchaînements :

Pré-conditions : Le membre est authentifié.

Post-conditions : L'offre est supprimée.

Enchaînement nominal : Ce cas d'utilisation commence lorsque le membre demande au système de supprimer une offre de la liste des offres préférées.

- 1) L'utilisateur demande la suppression de l'offre.
- 2) Le système récupère le numéro de l'offre.
- 3) Le système supprime l'offre de la liste des offres préférées du membre concerné.
- 4) Le système redirige le membre vers sa liste des offres préférées.

- **Valider Marchand :**

Sommaire d'identification :

Titre : Valider Marchand

But : Valider l'inscription d'un marchand.

Résumé : L'administrateur doit approuver l'inscription des marchands avant qu'ils puissent commencer à utiliser le site.

Acteur : Administrateur

Description des enchaînements :

Pré-conditions : L'administrateur est authentifié.

Post-conditions : L'inscription du marchand est approuvée.

Enchaînement nominal : Ce cas d'utilisation commence lorsque l'administrateur demande la validation de l'inscription d'un marchand

- 1) L'administrateur demande la liste des nouvelles inscriptions.
- 2) L'administrateur choisit l'inscription à valider.
- 3) Le système affiche l'inscription pour vérifier les informations envoyées par le marchand.
- 4) L'administrateur approuve l'inscription.
- 5) Le système enregistre le marchand approuvé.

Enchaînements alternatifs :

A1 : Inscription non valide

L'enchaînement A1 démarre au point (3) du scénario nominal

- 4) L'administrateur désapprouve l'inscription.

Le scénario nominal reprend au point (2).

- **Modifier Marchand :**

Sommaire d'identification :

Titre : Modifier Marchand

But : Modifier les différentes informations d'un marchand

Résumé : Permettre à l'administrateur de modifier les différentes informations du marchand.

Acteur : Administrateur

Description des enchaînements :

Pré-conditions : L'administrateur est authentifié.

Post-conditions : Le marchand est modifié.

Enchaînement nominal : Ce cas d'utilisation commence lorsque l'administrateur demande la modification des informations d'un marchand

- 1) L'administrateur sélectionne un marchand.
- 2) Le système les différentes informations relatives à ce marchand.
- 3) L'administrateur modifie les informations voulues et valide.
- 4) Le système vérifie que les champs sont corrects.
- 5) Le système enregistre les modifications.

Enchaînements alternatifs :

A1 : Champs invalides

L'enchaînement A1 démarre au point (4) du scénario nominal

- 5) Le système indique à l'utilisateur qu'un champ n'est pas valide

Le scénario nominal reprend au point (3).

- **Supprimer Marchand :**

Sommaire d'identification :

Titre : Supprimer Marchand

But : Supprimer un marchand du site

Résumé : Permettre à l'administrateur de supprimer un marchand qui ne respecte pas les conditions générales d'utilisation du site.

Acteur : Administrateur

Description des enchaînements :

Pré-conditions :- L'administrateur est authentifié.

- Le marchand existe.

Post-conditions : Le marchand est supprimé.

Enchaînement nominal : Ce cas d'utilisation commence lorsque l'administrateur demande la suppression d'un marchand.

- 1) L'administrateur demande la liste des marchands.
- 2) Le système affiche la liste des marchands.

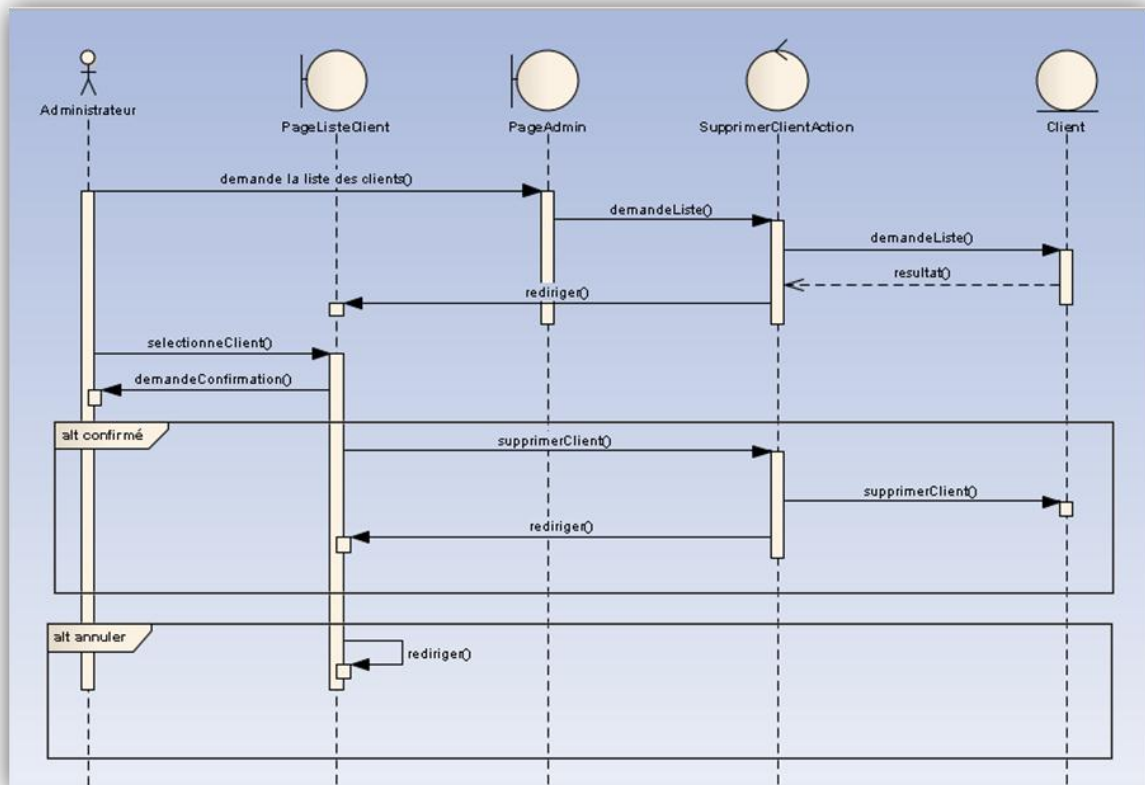
- 3) L'administrateur sélectionne le marchand à supprimer.
- 4) Le système demande à l'administrateur la confirmation de la suppression.
- 5) Le système supprime le marchand.
- 6) Le système revient vers la liste des marchands.

Enchaînements d'exception :

E1 : Le marchand annule la suppression

L'enchaînement E1 démarre au point (4) du scénario nominal

- 5) Le système revient vers la liste des marchands.



• **Ajouter Catégorie :**

Sommaire d'identification :

Titre : Ajouter Catégorie

But : Ajouter une catégorie

Résumé : Ajouter une catégorie qui regroupe les produits

Acteur : Administrateur

Description des enchaînements :

Pré-conditions : L'administrateur est authentifié.

Post-conditions : La catégorie est ajoutée.

Enchaînement nominal : Ce cas d'utilisation commence lorsque l'administrateur demande l'ajout d'une catégorie.

- 1) L'administrateur demande l'ajout d'une catégorie.
- 2) Le système affiche le formulaire d'ajout.
- 3) L'administrateur remplit les champs et valide.
- 4) Le système vérifie si les champs sont corrects.
- 5) Le système vérifie si la catégorie n'existe pas.
- 6) Le système enregistre la nouvelle catégorie.

Enchaînements alternatifs :

A1 : La catégorie existe

L'enchaînement A1 démarre au point (5) du scénario nominal

6) Le système indique à l'utilisateur que la catégorie existe déjà.
Le scénario nominal reprend au point (3).

- **Modifier Catégorie** :

Sommaire d'identification :

Titre : Modifier Catégorie

But : Modifier les informations d'une catégorie.

Résumé : Donner la possibilité à l'administrateur de modifier les informations d'une catégorie.

Acteur : Administrateur

Description des enchaînements :

Pré-conditions : - L'administrateur est authentifié.
- La catégorie existe.

Post-conditions : La catégorie est modifiée.

Enchaînement nominal : Ce cas d'utilisation commence lorsque le marchand demande au système la modification des informations d'une catégorie.

- 1) L'administrateur demande la liste des catégories.
- 2) Le système affiche la liste des catégories.
- 3) L'administrateur sélectionne la catégorie à modifier.
- 4) Le système affiche le formulaire de modification qui contient les informations de la catégorie.
- 5) L'administrateur modifie les informations désirées et valide.
- 6) Le système vérifie les champs s'ils sont corrects.
- 7) Le système enregistre les modifications.
- 8) Le système redirige L'administrateur vers la liste des catégories.