

Conception d'un site web de  
e- commerce

Réalisé par :

CHOUAIB MASSA

WADIE BOUKHARI

AHMED DARHNANI

Année :

2023-2024

# Conception

## Introduction

Dans le cycle de vie de notre projet, la conception représente une phase primordiale et déterminante pour produire une application de haute qualité. C'est dans ce stade que nous devons clarifier en premier lieu la vue globale, en décrivant l'architecture générale que nous allons suivre dans la partie réalisation de notre projet. Puis, dans un deuxième lieu nous allons détailler notre choix conceptuel à travers plusieurs types de diagrammes.

### I. Conception Générale

#### 1. Cycle de vie :

##### 1.1 Définition

Le cycle de vie d'une application comprend toutes les étapes depuis sa conception et sa réalisation jusqu'à sa mise en œuvre. L'objectif d'un tel découpage est de permettre de définir des jalons intermédiaires permettant la validation du développement du logiciel et la vérification de son processus de développement.

L'origine de ce découpage provient du constat que les erreurs ont un coût si élevé qu'elles sont détectées tardivement dans le processus de réalisation. Le cycle de vie permet de détecter les erreurs le plus tôt possible.

##### 1.2 Les activités d'un cycle de vie

Le cycle de vie suivi pour réaliser un site e-commercial, comprend généralement au minima les activités suivantes :

- **Spécification des besoins:** elle consiste à définir la finalité du projet et son intégration dans une stratégie globale.
- **Conception générale:** dans cette activité, il s'agit de la préparation de l'architecture générale du logiciel.

- **Conception détaillée:** elle consiste à définir précisément chaque sous-ensemble du logiciel.
- **Développement:** (Implémentation ou programmation) il s'agit d'une traduction des fonctionnalités définies dans la phase de conception en langage de programmation.
- **Tests unitaires:** ils permettent de vérifier individuellement que chaque sous-ensemble du logiciel est implémenté conformément aux normes définies dans la conception.
- **Intégration:** dite aussi tests systèmes, elle consiste à vérifier que le logiciel correspond exactement au cahier des charges du projet en obtenant enfin un manuel d'utilisation bien détaillé aux utilisateurs.
- **Validation:** c'est-à-dire la validation de conformité du site avec les buts spécifiés à la première étape du cycle de vie.

## 2. Les diagrammes des cas d'utilisation.

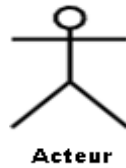
### Définition

Les rôles des diagrammes de cas d'utilisation sont de recueillir, d'analyser et d'organiser les besoins, ainsi que de recenser les grandes fonctionnalités d'un système. Il s'agit donc de la première étape UML pour la conception d'un système.

Un diagramme de cas d'utilisation capture le comportement d'un système, d'un sous-système, d'une classe ou d'un composant tel qu'un utilisateur extérieur le voit. Il scinde la fonctionnalité du système en unités cohérentes, les cas d'utilisation, ayant un sens pour les acteurs. Ainsi ces cas d'utilisation permettent d'exprimer le besoin des utilisateurs d'un système, ils sont donc une vision orientée. Il ne faut jamais négliger cette première étape pour produire un site web conforme aux attentes des utilisateurs ciblés. Pour élaborer les cas d'utilisation, il faut se fonder sur des entretiens avec les utilisateurs.

Le diagramme de cas se compose de trois éléments principaux :

## 2- Composition du diagramme de cas d'utilisation



**Un Acteur :** c'est l'idéalisation d'un rôle joué par une personne externe, un processus ou une chose qui interagit avec un système. Il se représente par un petit bonhomme avec son nom inscrit dessous.



**Un cas d'utilisation :** c'est une unité cohérente représentant une fonctionnalité visible de l'extérieur. Il réalise un service de bout en bout, avec un déclenchement, un déroulement et une fin, pour l'acteur qui l'initie.

Un cas d'utilisation modélise donc un service rendu par le système, sans imposer le mode de réalisation de ce service. Il représente par une ellipse contenant le nom du cas (un verbe à l'infinitif), et optionnellement, au-dessus du nom, un stéréotype.

**Les relations :** Trois types de relations sont pris en charge par la norme UML et sont graphiquement représentées par des types particuliers de ces relations. Les relations indiquent que le cas d'utilisation source présente les mêmes conditions d'exécution que le cas issu. Une relation simple entre un acteur et une utilisation est un trait simple.

### 2.1 Les acteurs de notre projet

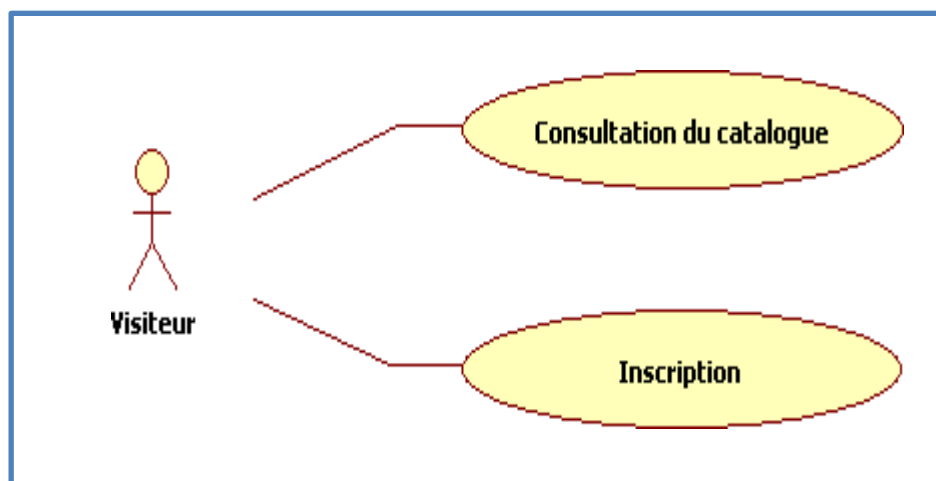
**Le visiteur :** c'est un individu qui est entrain de fouiller sur le net, cherchant un produit pour l'acheter ou pour avoir une idée sur les modèles et les prix. Jusqu'au ce stade c'est un utilisateur inconnu donc il n'est pas encore un client.

**Le Client** : cet acteur est un visiteur ayant déjà créé un compte sur notre site, il peut donc suivre le processus d'achat des produits en toute sécurité sachant que notre système doit être l'unique responsable de la confidentialité des données personnelles de ses clients.

**L'administrateur** : pour les sites web on l'appelle généralement « *le webmaster* ». C'est celui qui assure le dynamisme du site et veille sur les mises à jour des produits, de leurs prix, de leurs disponibilités, de la gestion des paiements et la gestion des livraisons.

## 2.2 Diagrammes de cas d'utilisation de notre site web

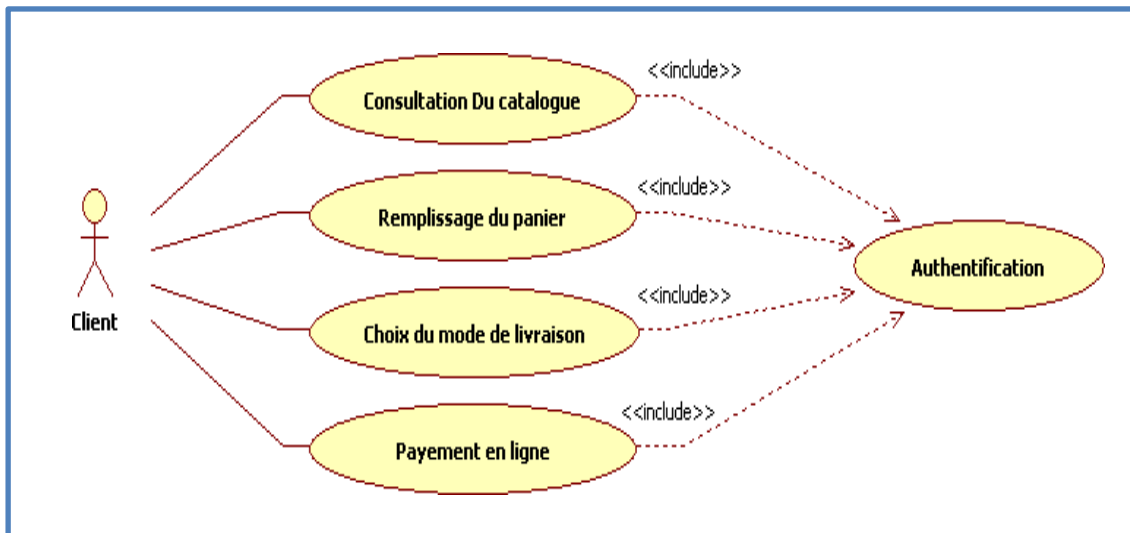
### a. Diagramme de cas d'un visiteur



**Figure 13: Cas d'utilisation d'un visiteur**

Avant de devenir client, un internaute ne possède que la possibilité de consulter le catalogue des produits disponibles dans le stock du fournisseur et la possibilité de s'inscrire pour devenir client sur notre site web.

### a. Diagramme de cas d'un client

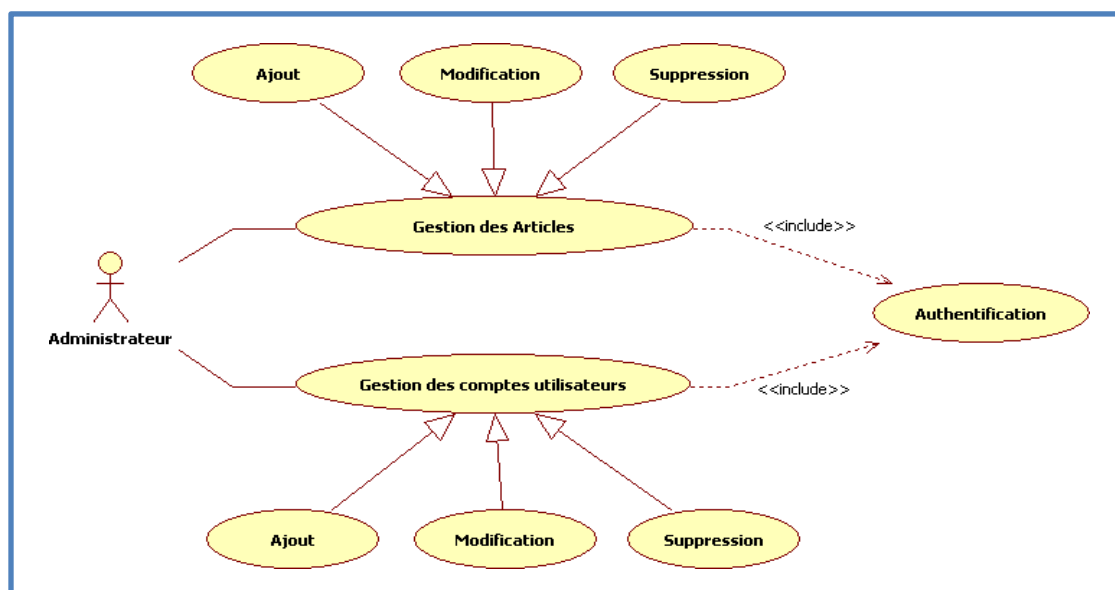


**Figure 14: Cas d'utilisation d'un client**

Après l'inscription, le visiteur devient client. Il est donc apte de continuer toute une procédure d'achat en ligne sur notre site.

#### b. Diagramme de cas du webmaster du site web

Le terme webmaster de site web désigne communément celui qui est chargé d'un site web. Il gère toute la mise en place technique et Parfois la mission éditoriale, il doit gérer au jour le jour la technique et mettre à jour le contenu du site web.



**Figure 15: Cas d'utilisation d'un webmaster ou administrateur**

### 3 Diagrammes des séquences

#### Définition

Un diagramme de séquences est un diagramme d'interaction qui expose en détail la façon dont les opérations sont effectuées : quels messages sont envoyés et quand ils le sont.

Les diagrammes de séquences sont organisés en fonction du temps qui s'écoule au fur et à mesure que nous parcourons la page.

Les objets impliqués dans l'opération sont répertoriés de gauche à droite en fonction du moment où ils prennent part dans la séquence.

#### 3-1 Composition d'un diagramme de séquences

Ce type des diagrammes est composé par les éléments suivants :

**Les lignes de vie :** Une ligne verticale qui représente la séquence des événements, produite par un participant, pendant une interaction, alors que le temps progresse en bas de ligne.

Ce participant peut être une instance d'une classe, un composant ou un acteur.

**Les messages :** deux types de messages dans le diagramme de séquences, le premier est dit message synchrone utilisé pour représenter des appels de fonction ordinaires dans un programme, le deuxième est appelé message asynchrone, étant utilisé pour représenter la communication entre des threads distincts ou la création d'un nouveau thread.

**Les occurrences d'exécution :** représente la période d'exécution d'une opération.

**Les commentaires :** Un commentaire peut être joint à tout point sur une ligne de vie.

**Les itérations** : représente un message de réponse suite à une question de vérification.

### 3.2 Les diagrammes de séquences de notre site web

#### a. Diagramme de séquences d'inscription

Pour bien profiter des privilèges Dédiés aux clients, un visiteur doit d'abords entamer la phase d'inscription avec succès et pour cela il faut qu'il passe par l'ensemble des séquences que nous allons simplifier par le schéma

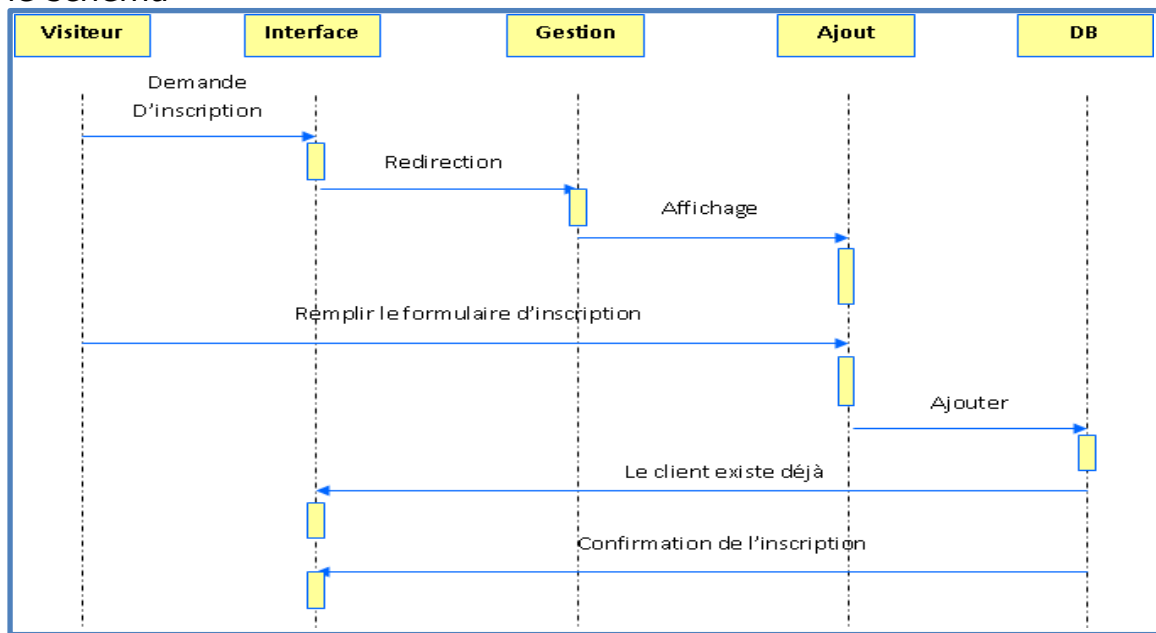


Figure 19: Diagramme de séquences de l'inscription

- Le visiteur demande le formulaire d'inscription.
- Le formulaire s'affiche.
- Le visiteur remplit le formulaire.
- Une vérification de l'existence du client dans la base se lance.
- Si le client existe déjà un message d'erreur s'affiche.
- Si c'est un nouveau client confirmation de l'inscription s'affiche.

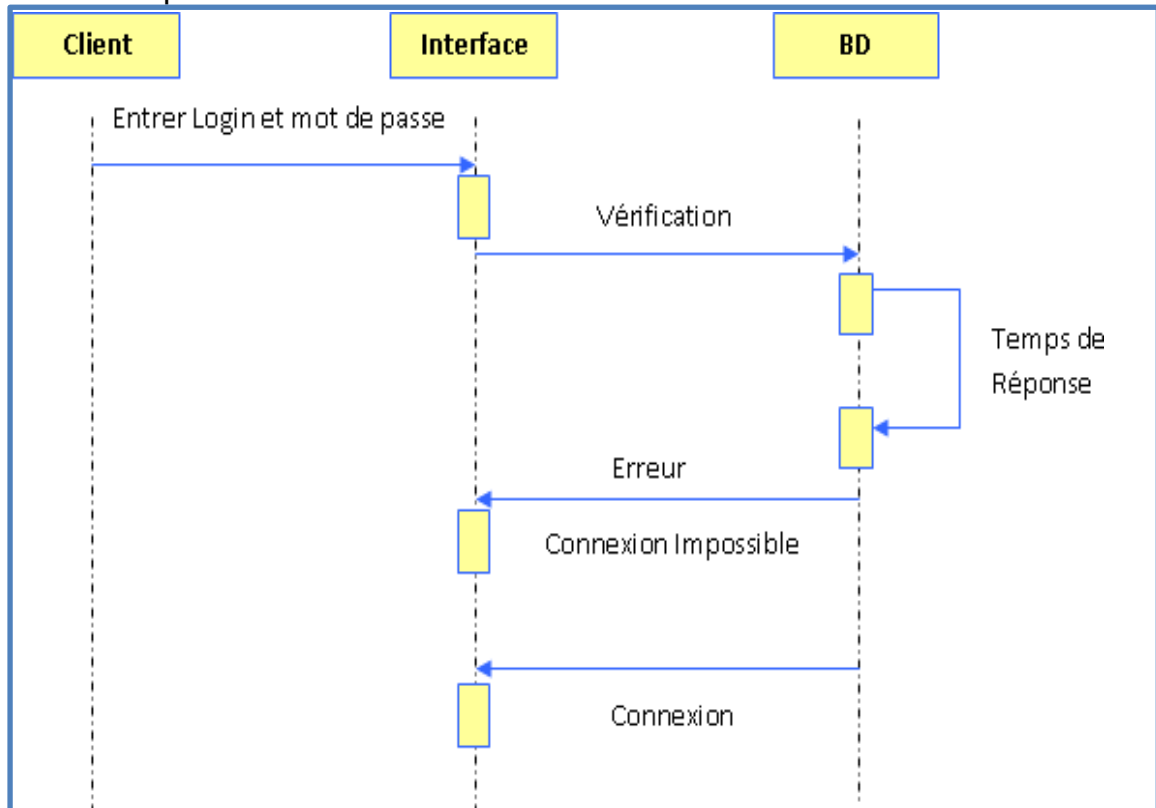
#### b. Diagramme de séquences d'authentification

Avant d'atteindre la phase d'authentification, notre visiteur est une personne présente sur notre site web d'une façon anonyme, d'où il devient indispensable d'entrer son login et son mot de passe. Puis, tout au long de sa navigation, il n'a la possibilité d'accéder qu'aux



services dont il est autorisé.

Le schéma suivant va vous montrer les séquences à effectuer pour entamer la phase d'authentification.

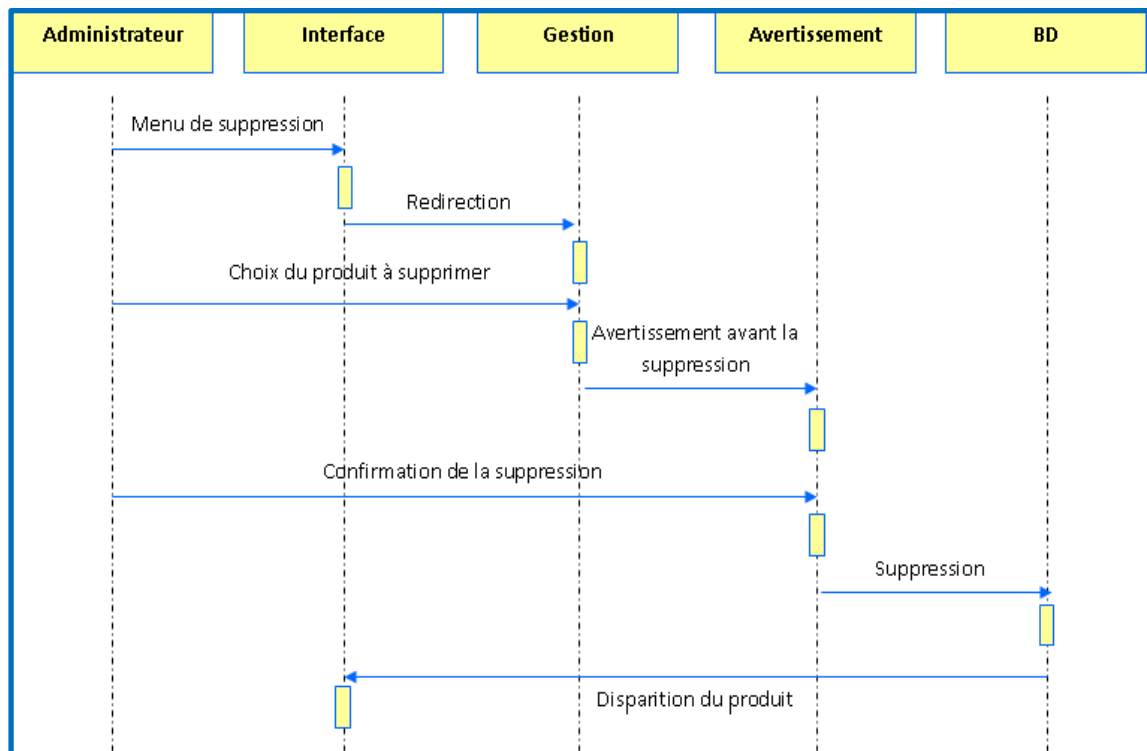


**Figure 20: Diagramme de séquences d'authentification**

- Le client entre son login et son mot de passe.
- Une vérification se lance dans la base de données.
- Après un temps de réponse ou l'authentification se valide ou un message d'erreur s'affiche

#### a. Diagramme de séquences de suppression d'un article

Parmi les scénarios dont l'administrateur est en charge nous pouvons mentionner la gestion des produits exposés sur notre site web telles que La consultation, l'ajout, la modification et la suppression que nous allons montrer dans le diagramme de séquence suivant.



**Figure 21: Diagramme de séquence de suppression d'un article**

- L'administrateur choisit l'interface de suppression.
- Le menu de suppression s'affiche.
- L'administrateur doit sélectionner le produit qu'il désire supprimer.
- Le système averti l'administrateur de l'opération de suppression.
- L'administrateur confirme la suppression.
- L'opération de suppression se termine avec succès.
- Le produit en question se disparaît définitivement de la base de données.

#### 4 Diagramme de classes

##### Définition :

Un diagramme de classes UML décrit les structures d'objets et d'informations utilisées sur notre site web, à la fois en interne et en communication avec ses utilisateurs. Il décrit les informations sans faire référence à une implémentation particulière. Ses classes et

relations peuvent être implémentées de nombreuses manières, comme les tables de bases de données, les nœuds XML ou encore les compositions d'objets logiciels.

#### 1.4 La composition d'un diagramme de classes

En général un diagramme de classe peut contenir les éléments suivants :

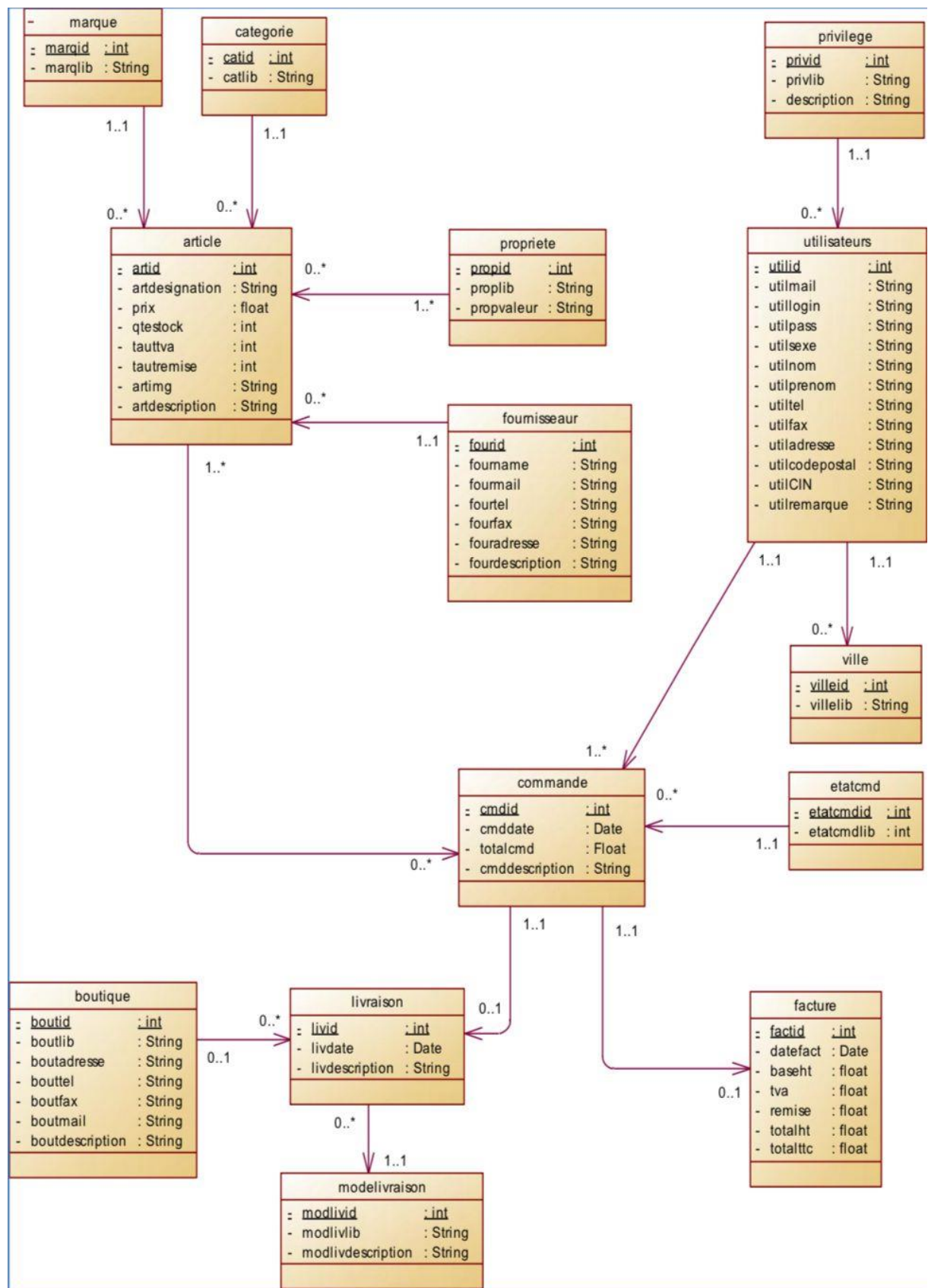
**Les classes:** une classe représente la description formelle d'un ensemble d'objets ayant une sémantique et des caractéristiques communes. Elle est représentée en utilisant un rectangle divisé en trois sections.

La section supérieure est le nom de la classe, la section centrale définit les propriétés de la classe alors que la section du bas énumère les méthodes de la classe.

**Les associations :** une association est une relation entre deux classes (association binaire) ou plus (association n-aire), qui décrit les connexions structurelles entre leurs instances. Une association indique donc que des liens peuvent exister entre des instances des classes associées.

**Les attributs :** les attributs représentent les données encapsulées dans les objets des classes. Chacune de ces informations est définie par un nom, un type de données, une visibilité et peut être initialisé. Le nom de l'attribut doit être unique dans la classe.

## Notre diagramme des classes



**article** (artid, #marqid, #catid, #fourid, artdesignation, pri  
x, qtestock, tauttva, tautremise, artimg, artdescription)

**categorie** (catid, catlib)

**marquee** (marqid, marqlib)

**fournisseur** (fourid, fourname, fourmail, fourtel, fourfax,  
fouradresse, fourdescription)

**boutique** (boutid ,boutlib, boutadresse, bouttel, boutfax,  
boutmail, boutdescription)

**utilisateur** (utilid, #privid, #villeid, utilmail, utillogin, utilpass,  
utilsexe, utilnom, utilprenom, utiltel, utilfax, utiladresse,  
utilcodepostal, utilCIN, utilremarque)

**privilege** (privid, privlib, description)

**ville** (villeid, villelib)

**propriete** (propid, proplib, propvaleur)

**facture** (factid, #cmdid, datefact, baseht, tva, remise, totalht,  
totalttc)

**commande** (cmdid, #utilid, #etatcmdid, cmddate,  
totalcmd, cmddescription)

**etatcmd** (etatcmdid, etatcmdlib)

**lignecmd** (artid, cmdid)

**ligneprop** (propid, artid)

**livraison** (livid, #modlivid, #boutid, #cmdid,  
livdate, livdescription)

**modèleivraison** (modlivid, modlivlib, modlivdescription)

### *Conclusion*

Nous venons de terminer cette partie de conception, qui consiste à déterminer aussi bien les méthodes de travail que les chartes graphiques de notre site web avec ses parties statiques et dynamiques.

Dans le chapitre suivant nous allons aborder la dernière partie qui représente la partie réalisation de notre site web, en se basant sur les mécanismes et les solutions déterminés dans la phase de conception.