UNIVERSITÉ D'ÉVRY – VAL D'ESSONNE
Département d'informatique

RAPPORT DE PROJET

de 3^{e} année de la licence d'informatique
parcours MIAGE
soutenu par

Myriam Fouda, Sema Sunner, Jennifer Testu

le 29 Août 2017

Gestion des commandes

Entreprise destinataire du projet
Autour de la moto

Année universitaire 2016-2017

Table des matières

1 Introduction

Etudiantes en L3 MIAGE à l'Université d'Evry Val d'Essonne, nous avions un projet remplaçant le stage obligatoire. Ce projet a pour objectif de mettre en oeuvre les connaissances acquises durant les trois années de licence.
Ce projet consiste en la création d'un site de vente en ligne ainsi qu'une application de gestion de commandes pour l'entreprise Autour de la Moto pour l'autoentrepreneur Monique Sommaï qui commercialise des produits du monde de la moto (vêtements, accessoires et équipements).
Le projet a duré le temps de l'été 2017 en commençant le 15 mai, jusqu'à sa présentation orale le 29 août.
Le site de vente en ligne permet à des clients potentiels de consulter la liste des produits ainsi que d'effectuer des achats, tandis que l'application de gestion de commandes permet à l'autoentrepreneur de gérer les commandes à traiter ainsi que ses stocks.
Premièrement, nous commencerons par présenter le contexte du stage en présentant tout d'abord l'entreprise, puis son activité.
Par la suite, nous présenterons ce qui a été réalisé dans le cadre de ce projet. Puis nous ferons un bilan du travail final avant de conclure.

2 Cahier des charges

2.1 Contenu

— Diagrammes UML et leurs descriptifs dans la partie des travaux réalisés
— Images représentatives du site et de l'application

2.2 Technologies utilisées

Pour réaliser ce projet, nous avons utilisé les outils :
— Modelio : pour la totalité des diagrammes UML
— Eclipse : pour la totalité des classes écrites en langage Java
— Swing : pour le contenu graphique
— WampServeur : pour la base de données de tests (inclus PhpAdmin et MySQL)
— NotePad : pour la création du site
— Google Chrome et Mozilla Firefox : pour les tests du site
— Git : pour le partage du contenu du projet
— Latex : pour la création du rapport
— Photoshop : pour la création d'un logo

Lien Git : `http://www.latex-project.org/`

2.3 Problématique

Le but était d'améliorer le système de vente de l'entreprise qui présentait uniquement ses produits sur les marchés et une page Facebook. Nous avons voulu faire un système simple d'utilisation car l'autoentrepreneur n'est pas adepte des technologies.
L'application peut être améliorée en fonction des besoins.
Nous avons décidé d'appliquer le modèle en spirale pour la gestion du projet, permettant de tester étape par étape, et potentiellement revenir en arrière en cas de besoin de modification d'une étape.

3 Contexte du projet

3.1 Présentation de l'entreprise Autour de la Moto

FIGURE 1 – Logo que nous avons créé pour l'entreprise

3.2 Mise en place du site de vente en ligne

L'entreprise ne disposait au départ d'aucun site internet. Les ventes s'effectuaient sur les marchés ou par le biais d'une page Facebook.
Nous avons donc décidé de créer un site de vente en ligne qui serait propre à l'entreprise et permettrait potentiellement d'élargir la clientèle.
Avant toute chose, nous avons commencé par un travail de recherche sur le langage PHP que nous n'avions jamais étudié. Ensuite nous avons travaillé sur l'agencement du site pour le rendre le plus simple d'utilisation et le plus efficace possible, en essayant d'intégrer un design agréable à la consultation.
L'application de gestion permet à l'entreprise de gérer les commandes à traiter, les stocks en ajoutant ou supprimant des produits et consulter la liste des fournisseurs et les produits qui leur sont associés.

3.3 Présentation de l'application de gestion de commandes

L'application de gestion permet à l'entreprise de gérer les commandes à traiter, les stocks en ajoutant ou supprimant des produits et consulter la liste des fournisseurs et les produits qui leur sont associés.
Tout comme le site, nous voulions une application qui soit simple d'utilisation y compris pour une personne n'ayant pas l'habitude de l'informatique.
Le site et l'application de gestion sont liés par une base de données, ainsi chaque achat de produit sur le site est directement déduit des stocks visibles dans l'application de gestion.

4 Travaux réalisés

4.1 Réalisation d'un site de vente en ligne

Nous avons commencé par créer un logo pour l'entreprise.
Le site de vente a été réalisé en PHP, HTML et CSS. Les pages sont sous forme de PHP nous permettant ainsi de communiquer avec la base de données.
Afin d'effectuer nos tests de base de données nous avons utilisé WampServeur.

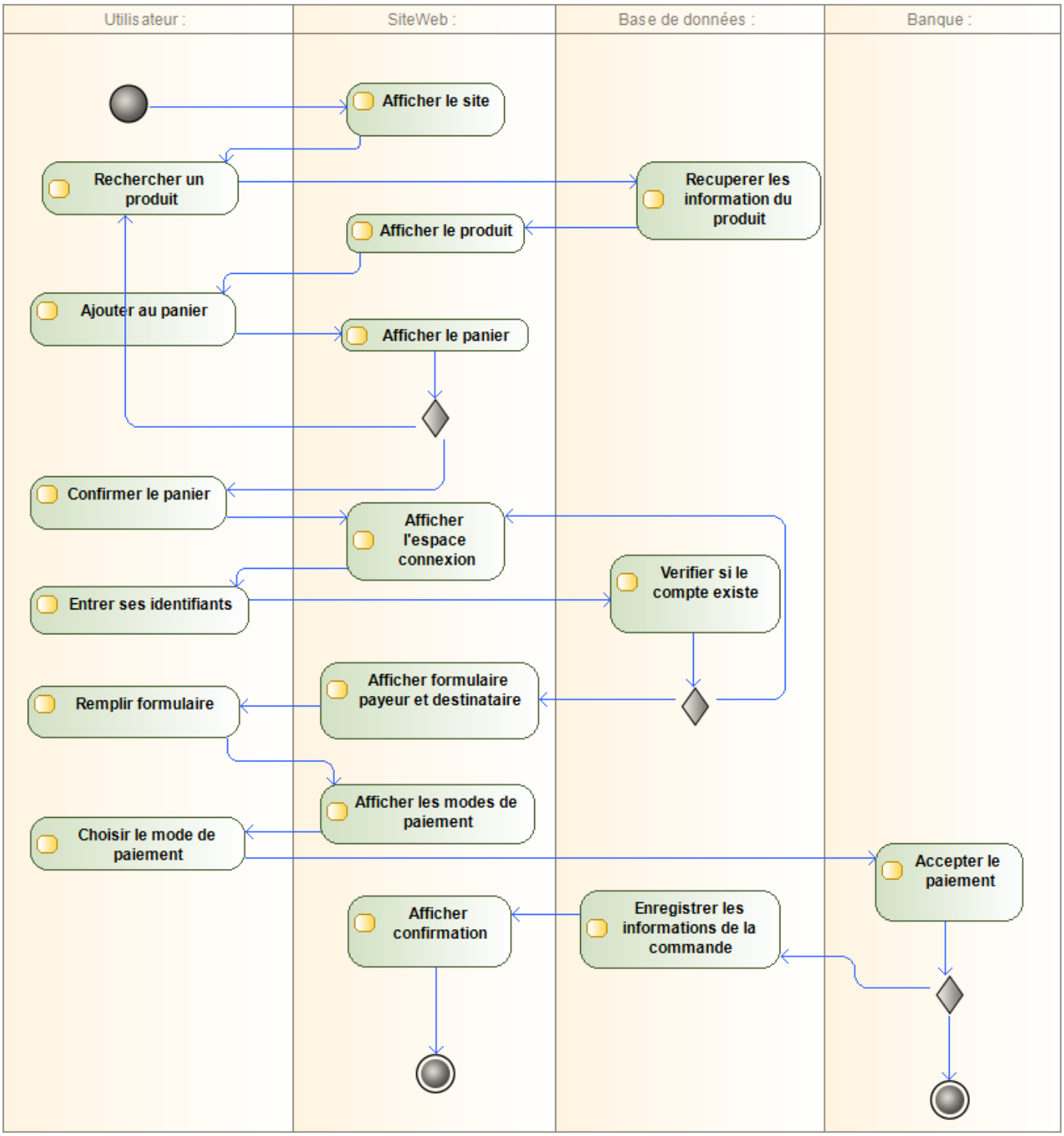

FIGURE 2 – Diagramme d'activité lors du passage d'une commande de la part d'un client sur le site

Le diagramme de séquences illustre les actions de l'utilisateur lorsqu'il consulte le site de vente de la recherche jusqu'à l'acceptation du paiement.

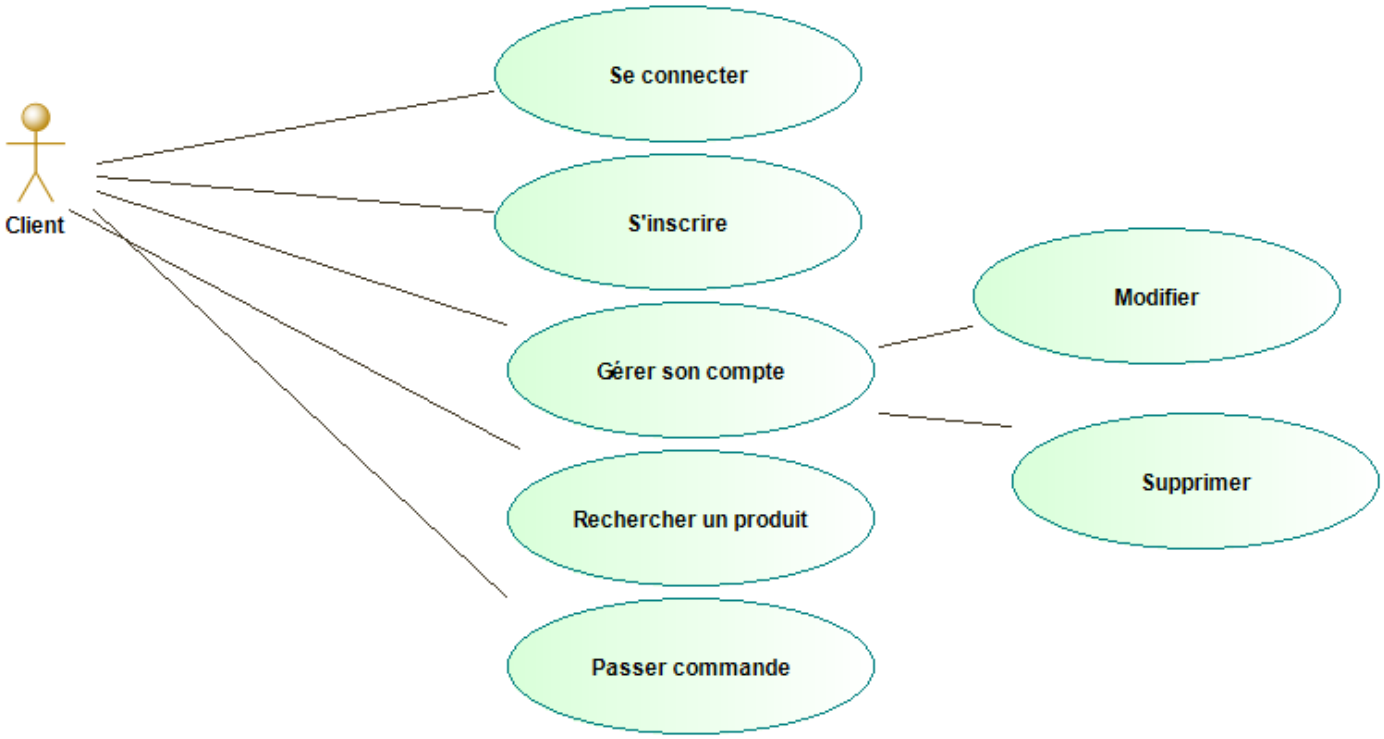

FIGURE 3 – Cas d'utilisation du site

Ce diagramme illustre toutes les actions possibles sur le site par un client.

4.2 Réalisation d'une application de gestion de commandes

L'application de gestion de commandes est composée de quatre classes principales. Trois classes permettant d'effectuer des requêtes SQL utilisent les objets des classes principales.
Cette application permet de consulter les commandes à traiter au fur à mesure que les commandes sont passées sur le site. Elle permet également de voir les commandes déjà traitées. Les listes des fournisseurs et des clients sont également disponibles.
Enfin, il est possible de voir la liste des stocks ainsi que les modifier.

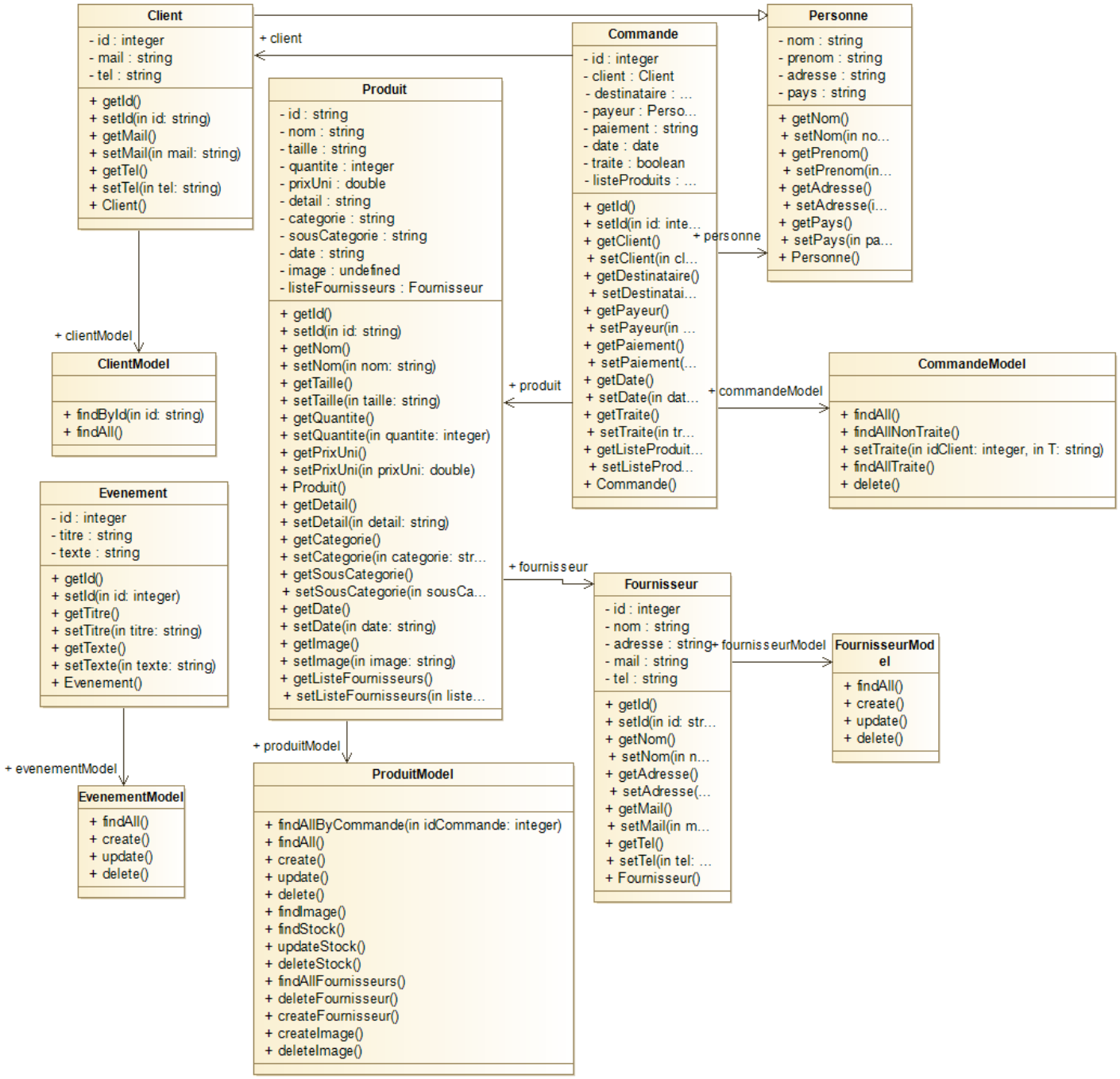

FIGURE 4 – Diagramme de classe de l'application Java

Ce diagramme illustre les actions possibles sur l'application pour l'entrepreneur.

Intitulé	Type	Visibilité	Explication
nom	String	private	Nom de la personne
prenom	String	private	Prenom de la personne
adresse	String	private	Adresse de résidence de la personne
pays	String	private	Pays de résidence de la personne

FIGURE 5 – Attributs de la classe Personne

Intitulé	Type	Visibilité	Explication
id	Integer	private	N° d'identifiant du client
mail	String	private	Mail de contact du client
tel	String	private	N° de téléphone du client

FIGURE 6 – Attributs de la classe Client

Intitulé	Type	Visibilité	Explication
id	Integer	private	N° d'identifiant du produit
nom	String	private	Nom qui désigne le produit
taille	String	private	Taille du produit
quantite	Integer	private	Quantité du produit
prixUni	Double	private	Prix à l'unité du produit
details	String	private	Description du produit
categorie	String	private	Categorie du produit
sousCategorie	String	private	Sous categorie du produit
date	Date	private	Date à laquelle le produit a été ajouté
image	ImageIcon	private	Image représentant le produit
listeFournisseurs	List<Fournisseur>	private	Liste des fournisseurs où on peut se réapprovisionner

FIGURE 7 – Attributs de la classe Produit

Intitulé	Type	Visibilité	Explication
id	Integer	private	N° d'identifiant de la commande
client	Client	private	Client qui a passé la commande
destinataire	Personne	private	Personne a qui est destiné la commande si ce n'est pas le client
payeur	Personne	private	Personne qui paye la commande si ce n'est pas le client
paiement	String	private	Mode de paiement de la commande
date	Date	private	Date à laquelle la commande a été passé
traite	Boolean	private	Traitement de la commande par l'entreprise
listeProduits	List<Produit>	private	Liste des produits composant la commande

FIGURE 8 – Attributs de la classe Commande

Intitulé	Type	Visibilité	Explication
id	integer	private	N° d'identifiant du fournisseur
nom	String	private	Nom du fournisseur
adresse	String	private	Adresse du fournisseur
mail	String	private	Mail de contact du fournisseur
tel	String	private	N° de téléphone du fournisseur

FIGURE 9 – Attributs de la classe Fournisseur

Intitulé	Type	Visibilité	Explication
id	integer	private	N° d'identifiant de l'événement
titre	String	private	Titre de l'événement
texte	String	private	Descriptif de l'événement (lieu, date)

FIGURE 10 – Attributs de la classe Evenement

Intitulé	Type retourné	Visibilité	Arguments	Explications
findById	Client	public	id : Integer	Donner le client correspondant à cet id
findAll	List<Client>	public	-	Donner la liste des clients

FIGURE 11 – Méthodes de la classe ClientModel

Intitulé	Type retourné	Visibilité	Arguments	Explications
findAll	List<Evenement>	public	-	Donner la liste des événements
create	boolean	public	e : Evenement	Créer un nouvel événement
update	boolean	public	e : Evenement	Mettre à jour les événements
delete	boolean	public	e : Evenement	Supprimer un événement

FIGURE 12 – Méthodes de la classe EvenementModel

Intitulé	Type retourné	Visibilité	Arguments	Explications
findAll	List<Fournisseur>	public	-	Donner la liste de tous les fournisseurs
create	Boolean	public	f : Fournisseur	Créer un fournisseur
update	Boolean	public	f : Fournisseur	Mettre à jour un fournisseur
delete	Boolean	public	f : Fournisseur	Supprimer un fournisseur

FIGURE 13 – Méthodes de la classe FournisseurModel

Intitulé	Type retourné	Visibilité	Arguments	Explications
findAll	List<Commande>	public	-	Donner la liste de toutes les commandes
findAllNonTraite	List<Commande>	public	-	Donner la liste de toutes les commandes à traiter
setTraite	boolean	public	idCommande : integer t : boolean	Spécifier dans la base de données si la commande est traitée ou non
findAllTraite	List<Commande>	public	-	Donner la liste de toutes les commandes traitées
delete	boolean	public	c : Commande	Supprimer une commande

FIGURE 14 – Méthodes de la classe CommandeModel

Intitulé	Type retourné	Visibilité	Arguments	Explications
findAllByCommande	List<Produit>	public	idCommande : Integer	Donner la liste des produits d'une commande
findAll	List<Produit>	public	-	Donner la liste de tous les produits
create	Boolean	public	p : Produit	Créer un produit
update	Boolean	public	p : Produit	Mettre à jour un produit
delete	Boolean	public	p : Produit	Supprimer un produit
findImage	ImageIcon	public	id : Integer	Donner l'image d'un produit
findStock	List<Produit>	public	id : Integer	Donner la quantité en stock d'un produit
updateStock	Boolean	public	p : Produit	Mettre à jour les stocks
deleteStock	Boolean	public	p : Produit	Supprimer le stock d'un produit
findAllFournisseurs	List<Fournisseur>	public	id : Integer	Donner la liste des fournisseurs
deleteFournisseurs	Boolean	public	p : Produit f : Fournisseur	Supprimer un fournisseur associé à un produit
createFournisseurs	Boolean	public	p : Produit f : Fournisseur	Associer un nouveau fournisseur à un produit
createImage	Boolean	public	path : String	Ajouter l'image d'un produit
deleteImage	Boolean	public	id : Integer	Supprimer l'image d'un produit

FIGURE 15 – Méthodes de la classe ProduitModel

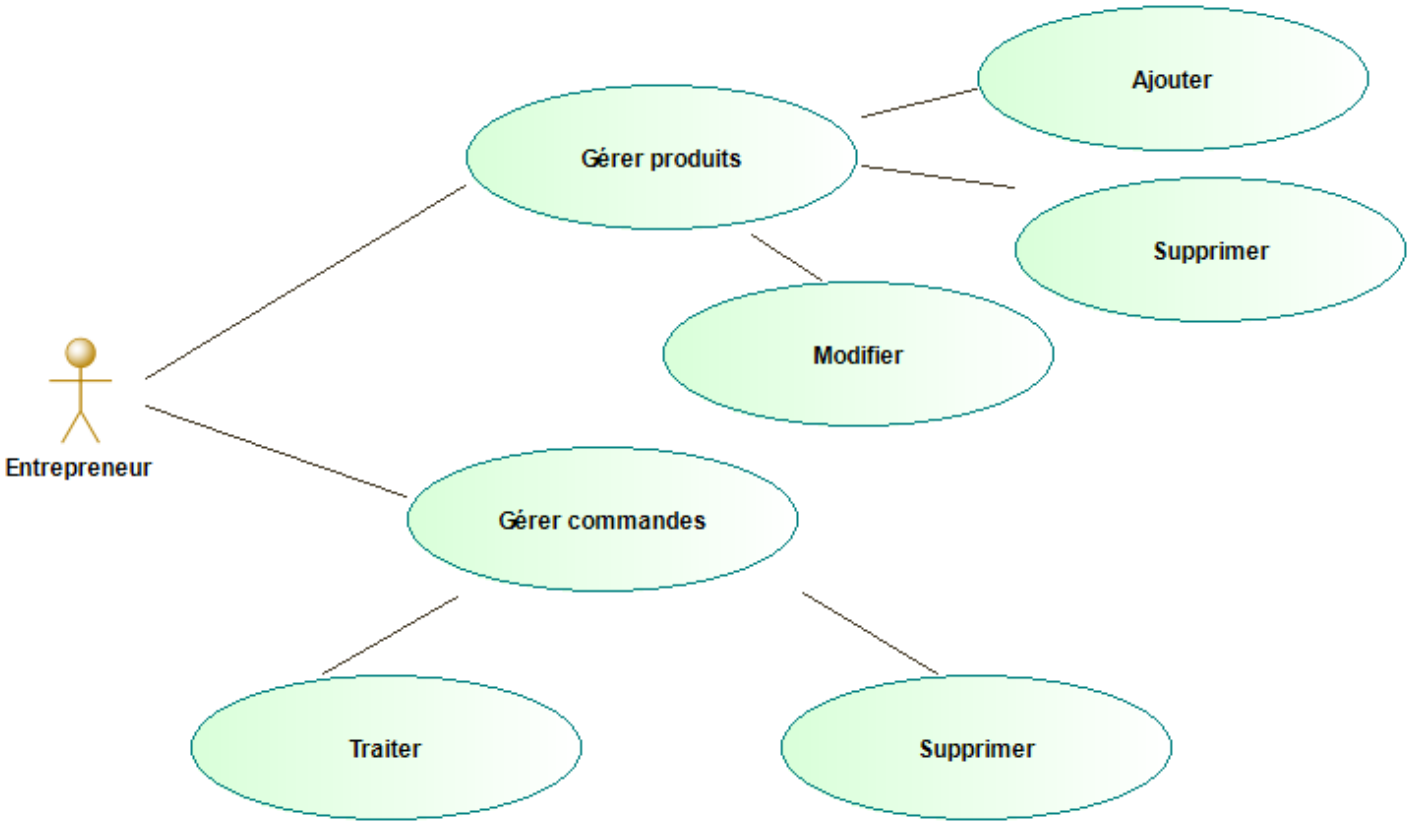

FIGURE 16 – Cas d'utilisation de l'application Java

5 Bilan

L'entreprise Autour de la Moto dispose donc actuellement d'un site nécessitant quelques corrections mais étant en état de marche. Il faut juste trouver un hébergeur. Il est également nécessaire d'ajouter un module de paiement.
L'application de gestion nécessite également quelques corrections mais il est tout de même possible de s'en servir correctement.
Nous avons atteint notre objectif de mettre à disposition des outils simples d'utilisation, et il sera possible à l'avenir de modifier des éléments ou en ajouter.

Bibliographie