

**Université Mouloud Mammeri Tizi Ouzou**  
Faculté de génie électrique et informatique  
Département Informatique



---

---

**REALISATION D'UN SITE WEB DYNAMIQUE  
COMMERCIALE**

---

---

**Par**  
**SLIMANI Yanis**

**PROJET DE FIN DE CYCLE**

**Présenté à**  
**Université Mouloud Mammeri**  
**Tizi Ouzou**

**En vue de l'obtention du diplôme**  
**De licence en Systèmes Informatiques (L.M.D)**

**Année Universitaire 2019 - 2020**

# *Remerciements*

*Nous voulons exprimer par ces quelques lignes de remerciements nos gratitude envers tous ceux en qui, par leur présence, leur soutien, leur disponibilité et leurs conseils*

*Nous avons trouvé courage afin d'accomplir ce projet.*

*En fin, on ne peut achever ce projet sans exprimer nos gratitude à nos professeurs jury Mr **M. SOUALAH**, Mme **A. Berkane** pour leur dévouement et leur assistance tout au long de cette année, qui nous ont beaucoup aidé dans la conception de notre rapport et nous a permis de nous développer dans plusieurs aspects, et qui nous ont guidé lors de la réalisation de notre projet et nous ont permis d'acquérir de nouvelles connaissances dans des domaines multiples et variés.*

*Nous voulons aussi envoyer tous nos sincères et distinctes salutations et remerciements aux parents qui nous ont soutenus moralement, aux membres de la famille qui nous ont motivé dans diverses démarches et difficultés auxquelles nous avons fait face .*

## **Introduction générale**

En vue de l'indisponibilité de certains articles dans les magasins physique, la perte de temps généré par le processus d'achat qui consiste à faire le déplacement vers ces boutiques, et les files d'attentes, et on dit souvent que les chiffres parlent d'eux-mêmes. Si l'on se fiait uniquement à ces critères, on pourrait dire que les magasins virtuels ont déjà gagné par K.O face aux magasins locales.

Aujourd'hui, le commerce électronique a brisé les barrières du physique, il est considéré comme une façon plus sûre et plus facile par de nombreuses personnes afin d'effectuer leurs achats ou et quand ils le souhaitent sans se déplacer, surtout depuis que de nouvelles options de vente attrayante ont été développées et aussi, cela permet de commercialiser de nouvelles gammes de produits et de diversifier les sources de revenus sans avoir à réaliser un gros investissement.

En effet, ces secteurs séduisent autant en ligne qu'en commerce physique.

# SOMMAIRE

## INTRODUCTION GENERALE

Erreur ! Signet non défini.

## CHAPITRE 1 - ETUDE DU PROBLÈME

Erreur ! Signet non défini.

### 1 – INTRODUCTION

Erreur ! Signet non défini.

### 2 – PROBLÉMATIQUE

Erreur ! Signet non défini.

### 3 – OBJECTIF DE NOTRE TRAVAIL

Erreur ! Signet non défini.

### 4 – CONCLUSION

5

## CHAPITRE 2 - ANALYSE ET CONCEPTION

6

### 1 – INTRODUCTION

6

### 2 – PRÉSENTATION D'UML

6

#### 2.1 – Définition

6

#### 2.2 - Extension d'UML pour le web

7

#### 2.3 - Modélisation avec UML

7

### 3 - ANALYSE ET CONCEPTION

8

#### 3.1 – Analyse

8

##### 3.1.1 - Spécification des besoins

8

#### 3.2 - Conception

15

##### 3.2.1 - Elaboration des diagrammes de séquences

15

##### 3.2.2 - Diagrammes d'activité

19

##### 3.2.3 - Diagramme de classe

24

##### 3.2.4 - Conception de la base de données

26

### 4 – CONCLUSION

28

## CHAPITRE 3 - RÉALISATION

29

### 1 – INTRODUCTION

30

### 2 - ENVIRONNEMENT DE DEVELOPEMENT

30

#### 2.1 – environnement logiciel

30

#### 2.2 - Les outils utilisés

30

##### 2.2.1 - Les langages/bibliothèques utilisés

30

##### 2.2.2 - Les logiciels utilisés

32

### 3 – LES PRINCIPALES INTERFACES GRAPHIQUES :

33

#### 3.1– Interfaces clients :

33

#### 3.2 – Interfaces administrateurs

36

### 4 - CONCLUSION

37

## CONCLUSION GENERALE

38

## BIBLIOGRAPHIE

39

## **Chapitre 1: PRÉSENTATION DU PROJET**

### **1- Introduction :**

Notre projet consiste à créer un magasin catalogue de vente de vêtements et accessoires en ligne. Pour la conception de ce projet nous avons utilisé une **démarche (UP)** s'appuyant sur la modélisation UML pour la description de l'architecture du logiciel (fonctionnelle, logicielle et physique) et la mise au point de cas d'utilisation permettant de décrire les besoins et exigences des utilisateurs.

### **2- Problématique :**

Pour acheter un vêtement telle qu'un t-shirt ou une chemise, le client doit se déplacer directement au local de la boutique afin de chercher une offre de vente qui satisfait ses besoins. Ses déplacements peuvent être inutiles et même peuvent provoquer un gaspillage de temps. D'ailleurs, même le vendeur n'a aucun moyen pour mettre à disposition ses annonces de vente et services, à l'exception des supports traditionnels tels que les journaux ou les petites affiches. Ainsi, un moyen fiable et automatisé permettant d'informer un grand nombre de clients des offres de vente et des services nécessaires afin de leur faciliter leur achat, et satisfaire leur besoin avec de bon plans et un gain de temps précieux.

Vu l'accroissement de la technologie Internet, les personnes se tournent vers la solution d'achat en ligne qui est devenu une nécessité incontournable pour les commerçants et pour les clients (consommateurs).

### **3- Objectif à atteindre :**

Dans le cadre d'une mémoire afin d'obtenir le diplôme de la Licence, l'objectif consiste à créer un site web pour vente de vêtements et accessoires en ligne qui vise à développer le processus de vente avec internet afin de faciliter et d'encourager la communication avec le client.

Notre site doit réaliser les objectifs suivants :

Permettre la consultation et la recherche des vêtements et accessoires disponibles.

Permettre au visiteur de s'inscrire et de se connecter.

Permettre à l'utilisateur d'acheter un article.

Permettre au visiteur de s'informer sur les prix des produits et les promotions.

Leur faciliter l'achat avec les options d'achats et de paiements proposées par notre site.

Permettre au visiteur d'obtenir des informations spécifiques sur les produits et services.

Commander ou réserver des produits en ligne pour ensuite les acheter.

## **5- Conclusions :**

A travers la problématique qu'on a cité ci-dessus, on a pu comprendre et identifier les besoins des clients, Ce chapitre donne une vision globale et essentiel car il constitue le point de départ pour la conception de la solution proposée, afin d'éviter les problèmes existants et atteindre nos objectifs. cette dernière sera détaillée dans le prochain chapitre.

## Chapitre2 : ANALYSE ET CONCEPTION

### 1 – INTRODUCTION

Dans le but d'une meilleure organisation et une bonne maîtrise du travail, tout processus de développement d'applications ou système informatique doit suivre une méthode ou démarche bien définie.

Dans ce chapitre, nous allons entamer ce processus par une analyse qui mettra en évidence les différents acteurs intervenants dans le système ciblé ainsi que leurs besoins.

La phase de conception s'appuyant sur les résultats de la phase d'analyse, celle-ci donnera la modélisation des objectifs à atteindre. Pour ce faire, notre démarche va s'appuyer sur le langage UML, Conçu pour la visualisation, la spécification et la construction des systèmes logiciels.

### 2 – PRÉSENTATION D'UML

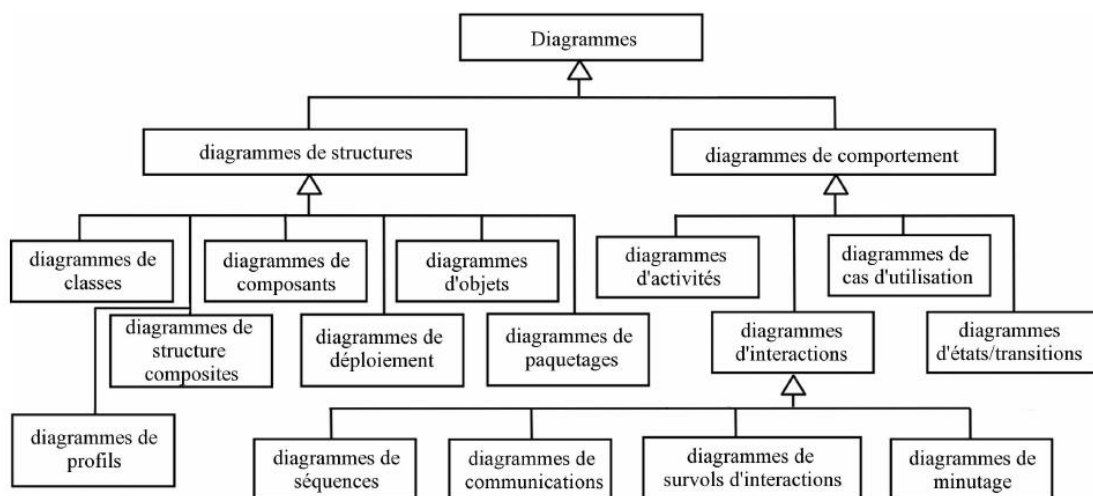
#### 2.1 – Définition :

Le langage UML (Unified Modeling Language, ou langage de modélisation unifié) a été pensé pour être un langage de modélisation visuelle commun, et riche sémantiquement et syntaxiquement. Il est destiné à l'architecture, la conception et la mise en œuvre de systèmes logiciels complexes par leur structure aussi bien que leur comportement. L'UML a des applications qui vont au-delà du développement logiciel, notamment pour les flux de processus dans l'industrie.

Il ressemble aux plans utilisés dans d'autres domaines et se compose de différents types de diagrammes. Dans l'ensemble, les diagrammes UML décrivent la limite, la structure et le comportement du système et des objets qui s'y trouvent.

L'UML n'est pas un langage de programmation, mais il existe des outils qui peuvent être utilisés pour générer du code en plusieurs langages à partir de diagrammes UML.

Le langage UML a une relation directe avec l'analyse et la conception orientées objet et s'articule autour de treize diagrammes officiels, représentés dans la figure suivante :



**Figure 1 - les diagrammes d'UML**

## 2.2 - Extension d'UML pour le web :

Une extension d'UML définit un ensemble de stéréotype et de contraintes, qui rend possible la modélisation d'application web.

- **Un stéréotype** est une extension du vocabulaire d'UML, il permet d'associer une nouvelle signification à un modèle. Les stéréotypes peuvent être appliquées à presque tous les éléments du modèle et sont habituellement représentés par une chaîne de caractère entre guillemets (<<>>).
- **Une étiquette** est une extension des propriétés d'un élément. Elle permet la description d'une nouvelle propriété du modèle elle est présentée dans un diagramme par une chaîne de caractère entre chevrons ().
- **Une contrainte** est une extension de la sémantique d'UML. Elle prescrit la règle que le modèle doit vérifier pour être qualifié. Elle est représentée par une chaîne de caractère.

Ces stéréotypes et ces contraintes sont appliqués sur des applications web, permettant ainsi de les représenter au sein du même modèle et sur les mêmes diagrammes que ceux qui décrivent le reste du système.

Le principal élément spécifique des applications web étant la page web, plusieurs stéréotypes qui lui sont destinés, décrits plus loin, conçu pour les éléments tel que les cadres, les cibles et les formulaires représentant eux aussi des composants architecturaux significatifs.

## 2.3 - Modélisation avec UML :

L'UML permet de représenter des modèles, mais il ne définit pas de processus d'élaboration. Les auteurs d'UML, conseillent tout de même une démarche pour favoriser la réussite d'un projet, cette démarche doit être :

- **Itératif et incrémental** : le projet est découpé en itération de courte durée qui aident à mieux suivre l'avancement global. A la fin de chaque itération, une partie exécutable du système final est produite, de façon incrémentale.
- **Centrée sur l'architecture** : c'est la clé de voûte d'un développement réussi. Les choix stratégiques définiront la qualité du logiciel.
- **Conduit par les cas d'utilisation** : tout est basé sur les besoins des utilisateurs du système, le but du développement lui-même est de répondre à leurs besoins. Chaque étape sera affichée et validée en fonction des besoins des utilisateurs.



### 3 - ANALYSE ET CONCEPTION :

#### 3.1 – Analyse :

##### 3.1.1 - Spécification des besoins :

###### 3.1.1.1 - Définition d'un acteur :

Un acteur représente un élément externe au système qui interagit avec ce dernier dans le but de le faire fonctionner et d'en tirer profit. Généralement, il s'agit d'un utilisateur du système ou bien d'une entité qui coopère avec le système.

###### 3.1.1.2 - Identification des acteurs de notre système :

###### Administrateur :

Pour les sites web on l'appelle généralement « le webmaster ». C'est celui qui assure le dynamisme du site, en plus de l'utilisateur il peut:

- Modifier un produit
- Ajouter une catégorie
- Modifier une modifier une catégorie
- Supprimer une supprimer une catégorie
- Ajouter une catégorie de produit
- Supprimer une catégorie de produit
- Modifier une catégorie de produit
- Supprimer un utilisateur
- Modifier le CSS
- Ajouter une marque
- Supprimer une marque
- Modifier une marque
- Ajouter des diapositives
- Supprimer une diapositive
- Afficher les clients

###### Le client en ligne :

Est un visiteur authentifié, c'est-à-dire un visiteur connecté grâce à une adresse email et un mot de passe, il peut donc :

- Consulter les produits existants
- Faire une recherche
- Acheter des produits
- Consulter son profil
- Modifier son profil

###### Le visiteur du site :

Représente l'internaute qui viendrait se connecter au site pour :

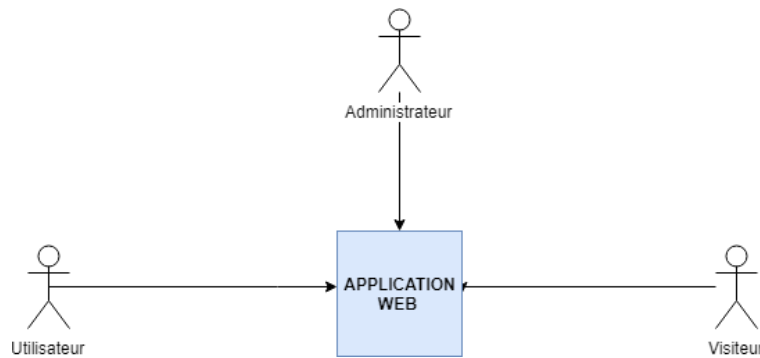
- Visualisez les produits
- Rechercher des produits
- S'inscrire au site
- Se connecter

Un visiteur peut aussi devenir un client en ligne en s'inscrivant sur le site.

### 3.1.1.3 - Diagramme de contexte :

Le diagramme de contexte est un modèle conceptuel de flux qui permet d'avoir une vision globale des interactions entre les systèmes et l'environnement extérieur. Il permet aussi de bien délimiter le champ d'étude.

L'identification des acteurs de notre système nous permet de construire le diagramme de contexte suivant :



**Figure 2 - Diagramme de contexte**

### 3.1.1.4 - Identification des cas d'utilisation et les scénarios

Pour chaque acteur identifié précédemment il convient de rechercher les différentes intentions pour lesquelles il utilise le système.

#### A - Quelques définition de base :

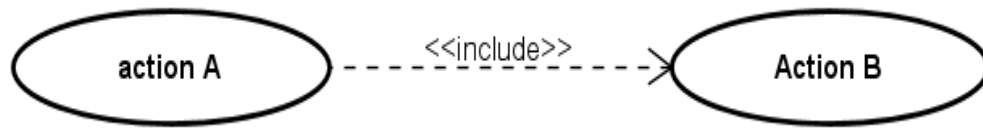
- **Une tâche** Est l'ensemble logique d'opérations permettant l'exécution d'un programme ou d'une partie d'un programme.
- **Un scénario** est une succession particulière d'enchaînement, qui est l'unité de description de la séquence d'action, s'exécutant du début à la fin du cas d'utilisation. Un ensemble de scénarios pour un cas d'utilisation identifie tout ce qu'il peut arriver lorsque ce cas d'utilisation est mis en œuvre.
- **Les cas d'utilisation** expriment le comportement du système (ses fonctionnalités) selon le point de vue des utilisateurs. Ils constituent une technique qui permet de déterminer les besoins des utilisateurs et de capturer les exigences fonctionnelles d'un système. Ils décrivent les interactions entre les utilisateurs d'un système et le système lui-même.

#### B - Relations entre cas d'utilisation :

- **La relation d'inclusion** : Le cas d'utilisation source incorpore explicitement et de manière obligatoire le comportement décrit dans le cas d'utilisation.

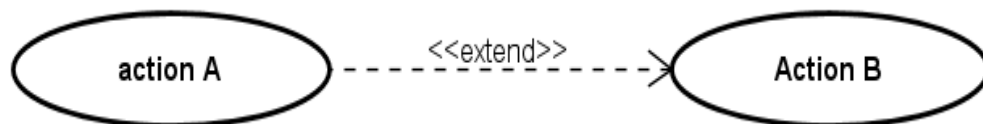
Un cas A est inclus dans un cas B si le comportement décrit par le cas A est inclus dans

le comportement du cas B, on dit alors que B dépend de A. Cette dépendance est symbolisée par le stéréotype **include**.



• **La relation d'extension** : le cas d'utilisation incorpore implicitement et de manière facultative un autre cas d'utilisation à l'endroit spécifié. Ceci veut dire que le cas d'utilisation source <<étend>> ou précise le cas d'utilisation destination.

Si le comportement de B peut être étendu par le comportement de A, on dit alors que A étend B. Une extension est souvent soumise à une condition, elle est symbolisée par le stéréotype **extend**



• **La relation de généralisation** : un cas A est une généralisation d'un cas B, si B est un cas particulier de A.

### C - Spécification des cas d'utilisation

Acteur	Cas d'utilisation
Visiteur	<b>Consulter</b> <ul style="list-style-type: none"> <li>• Produits</li> <li>• Catégories</li> <li>• Catégories produites</li> <li>• Marques</li> <li>• Diapositives</li> <li>• Boîte</li> <li>• Conditions d'utilisation</li> <li>• Panier</li> </ul> <b>Se connecter</b> <ul style="list-style-type: none"> <li>• Remplir formulaire</li> </ul> <b>S'inscrire</b> <ul style="list-style-type: none"> <li>• Remplir formulaire</li> </ul> <b>Contacter</b> <ul style="list-style-type: none"> <li>• Remplir un formulaire</li> </ul>
Utilisateur	<b>Consulter</b> <ul style="list-style-type: none"> <li>• Produits</li> <li>• Catégories</li> <li>• Catégories produites</li> </ul>

	<ul style="list-style-type: none"> <li>• Marques</li> <li>• Diapositives</li> <li>• Boite</li> <li>• Conditions d'utilisation</li> <li>• Procéder au paiement</li> <li>• Panier</li> <li>• Commande</li> </ul> <p><b>Rechercher</b></p> <ul style="list-style-type: none"> <li>• Par mot clé</li> <li>• Par catégorie</li> <li>• Par catégorie produit</li> </ul> <p><b>Acheter des Produits</b></p> <p><b>Visualiser son profil</b></p> <ul style="list-style-type: none"> <li>• Modifier ses informations</li> <li>• Supprimer le compte</li> </ul>
Administrateur	<p><b>Consulter</b></p> <ul style="list-style-type: none"> <li>• Produits</li> <li>• Catégories</li> <li>• Catégories produites</li> <li>• Marques</li> <li>• Diapositives</li> <li>• Boite</li> <li>• Conditions d'utilisation</li> <li>• Procéder au paiement</li> <li>• Commandes</li> <li>• CSS</li> <li>• Administrateurs</li> <li>• Clients</li> </ul> <p><b>Modifier</b></p> <ul style="list-style-type: none"> <li>• Produits</li> <li>• Catégories</li> <li>• Catégories Produits</li> <li>• Marques</li> <li>• Boites</li> <li>• Diapositives</li> <li>• Conditions d'utilisation</li> <li>• Administrateur</li> </ul> <p><b>Supprimer</b></p> <ul style="list-style-type: none"> <li>• Produits</li> <li>• Catégories</li> <li>• Catégories Produits</li> <li>• Marques</li> <li>• Boites</li> <li>• Diapositives</li> <li>• Conditions d'utilisation</li> <li>• Administrateur</li> <li>• Paiement</li> </ul> <p><b>Ajouter</b></p> <ul style="list-style-type: none"> <li>• Produits</li> </ul>

	<ul style="list-style-type: none"><li>• Catégories</li><li>• Catégories Produits</li><li>• Marques</li><li>• Boites</li><li>• Diapositives</li><li>• Conditions d'utilisation</li><li>• Administrateur</li><li>• Paiement</li></ul> <b>Visualiser son profil</b> <ul style="list-style-type: none"><li>• Modifier ses informations</li></ul>
--	--

**Figure 3 - Spécification des cas d'utilisation**

## D - Diagrammes des cas d'utilisation :

### 1. Diagramme de cas d'utilisation <<Administrateur>> :

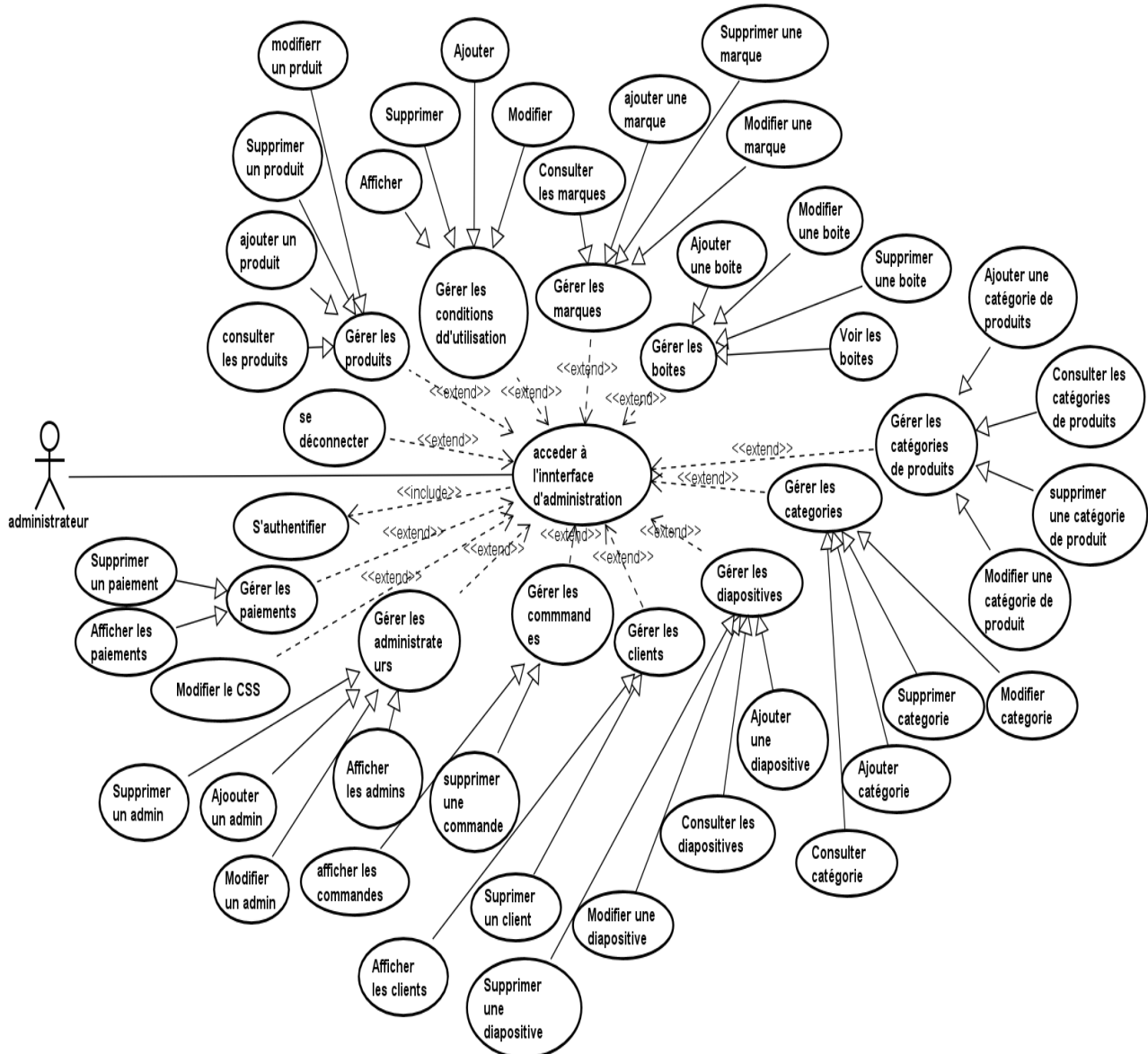
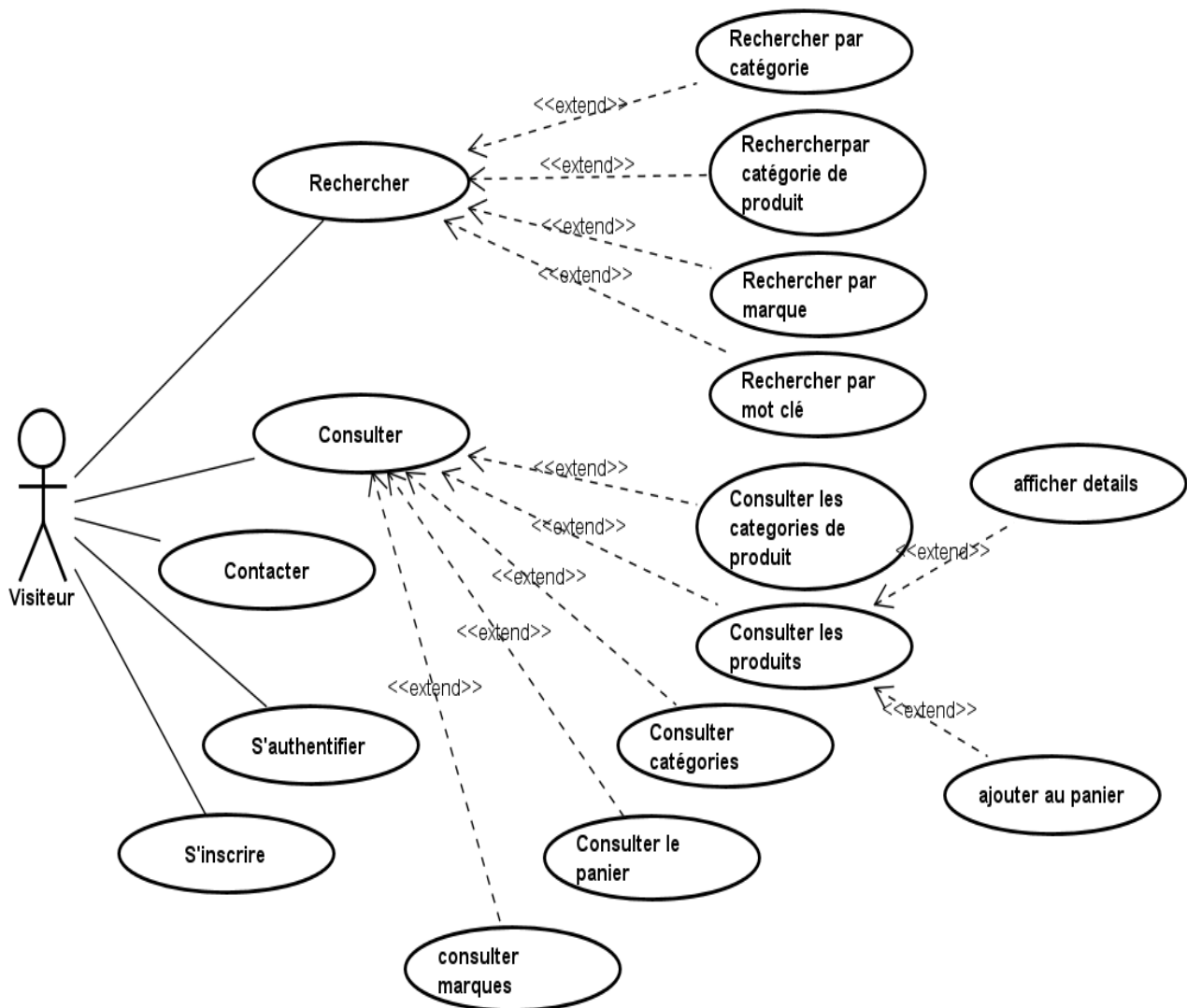


Figure 4 - Diagramme cas d'utilisation Administrateur

**2-Diagramme de cas d'utilisation<<Visiteur>> :****Figure 5 - Diagramme de cas d'utilisation visiteur**

### 3-Diagramme de cas d'utilisation<< Utilisateur >> :

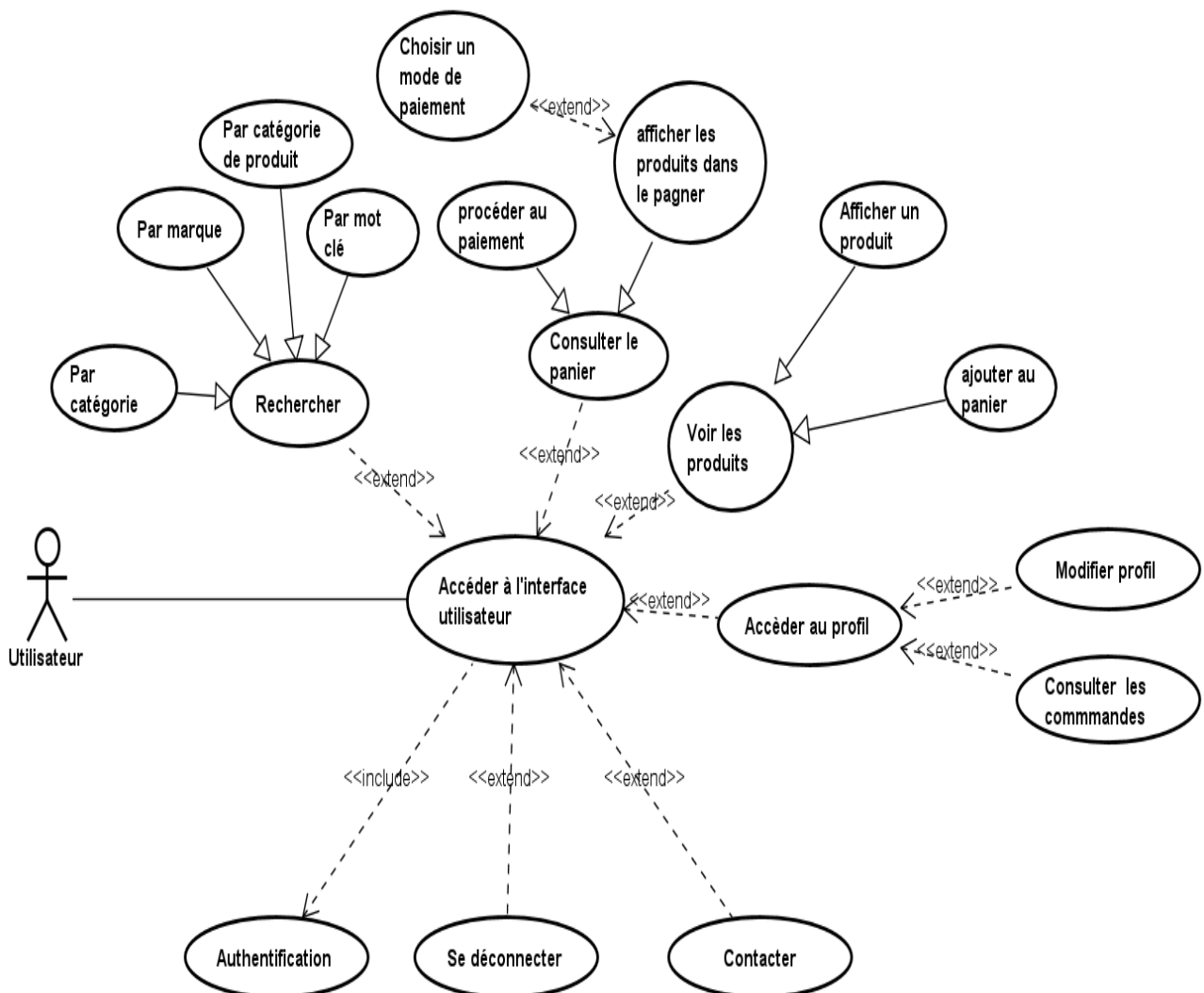


Figure 6 – Cas d'utilisateur pour l'utilisateur.

## 3.2 – Conception :

Après avoir déterminé tous les besoins et réalisé les diagrammes des cas d'utilisation nous allons élaborer les diagrammes de séquences avec les objets d'analyse.

### 3.2.1 - Élaboration des diagrammes de séquences :

#### 3.2.1.1 – Définition :

Le diagramme de séquence est la représentation graphique des interactions entre les acteurs et le système selon un ordre chronologique dans la formulation UML.

La dimension verticale du diagramme représente le temps, permettant de visualiser l'enchaînement des actions dans le temps, et de spécifier la naissance. Les périodes d'activité des objets sont symbolisées par des rectangles, et ces objets dialoguent à l'aide de messages.

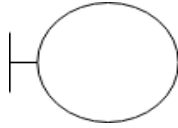


### 3.2.1.2 - Les objets du diagramme de séquence

#### A - Objet interface

Représente l'interface entre l'acteur et le système, tel que les interfaces graphiques et les formulaires. C'est une description des opérations visibles.

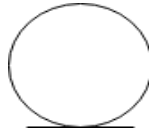
**L'icône :**



#### B - Objet entité

Représente les concepts métier. Ils sont des objets décrits dans un cas d'utilisation et se trouvent dans un autre cas d'utilisation tel qu'un opérateur, utilisateur, ... etc.

**L'icône :**



#### C - Objet contrôle

C'est un objet actif, il possède un flot de contrôle. Un objet actif peut activer un objet passif pour le temps d'une opération en lui envoyant un message. Il dirige les activités des entités et interfaces.

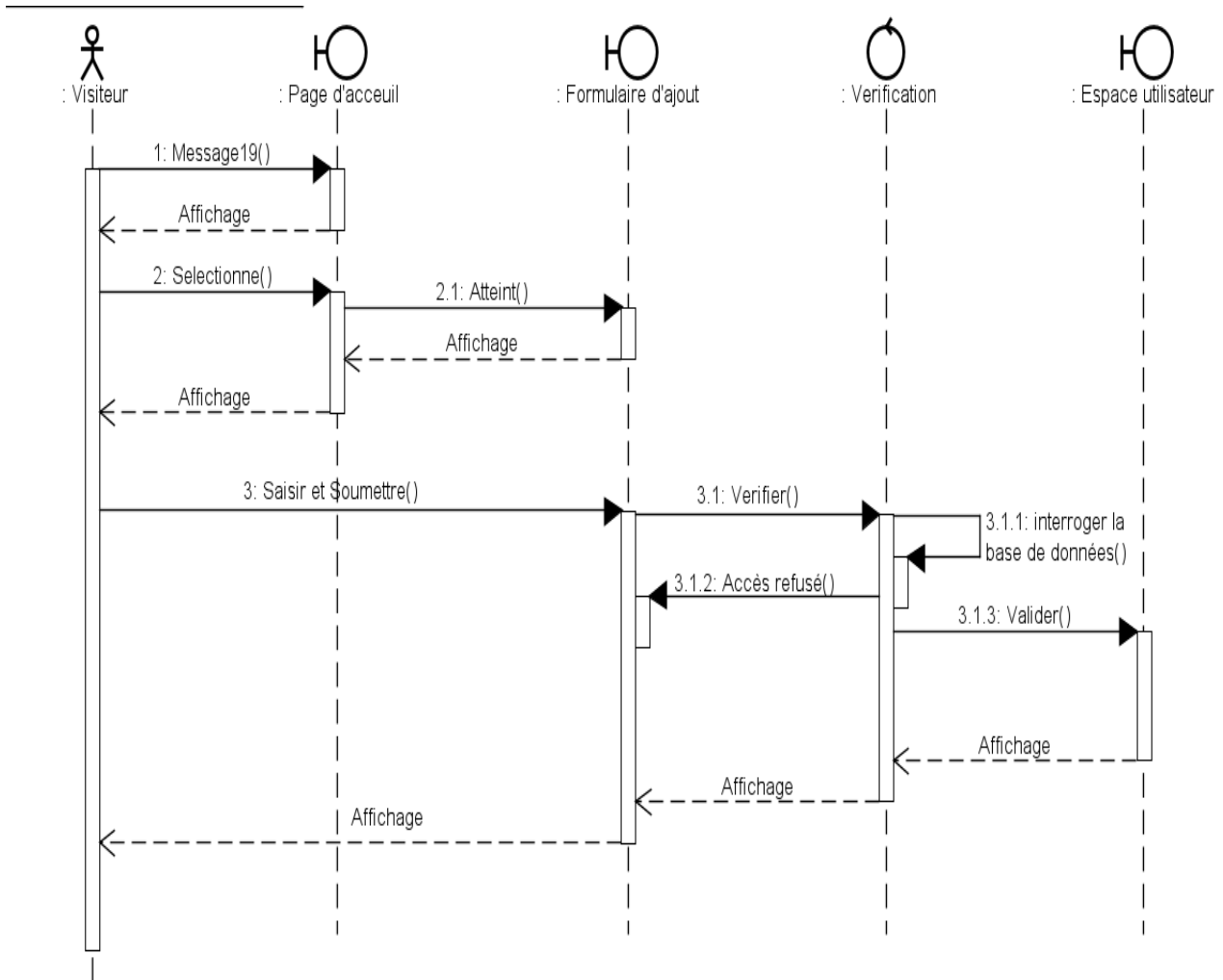
Ces objets sont obtenus en extrayant les verbes des cas d'utilisation.

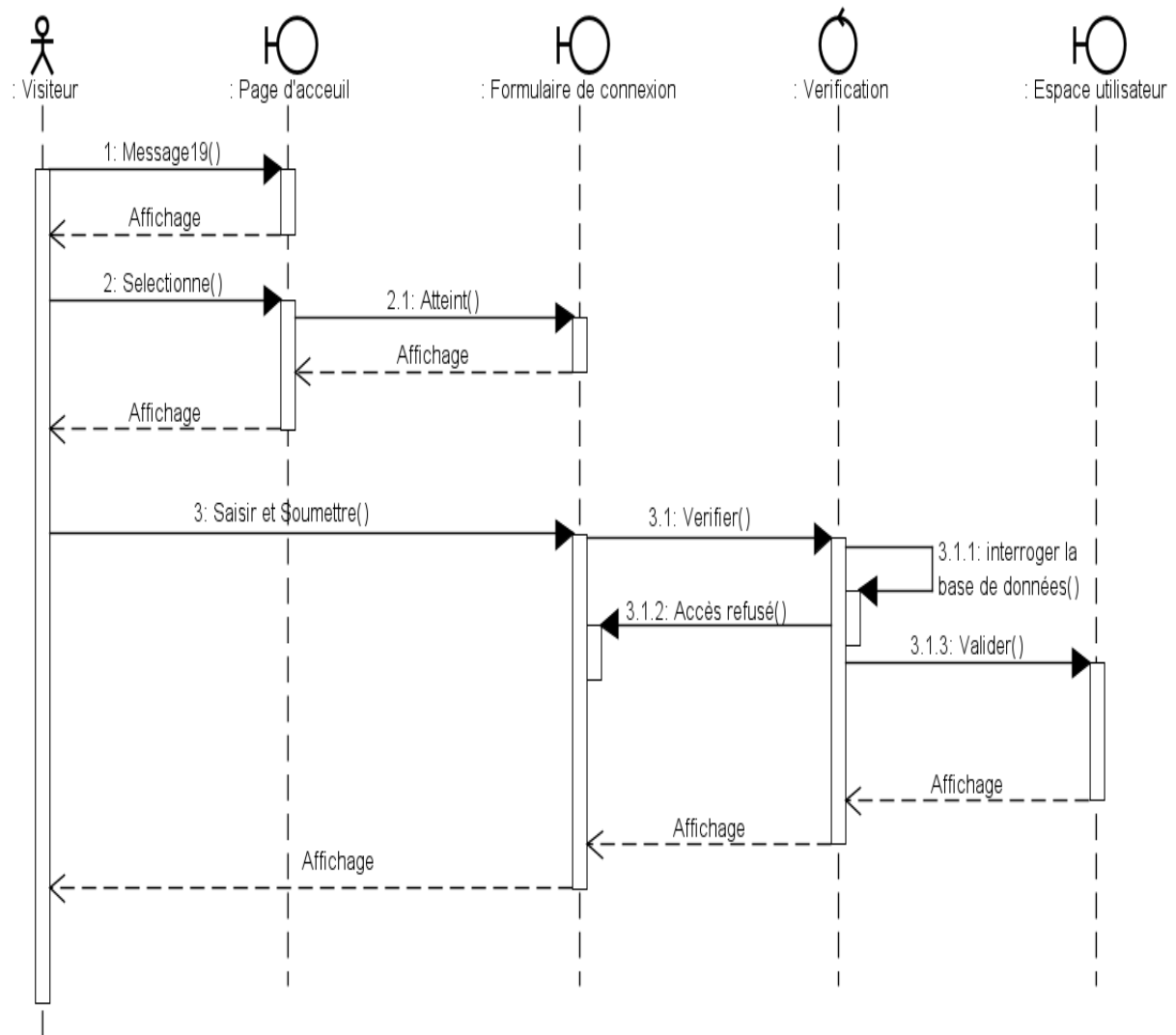
**L'icône :**

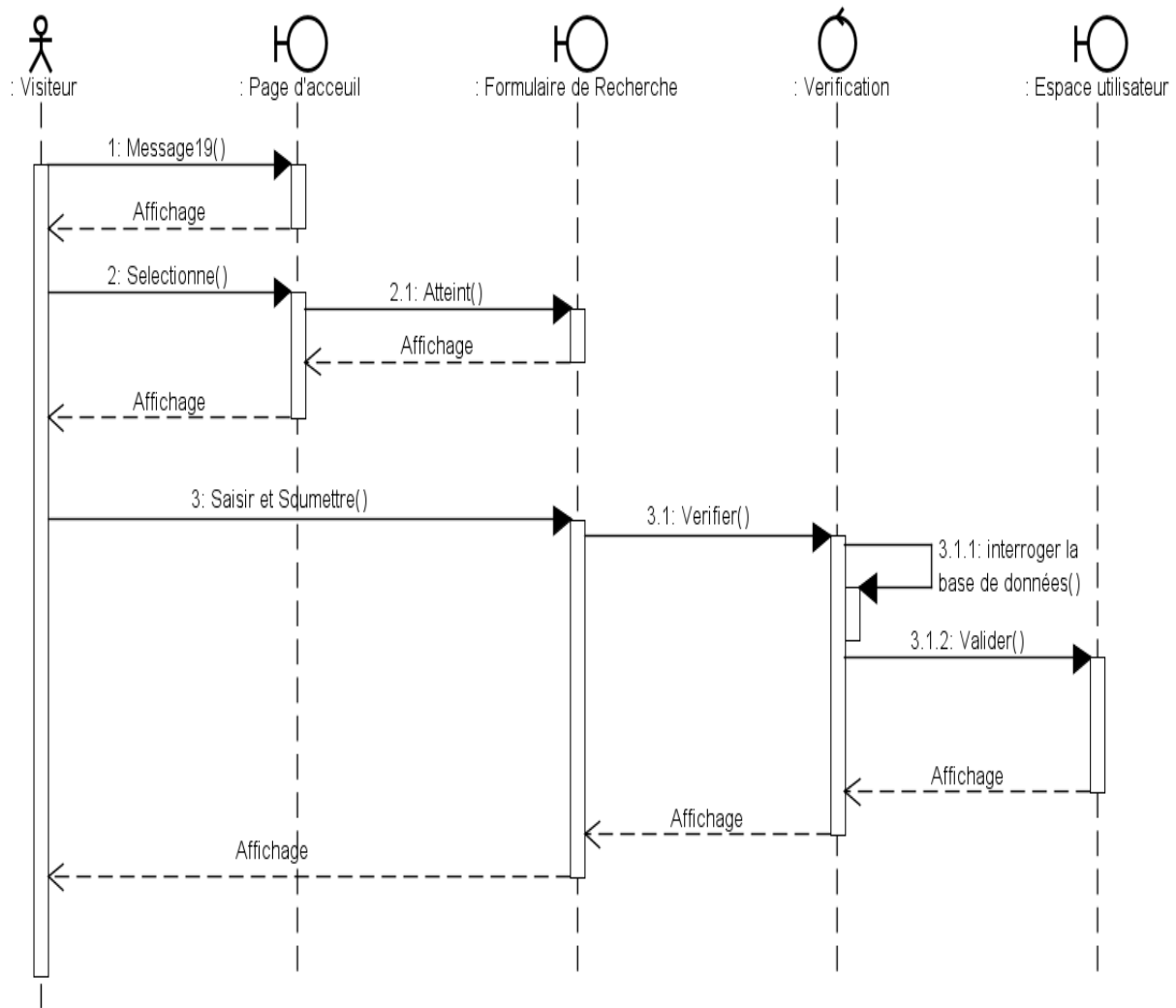


### 3.2.1.3 - Les diagrammes de séquences

Comme nous avons un nombre important de cas d'utilisation, nous allons présenter l'analyse de quelques cas seulement :

**Figure 7 - Diagramme de séquence <<Ajout produit>>**

**Figure 8 - Diagramme de séquence <<Authentification>>**



**Figure 9 - Diagramme de séquence <<Recherche>>**

### 3.2.2 - Diagrammes d'activité :

#### 3.2.2.1 – Définition :

Un diagramme d'activité permet de modéliser un processus interactif global ou partiel pour un système donné (logiciel, système d'information).

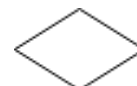
Il est recommandable pour exprimer une dimension temporelle sur une partie du modèle, le diagramme d'activité est également utilisé pour décrire un flux de travail.

#### 3.2.2.2 - Les composants graphiques :

Action du système :



Condition :



Début du système :

Fin du système :

### 3.2.2.3 - Les diagrammes d'activités :

#### A - Diagramme d'activité <<S'authentifier>> :

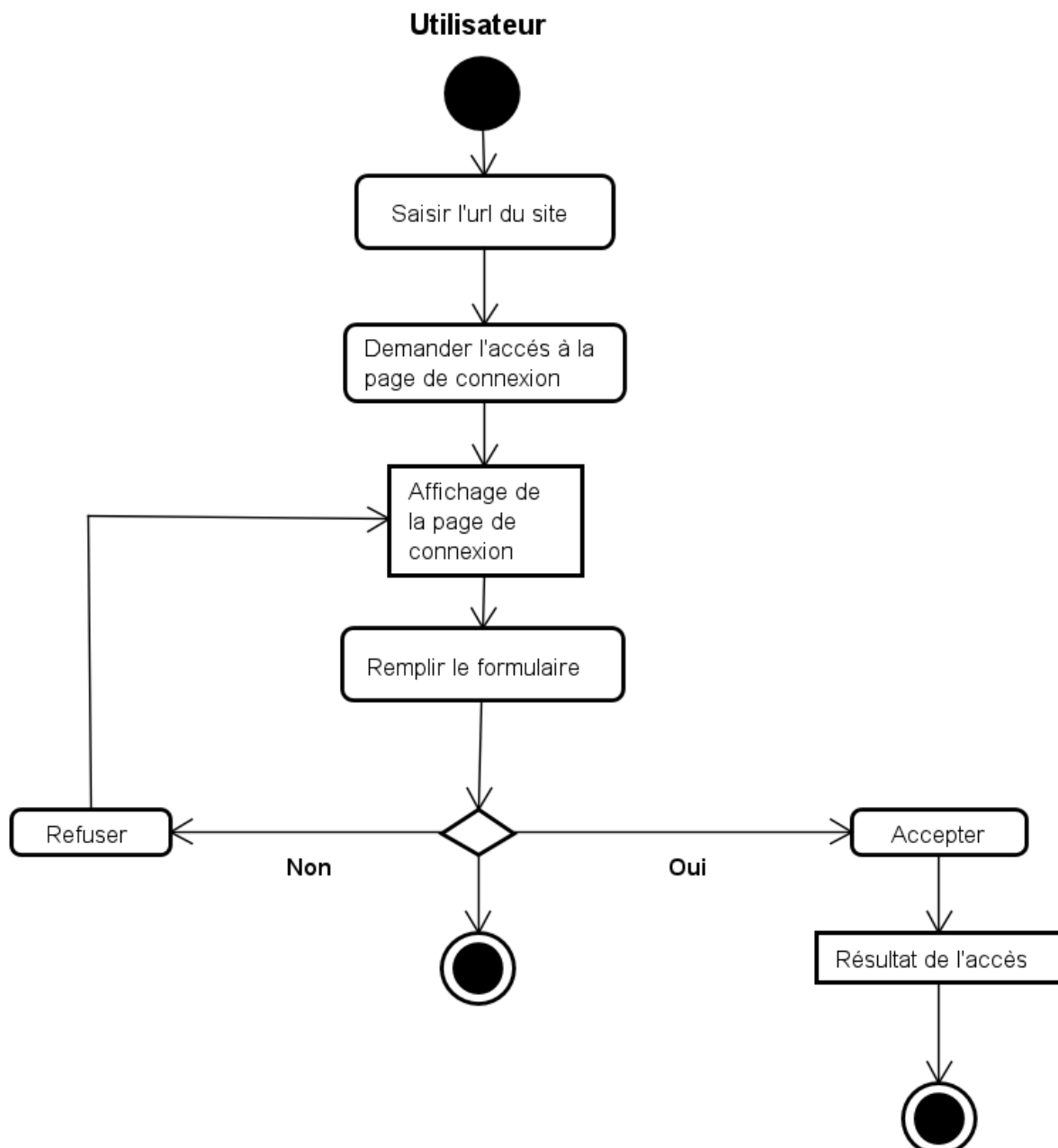
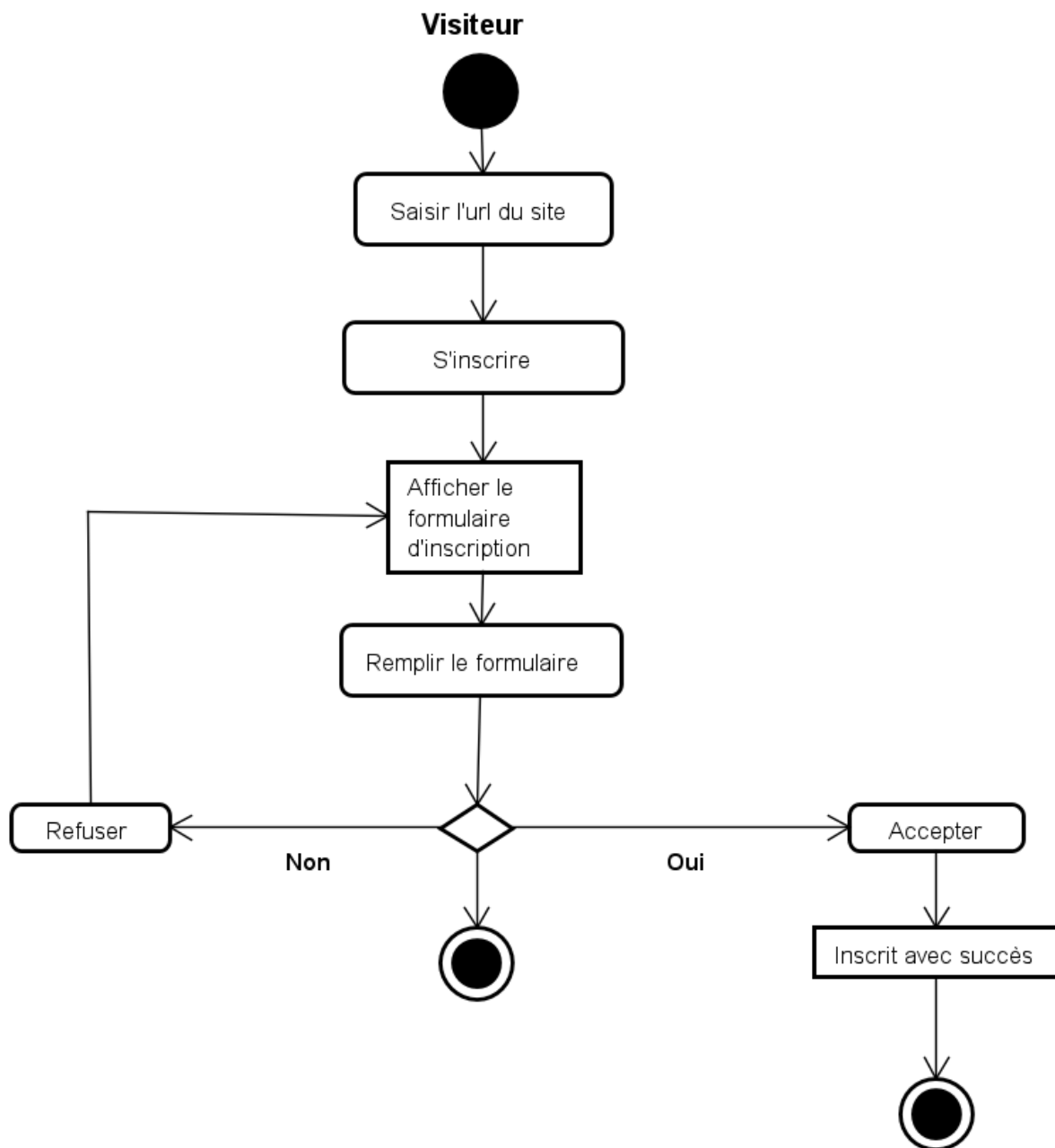
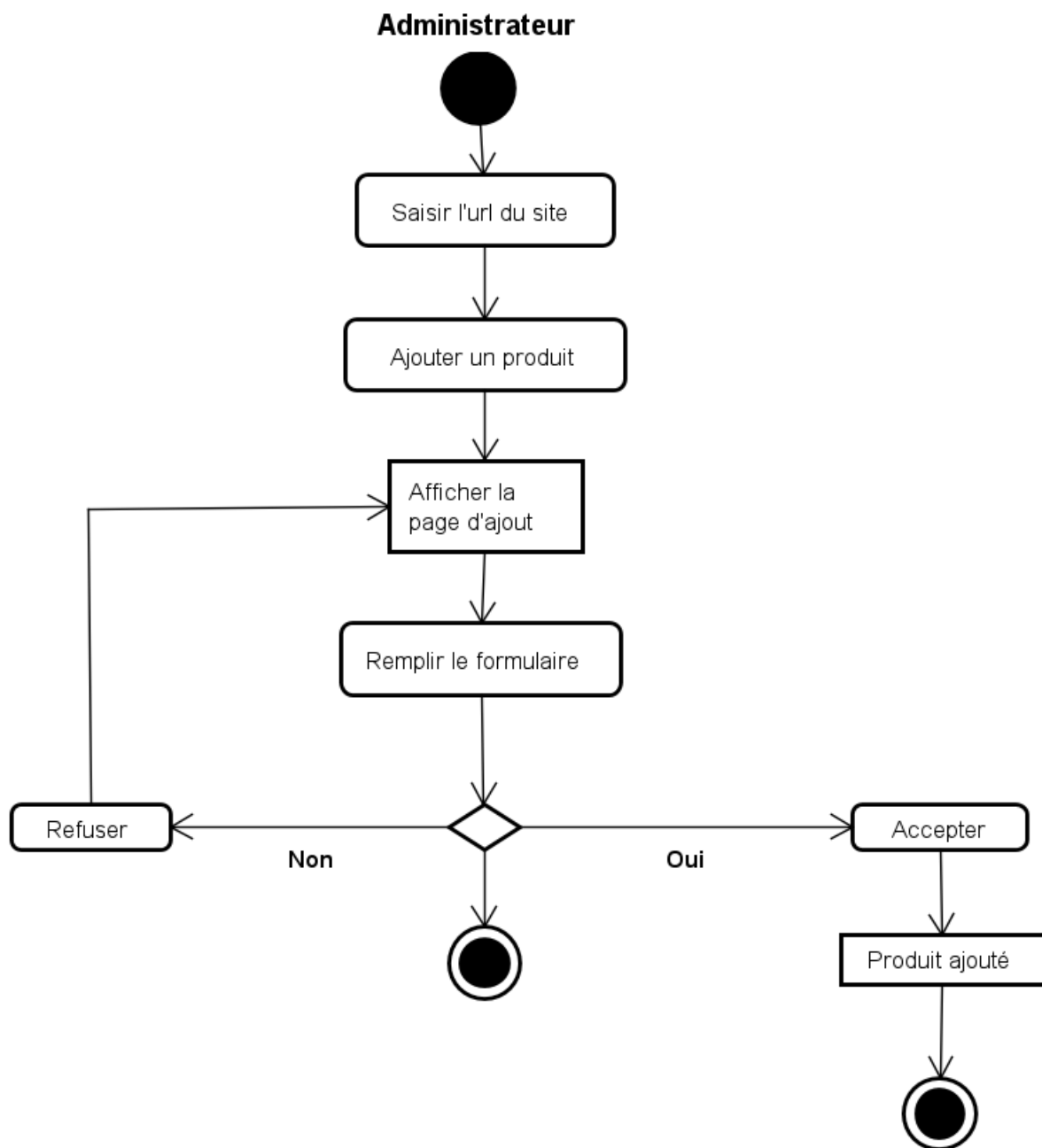


Figure 10 - Diagramme d'activité <<S'authentifier>>

**B - Diagramme d'activité <<S'inscrire>> :****Figure 11 - Diagramme d'activité <<S'inscrire>>**

**C - Diagramme d'activité <<Ajouter un produit>>****Figure 13 - Diagramme d'activité <<Ajouter un produit>>**

## D - Diagramme des composants :

Les diagrammes des composants font partie de la construction du projet informatique. Élaborés avec le logiciel de création de diagrammes UML, ils aident les développeurs à comprendre la structure des systèmes existants, puis à en construire de nouveaux.

L'objectif d'un diagramme de composants est de montrer la relation entre les différents composants d'un système.

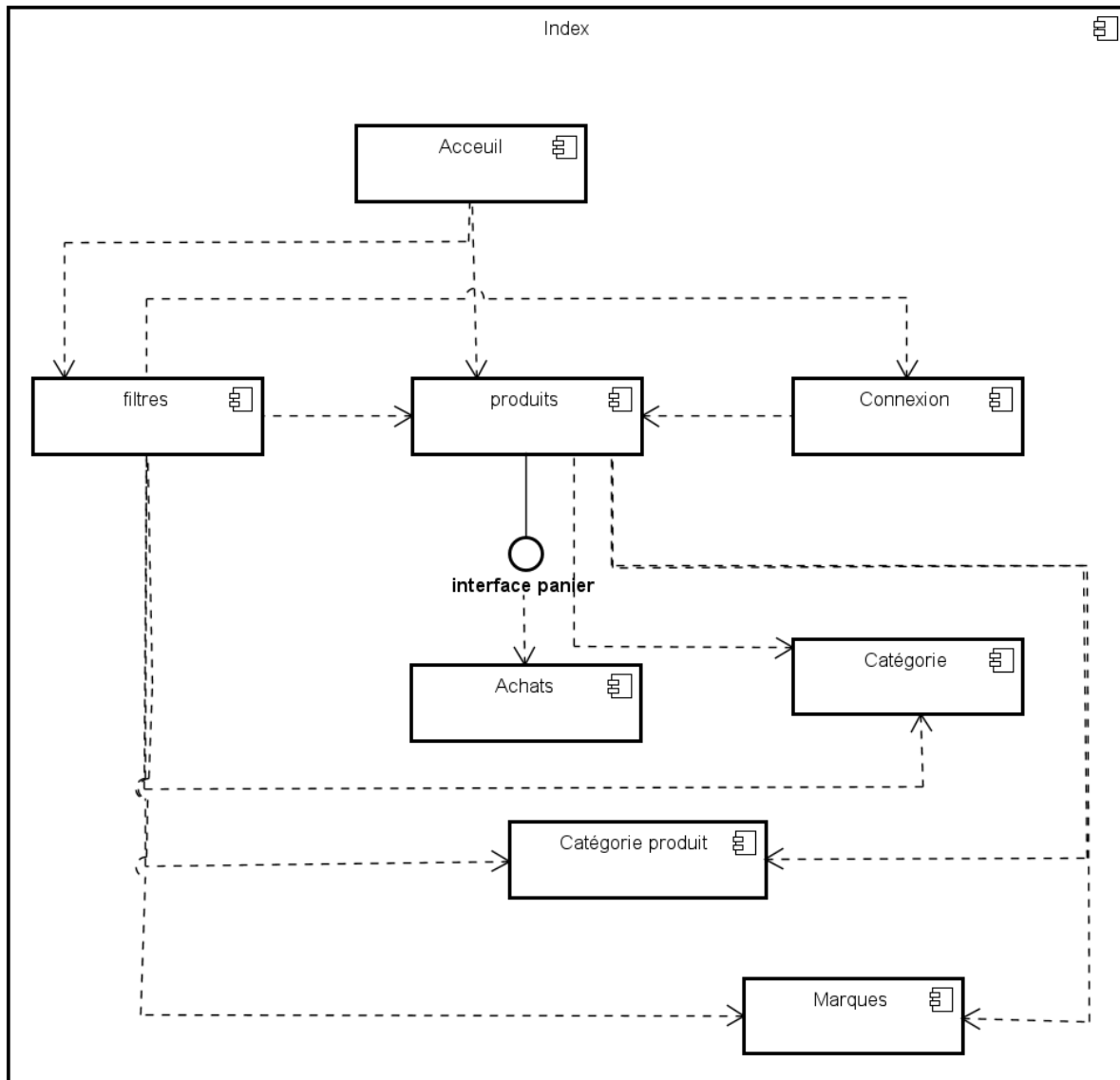
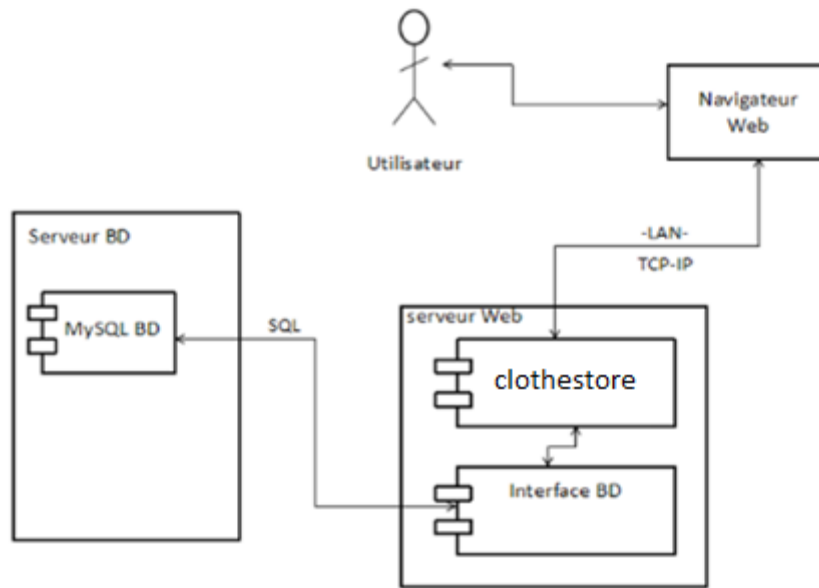


Figure 13 - Diagramme des Composants

## E - Diagramme de déploiement :

Les diagrammes de déploiement, que vous préparez généralement pendant la phase d'implémentation du développement, présentent la disposition physique des nœuds dans un système réparti, les artefacts qui sont stockés sur chaque nœud et les composants et autres éléments que les artefacts implémentent.





**Figure 14 - Diagramme de déploiement**

### 3.2.3 - Diagramme de classe :

#### 3.2.3.1 – Définition :

Le diagramme de classe est un schéma utilisé en génie logiciel pour présenter les classes et les interfaces des systèmes ainsi que les différentes relations entre celles-ci. Ce diagramme fait partie de la partie statique de l'UML car il fait abstraction des aspects temporels et dynamique. Une classe décrit les responsabilités, le comportement et le type d'un ensemble d'objets, les éléments de cet ensemble sont les instances de la classe.

**A - Relation entre les classes**

Ces relations ne sont pas exclusives au diagramme de classe, elles peuvent également s'appliquer à l'ensemble des diagrammes statiques.

**B – Héritage**

Est un principe de division par généralisation et spécialisation représenté par un trait reliant les deux classes et dont l'origine (la classe mère) se distingue de l'autre extrémité (la classe fille) par un triangle.

**C – Association**

Est une connexion sémantique entre deux classes (relation logique). Une association peut être nommée. L'invocation d'une méthode est une association.

**D – Agrégation**

Est une association avec relation de subordination représentée par un trait reliant les deux classes et dont l'origine se distingue de l'autre extrémité (la classe subordonnée) par un losange vide. Une des classes regroupe d'autres classes.

**E – Composition**

Est une agrégation avec cycle de vie dépendant : la classe composée est détruite lorsque la classe mère disparaît. L'origine de cette association est représentée par un losange plein.

**3.2.3.1 - Diagramme de classe globale :**

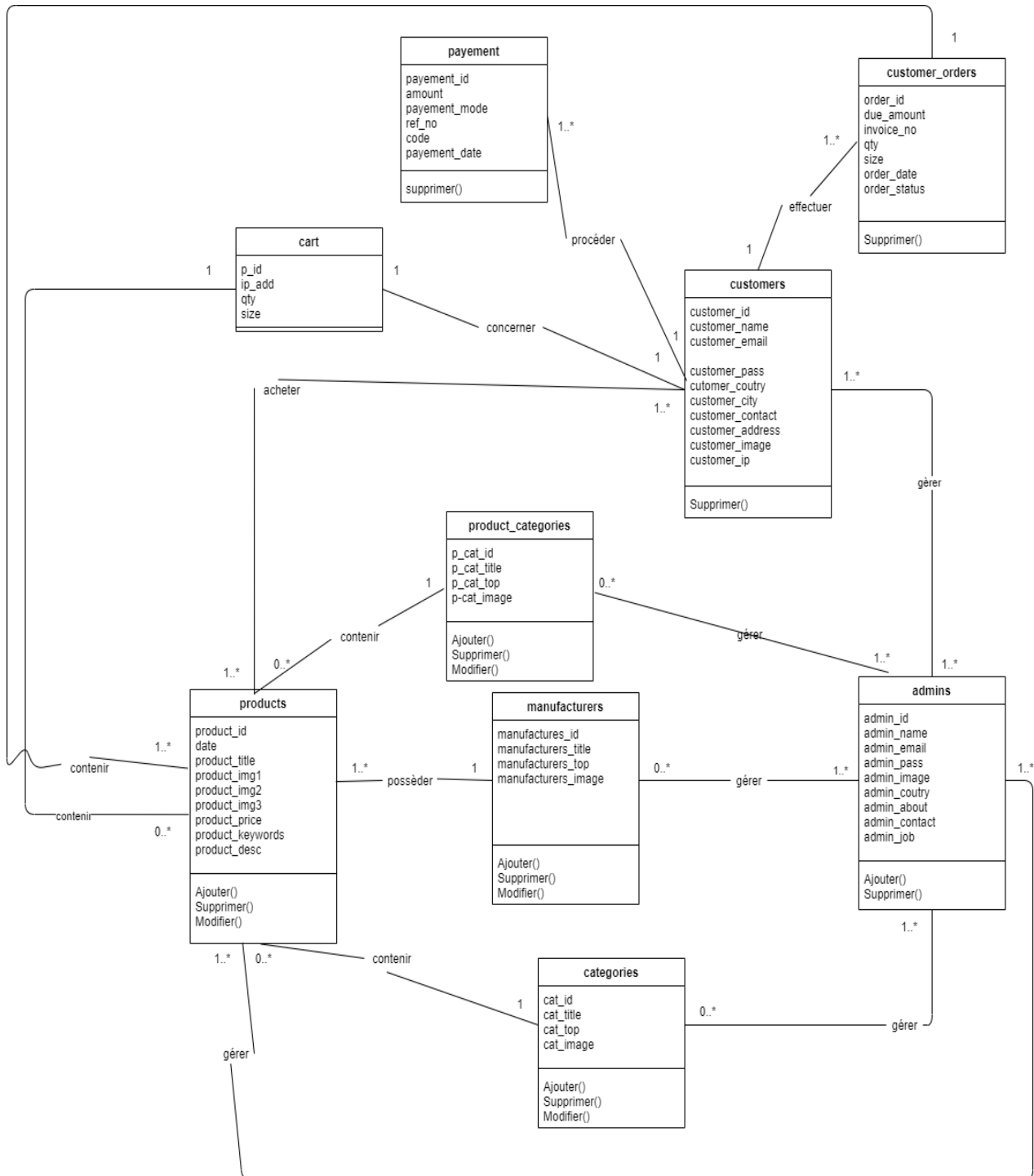


Figure 15 - Diagramme de classe globale

### 3.2.4 - Conception de la base de données

Après avoir modélisé notre application web avec les différents diagrammes offerts par le langage de modélisation UML, nous avons suivi le déroulement de chaque cas d'utilisation et nous avons utilisé les diagrammes de la classe pour en extraire les différentes données nécessaires.

Pour l'implémentation de la base de données nous aurons besoin d'élaborer un modèle logique de données qui contiendra :

### 3.2.4.1 - Les tables dans la base de données

L'application, dans tous ces cas d'initialisation, manipule les données stockées dans la base de données, qui contient les tables suivantes :

#### A - Le niveau logique de la BDD

Après avoir élaboré le schéma conceptuel de la base de données, nous élaborons le schéma relationnel de celle-ci en respectant les règles de passage du niveau conceptuelle au niveau logique.

##### Le modèle relationnel :

- **Admins**(admin\_id, admin\_name, admin\_email, admin\_pass, admin\_image, admin\_country, admin\_about, admin\_contact, admin\_job)
- **Boxes\_section**(box\_id, box\_title, box\_desc)
- **Cart**(p\_id, ip\_add, qty, size)
- **Categories**(cat\_id, cat\_title, cat\_top, cat\_image)
- **Customers**(customer\_id, customer\_name, customer\_email, customer\_pass, customer\_country, customer\_city, customer\_contact, customer\_adress, customer\_image, customer\_ip)
- **Customer\_orders**(order\_id, #customer\_id, due\_amount, invoice\_no, qty, size, order\_date, order\_status)
- **Manufacturers**(Manufacturers\_id, Manufacturers\_title, Manufacturers\_top, Manufacturers\_image)
- **Payment**(payment\_id, invoice\_no, amount, payment\_mode, ref\_no, code, payment\_date)
- **Pending\_orders**(order\_id, #customer\_id, invoice\_no, #product\_id, qty, size, orders\_status)
- **Product**(product\_id, #p\_cat\_id, cat\_id, manufacturers\_id, date, product\_title, product\_keywords, product\_desc, product\_img1, product\_img2, product\_img3)
- **Product\_categories**(p\_cat\_id, p\_cat\_title, p\_cat\_top, p\_cat\_image)
- **Slider**(slider\_id, slider\_name, slider\_image, slider\_url)
- **Terms**(term\_id, term\_title, term\_link, term\_desc)

Les clés primaires sont soulignées les clés étrangères sont précédées par #

#### B - Le niveau physique de la BDD

Le modèle physique de données est la traduction du modèle logique de données dans un langage de description de données spécifique au système de gestion des bases de données utilisé. Il décrit le modèle de stockage des données et les fonctions d'accès.


Table Admins		
#	Name	Type
1	<u>admin_id</u> 	int(10)
2	admin_name	varchar(255)
3	admin_email	varchar(255)
4	admin_pass	varchar(255)
5	admin_image	text
6	admin_country	text
7	admin_about	text
8	admin_contact	varchar(255)
9	admin_job	varchar(255)


Table boxes_section		
#	Name	Type
1	<u>box_id</u> 	int(10)
2	box_title	text
3	box_desc	text


Table cart		
#	Name	Type
1	<u>p_id</u> 	int(10)
2	ip_add	varchar(255)
3	qty	int(10)
4	size	text


Table manufacturers		
#	Name	Type
1	<u>manufacturer_id</u> 	int(10)
2	manufacturer_title	text
3	manufacturer_top	text
4	manufacturer_image	text


Table payments		
#	Name	Type
1	<u>payment_id</u> 	int(10)
2	invoice_no	int(10)
3	amount	int(10)
4	payment_mode	text
5	ref_no	int(10)
6	code	int(10)
7	payment_date	text

Table categories


#	Name	Type
1	cat_id 	int(10)
2	cat_title	text
3	cat_top	text
4	cat_image	text

Table customers


#	Name	Type
1	customer_id 	int(10)
2	customer_name	varchar(255)
3	customer_email	varchar(255)
4	customer_pass	varchar(255)
5	customer_country	text
6	customer_city	text
7	customer_contact	varchar(255)
8	customer_address	text
9	customer_image	text
10	customer_ip	varchar(100)

Table customer\_orders


#	Name	Type
1	order_id 	int(10)
2	customer_id	int(10)
3	due_amount	int(100)
4	invoice_no	int(100)
5	qty	int(10)
6	size	text
7	order_date	date
8	order_status	text

Table pending\_orders


#	Name	Type
1	order_id 	int(10)
2	customer_id	int(10)
3	invoice_no	int(10)
4	product_id	text
5	qty	int(10)
6	size	text
7	order_status	text

Table products


#	Name	Type
1	product_id 	int(10)
2	p_cat_id	int(10)
3	cat_id	int(10)
4	manufacturer_id	int(10)
5	date	timestamp
6	product_title	text
7	product_img1	text
8	product_img2	text
9	product_img3	text
10	product_price	int(10)
11	product_keywords	text
12	product_desc	text

Table product\_categories


#	Name	Type
1	p_cat_id 	int(10)
2	p_cat_title	text
3	p_cat_top	text
4	p_cat_image	text

Table slider


#	Name	Type
1	slide_id 	int(10)
2	slide_name	varchar(255)
3	slide_image	text
4	slide_url	varchar(255)

Table terms


#	Name	Type
1	term_id 	int(10)
2	term_title	varchar(100)
3	term_link	varchar(100)
4	term_desc	text

Figure 16 : Tables de la base de données

## 4 - CONCLUSION

Dans ce chapitre nous avons présenté les différents aspects analytiques et conceptuels de notre application. Nous avons commencé par la présentation de quelques détails concernant

# **Chapitre3 :**

# **RÉALISATION**

## 1. Introduction :

Ce chapitre a pour objectif majeur de présenter le produit final. C'est la phase de réalisation de ce site web dynamique qui utilise des stacks de développement spécifiques. Ce chapitre est composé de deux parties : la première partie présente l'environnement de développement, les outils qui ont servi à la réalisation de notre application, alors que la seconde partie par la présentation de ses fonctionnalités à travers de ses quelques différentes interfaces graphiques .

## 2. L'environnement de développement :

### 2.1 - Environnement Logiciel :

Lors du développement de cette application, on a utilisé, les outils logiciels suivants:

- Adobe XD.
- Les langages/bibliothèques utilisés
- Javascript.
- Xampp.
- Serveur MySQL.
- Serveur Apache.

Je vais présenter ces différents logiciels dans la section suivante :

### 2.2 - Les outils utilisés :

#### 2.2.1 - Les langages/bibliothèques utilisés :

- **HTML [2]**

HTML signifie « *HyperText Markup Language* » qu'on peut traduire par « langage de balises pour l'hypertexte ». Il est utilisé afin de créer et de représenter le contenu d'une page web et sa structure.

Ce langage fonctionne grâce à des « balises » qui sont insérées au sein d'un texte normal. Chacune de ces balises indique la signification de telle ou telle portion de texte dans le site. On parle d'« hypertexte » en référence aux liens qui connectent les pages web entre elles. C'est la mécanique originelle du « World Wide Web » que nous connaissons aujourd'hui.

- **CSS [3]**

Les CSS, *Cascading Style Sheets* (feuilles de styles en cascade), servent à mettre en forme des documents web, type page HTML ou XML, par l'intermédiaire de propriétés d'apparence (couleurs, bordures, polices, ...etc.) et de placement (largeur, hauteur, côte à côte, dessus-dessous, ...etc.).

Le rendu d'une page web peut être intégralement modifié sans aucun code supplémentaire dans la page web. Les feuilles de styles ont d'ailleurs pour objectif principal de dissocier le contenu de la page de son apparence visuelle. Ceci permet :

- De ne pas répéter dans chaque page le même code de mise-en-forme.
- D'utiliser des styles génériques, avec des noms explicites (par exemple un style encadré pour du texte ou des images).

- De pouvoir changer l'apparence d'un site web complet en ne modifiant qu'un seul fichier.
- De faciliter la lecture du code de la page.

- **PHP [4]**

Le PHP est un langage informatique utilisé sur internet. Le terme PHP est un acronyme récursif de « *HyperText Preprocessor* ». Ce langage est principalement utilisé pour produire un site web dynamique. Il est courant que ce langage soit associé à une base de données, tel que MySQL.

Exécuté du côté serveur (l'endroit où est hébergé le site), il n'y a pas besoin aux visiteurs d'avoir des logiciels ou plugins particulier. Néanmoins, les webmasters qui souhaitent développer un site en PHP doivent s'assurer que l'hébergeur prend en compte ce langage. Lorsqu'une page PHP est exécuté par le serveur, celui-ci renvoie généralement au client (aux visiteurs du site) une page web qui peut contenir du HTML, XHTML, CSS, JavaScript.

- **SQL [5]**

MySQL est un système de gestion de base de données (SGBD). Selon le type d'application, la 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.

MySQL est un serveur de bases de données relationnelles SQL développé dans un souci de performances élevées en lecture, ce qui signifie qu'il est davantage orienté vers le service de données déjà en place que vers celui de mises à jour fréquentes et fortement sécurisées. Il est multi-threads et multi-utilisateurs.

C'est un logiciel libre développé sous double licence en fonction de l'utilisation qui en est faite : dans un produit libre ou dans un produit propriétaire. Dans ce dernier cas, la licence est payante, sinon c'est la licence publique générale GNU (GPL) qui s'applique. Ce type de licence double est utilisé par d'autres produits comme le framework de développement de logiciels (pour les versions antérieures à la 4.5).

Le couple PHP/MySQL est très utilisé par les sites Web et proposé par la majorité des hébergeurs Web. Plus de la moitié des sites Web fonctionnent sous Apache, qui est le plus souvent utilisé conjointement avec PHP et MySQL.

- **JavaScript [6]**

Le JavaScript est un langage informatique utilisé sur les pages web. Ce langage a la particularité de s'activer sur le poste client, en d'autres mots c'est votre ordinateur qui va recevoir le code et qui devra l'exécuter. C'est en opposition à d'autres langages qui sont activés côté serveur. L'exécution du code est effectuée par votre navigateur internet tel que Firefox ou Internet Explorer.

La particularité du JavaScript consiste à créer des petits scripts sur une page HTML dans le but d'ajouter une petite animation ou un effet particulier sur la page. L'intérêt du JavaScript est d'exécuter un code sans avoir à recharger une nouvelle fois la page.



- **jQuery [7]**

jQuery est un Framework JavaScript sous licence libre qui permet de faciliter des fonctionnalités communes de JavaScript.

L'utilisation de cette bibliothèque permet de gagner du temps en développement lors de l'interaction sur le code HTML d'une page web, l'AJAX ou la gestion des événements.

jQuery possède par la même occasion l'avantage d'être utilisable sur plusieurs navigateurs web (Internet Explorer, Firefox ...).

La bibliothèque jQuery possède les fonctionnalités suivantes :

- Manipulation du DOM (HTML ou CSS)
- Gestion des événements (clic, survol, soumettre un formulaire ...)
- AJAX
- Effet d'animation

- **AJAX [8]**

AJAX « *Asynchronous JavaScript And XML* » n'est ni une technologie ni un langage de programmation ; AJAX est un concept de programmation Web reposant sur plusieurs technologies comme le JavaScript et le XML – d'où le nom AJAX. A l'heure actuelle, le XML tend à être délaissé au profit du JSON, ce qui explique que certains puristes utilisent l'acronyme *AJAJ* – dont la prononciation laisse plutôt à désirer.

L'idée même d'AJAX est de faire communiquer une page Web avec un serveur Web sans occasionner le rechargement de la page. C'est la raison pour laquelle JavaScript est utilisé, car c'est lui qui va se charger d'établir la connexion entre la page Web et le serveur.

## 2.2.2 - Logiciels utilisés :

- **XAMPP Control Panel [9]**

C'est une plate-forme de développement Web de type Xampp , permettant de fournir une distribution facile à installer. Ainsi, les développeurs peuvent se lancer facilement dans le monde d'Apache. Pour le rendre encore plus pratique, XAMPP est livré avec toutes les fonctionnalités pour faire fonctionner en local des scripts PHP.

Xampp n'est pas en soi un logiciel, mais un environnement comprenant deux serveurs (Apache et MySQL), un interpréteur de script (PHP), ainsi que phpMyAdmin pour l'administration Web des bases MySQL.



Figure 17 - Interface de Xampp Control panel.

- **VisualStudio Code [10]**

C'est un éditeur de code open-source, gratuit et multi-plateforme (Windows, Mac et Linux), développé par Microsoft, à ne pas confondre avec Visual Studio, l'IDE propriétaire de Microsoft.

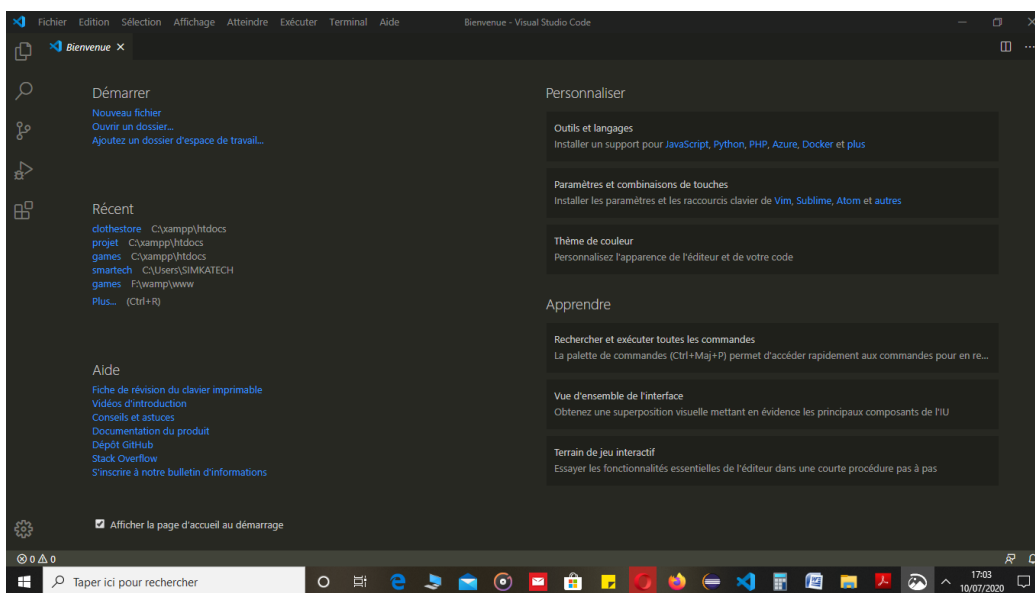


Figure 18 – Interface VisualStudio Code.

### 3 - Les principales interfaces graphiques :

Cette partie permet de je mettre dans les conditions réelles d'utilisation de l'application.

#### 3.1 - Interface client :

Dans les paragraphes qui suivent, je vais exposer différentes situations que peut rencontrer le client.

- **Accueil :**

C'est la première page visualisée par les clients, c'est la page principale du site et elle est aussi appelée **Accueil**.

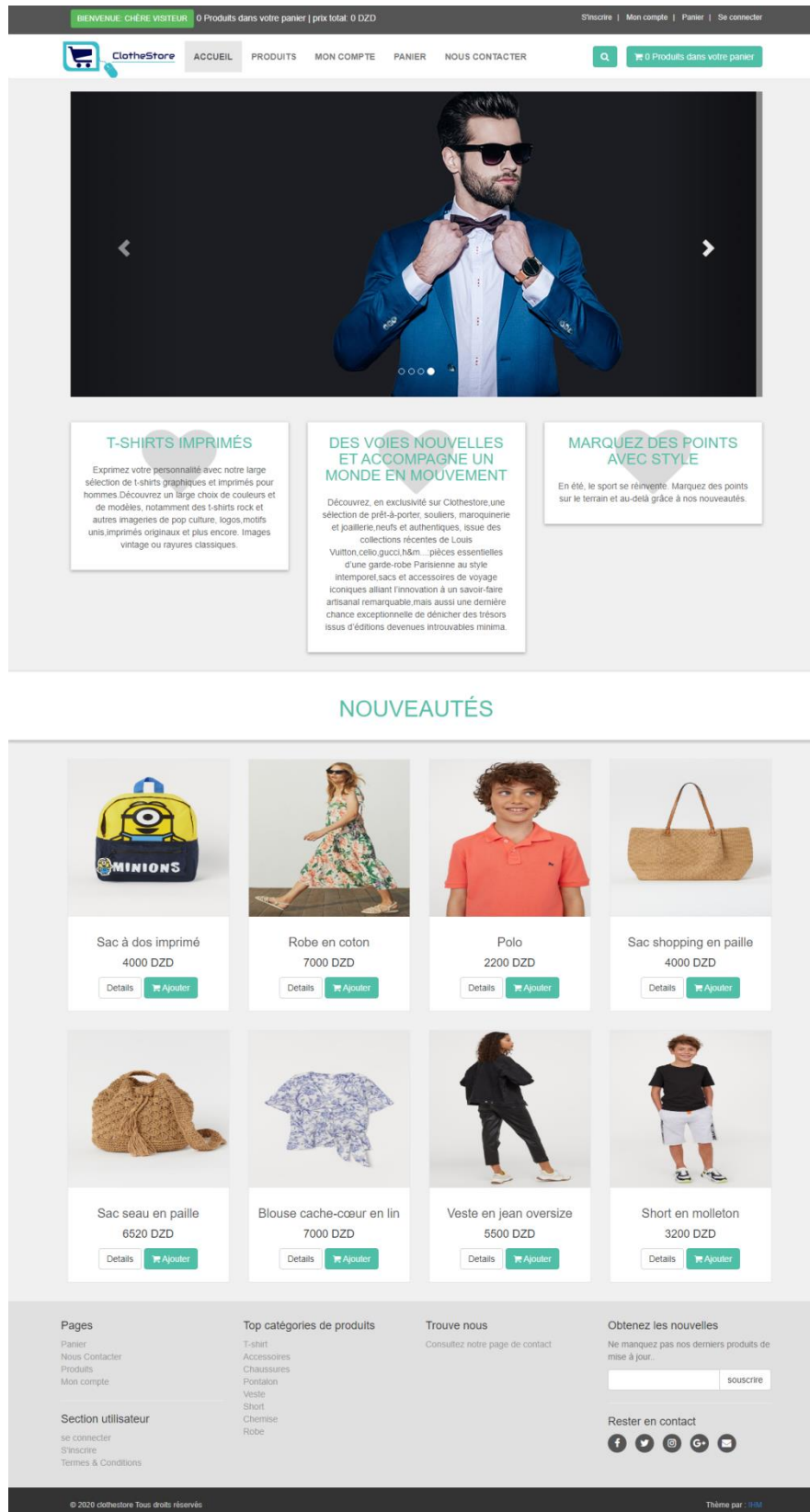
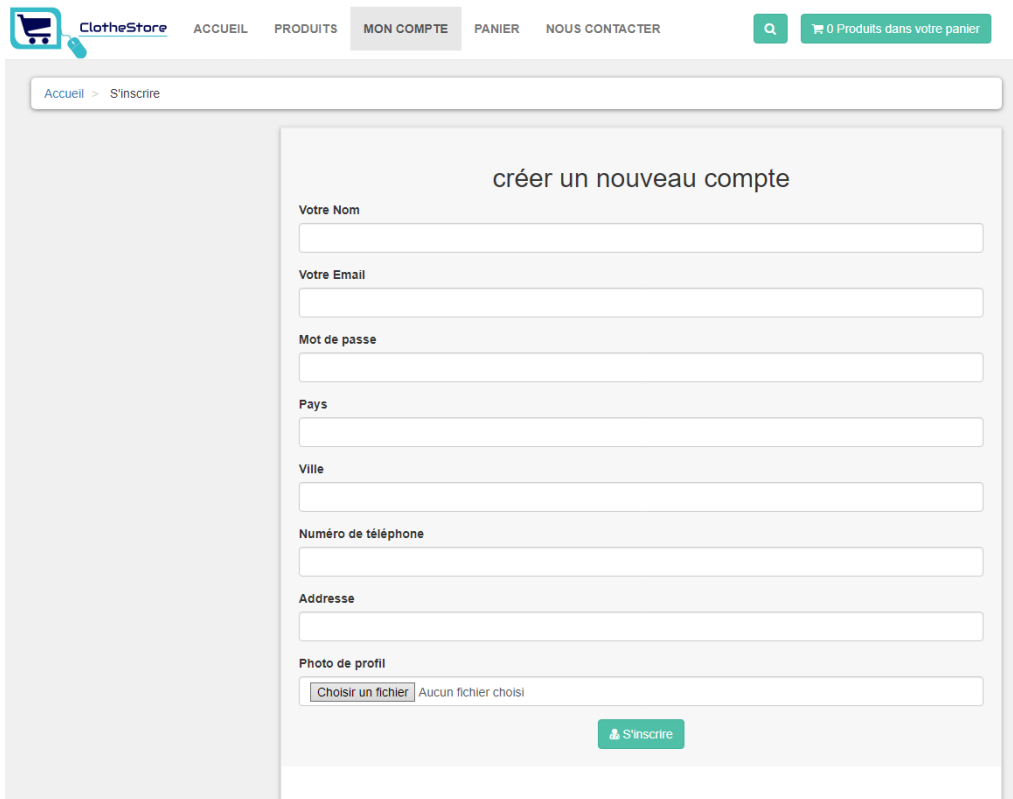


Figure 19 : Accueil client

### • Inscription :

Cette page permet à tout nouveau client de s'inscrire dans ce base de données pour pouvoir, par la suite, effectuer un achat.

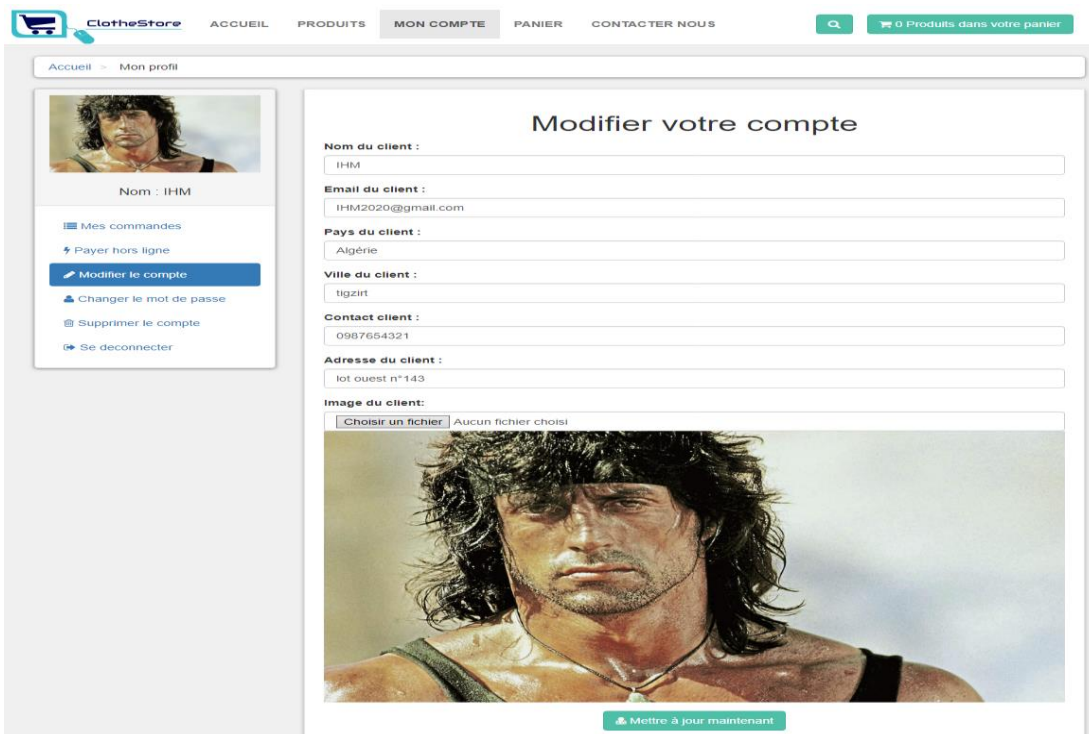
Les champs de saisie contiennent un nombre de caractères .La page n'est validée que si tous les champs de saisie sont conformes aux exigences demandées.



**Figure 20 : Inscription**

- La modification des coordonnées du client :

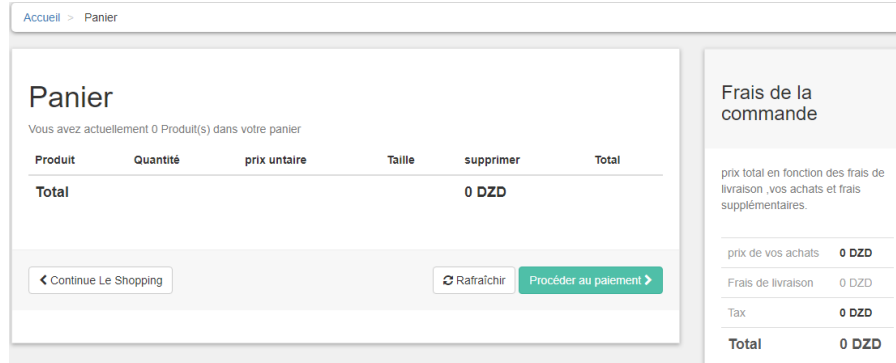
Cette page permet à un client, qui s'est préalablement connecté, de modifier ses coordonnées.



**Figure 21 :  
Modifier  
coordonné**

- **Le panier :**

Cette page permet également au client de vérifier la quantité choisie et le montant total de son panier, ensuite, il peut commander ses produits mais il faut absolument se connecter pour pouvoir passer sa commande.

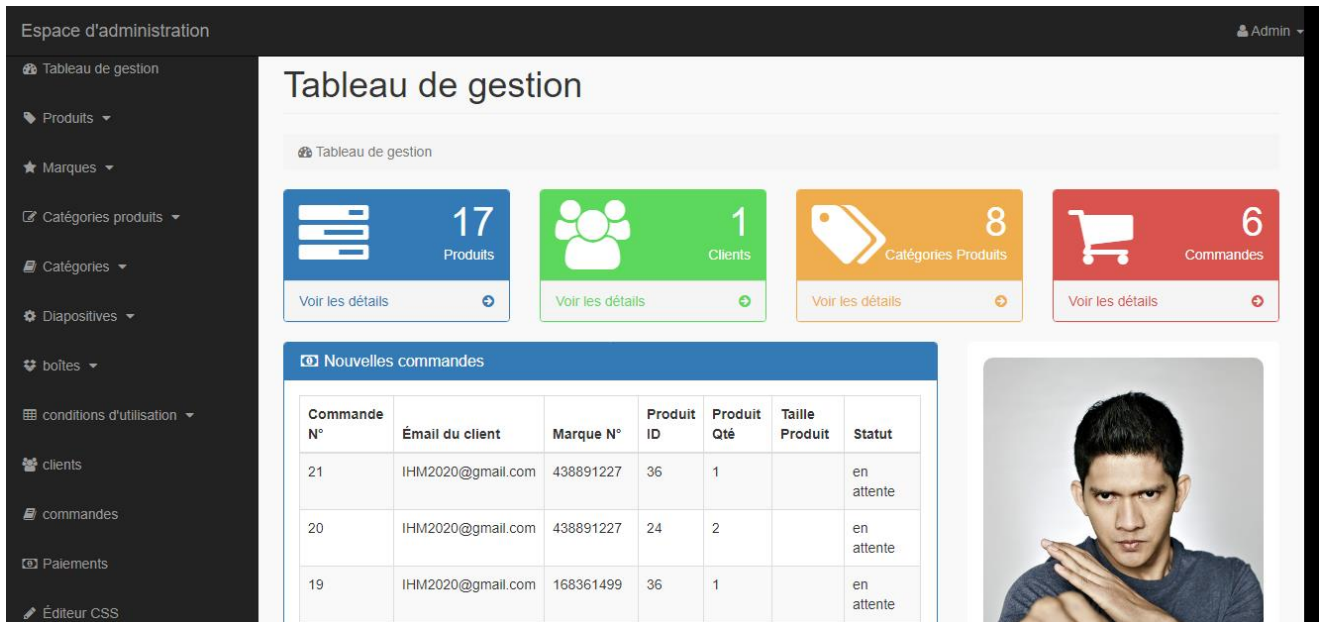


**Figure22: Panier**

### 3.2.Interface administrateur :

- **Accueil :**

L'administrateur tape son nom et mot de passe pour accéder à cette interface, il a le droit d'administrer les actions possibles dans le site comme (modifier, ajouter et supprimer des produits ou des catégories .....).



**Figure 23: accueil administrateur**

- **Ajouter produit :**

Cette page permet à l'administrateur de rajouter des produits dans la base actuelle. Suivant le type du produit, il devra entrer différents champs afin de renseigner l'article à rajouter. Chaque champ de

saisie est contrôlé lors de la validation de la page et l'internaute est averti par les alertes de couleur rouge si un champ n'est pas rempli.

Tableau de gestion / Ajouter un produit

Ajouter un produit

Nom produit

Marque

Catégorie du produit

Catégorie

Image du produit 1  Aucun fichier choisi

Image du produit 2  Aucun fichier choisi

Image du produit 3  Aucun fichier choisi

Prix du produit

clé du produit

Description du produit

File Edit View Format

Formats B I

POWERED BY TINYMCE

Ajouter produit




**Figure 24 : Ajouter produit**

- **Lister les administrateurs :**

Cette tâche est associée au propriétaire de site qui a le droit de supprimer ou de modifier les administrateurs.

Tableau de gestion / Voir administrateurs

Voir administrateurs

N°	Nom administrateurs	Image administrateurs	E-Mail administrateurs	Pays administrateurs	Pays administrateurs	Contact administrateurs	Modifier	Supprimer
1	Admin		admin@gmail.com	Algérie	développeur / étudiant	061806833157	 Modifier	 Supprimer

**Figure 25 : Liste des administrateurs**

## 4. Conclusion :

La partie de réalisation détermine une idée plus claire sur les tâches qui sont réalisées dans ce site web par la présentation des interfaces graphiques. Enfin avec ce chapitre, on termine la phase de développement de ce site.

### Conclusion générale

Ce projet fin d'étude consiste à concevoir un site web dynamique qui permet de réaliser le commerce électronique de vêtements et accessoires .

C'est une application presque finalisée et accompagnée de tous les documentations technique et conceptuelle nécessaire à sa bonne évolution qui permet aux utilisateurs de consulter et d'acheter des articles en ligne ou hors ligne.

Pour concevoir ce travail on a présenté premièrement le cadre de ce projet, puis on a analysé l'étude théorique sur le langage de modélisation UML et son extension pour le web afin de bien concevoir notre application.

En second, on a montré la phase de conception d'une part ,et on a traité toutes les phases nécessaires à la réalisation de ce application et répondre aux objectifs fixés tout en utilisant le langage HTML pour créer les pages web et PHP et JavaScript pour dynamiser ces pages et SQL pour interroger la base de données d'autre part.

Cette phase nous a permis :

- De mieux manipuler les langages PHP, HTML et Java Script, on a approfondi connaissances sur le langage SQL avec le MySQL.
- D'approfondir nos connaissances théorique et pratique en rapport avec le web et les techniques de programmation web.
- D'acquérir de nouvelles connaissances sur les langages de modélisation en particulier l'UML.
- D'acquérir de nouvelles connaissances sur les bases de données et les systèmes de gestion de base de données en particulier SQL/MySQL.
- De se familiariser avec les outils d'analyse et de conception UML.

Par ailleurs, l'objectif principal de ce projet était la découverte du monde du ecommerce et dans cette optique, ce projet a totalement répondu à nos attentes. Bien sur des améliorations pourraient aussi être apportées à ce site par exemple dans le cas d'une réelle utilisation commerciale du site, proposer une connexion sécurisée lors du paiement de la commande ou de la consultation du compte.

Enfin, la réalisation de ce projet de travail en équipe sur une durée limitée est un bon entraînement pour notre futur projet d'étude , mais aussi pour ce futur métier.

## BIBLIOGRAPHIE

### 1 – BIBLIOGRAPHIE

#### Livre :

- Concevez votre site web avec PHP et MySQL (le livre du zéro).
- Dynamisez vos sites web avec JavaScript ( Livre du Zéro).
- PHP 5 Avancé d'Eric Daspet & Cyril Pierre de Geyer.
- UML2 de Laurent AUDIBERT Edition 2007-2008

#### Thèses :

- Conception et réalisation d'un site web pour la gestion des ventes et promotions de matériels informatiques Cas : « GLOBAL INFO LAGA » réalisé par :
  - Mme C. Chérifi
  - Mr Messas Kouseila
  - Mr Meghari Aghiles
  - Mr Djouzi kheyreddine

### 2 – WEBOGRAPHIE :

- <https://www.pierre-giraud.com/bootstrap-apprendre-cours>
- <https://www.journaldunet.fr/web-tech/dictionnaire-du-webmastering/1203603-sql-structured-query-language-definition-traduction-et-acteurs>
- <http://glossaire.infowebmaster.fr/javascript>
- <http://glossaire.infowebmaster.fr/jquery>
- <https://openclassrooms.com/fr/courses/245710-ajax-et-lechange-de-donnees-en-javascript/244425-le-concept-daja>
- <http://dictionnaire.sensagent.leparisien.fr/wam>
- [pserver/fr-fr/](http://pserver.fr-fr/)
- <http://www.memoireonline.com>
- <http://www.w3schools.com>
- <http://www.openclassroom.com>
- <http://fr.wikipedia.org/wiki/MySQL>
- <http://fr.wikipedia.org/wiki/EasyPHP>
- <http://fr.wikipedia.org/wiki/PhpMyAdmin>
- <http://fr.wikipedia.org/wiki/JavaScript>
- <http://www.agilemodeling.com>