

Ref. /13

## Centre Universitaire de Mila

Institut des sciences et de la technologie

Département de Mathématiques et Informatique

# Conception et réalisation d'une application de gestion des produits chimiques au sein des laboratoires du Centre Universitaire De Mila.

Mémoire préparé en vue de l'obtention du diplôme de Licence en  
Informatique

Préparé par :

1- MECHITOUA AMAL

2-BENASKEUR MANAL

Encadré par :

-Mme MAOUCHE FADHILA

Filière : Informatique

Année universitaire : 2012/2013

# Remerciements



*Nous désirons remercier des personnes importantes qui nous ont fortement aidés à concrétiser ce projet, que ce soit de près ou de loin.*

*Un grand merci pour :*

*les ingénieurs du Service informatique de l'institut des sciences et de la technologie surtout Boumaali Fodil.*

*Tous nos enseignants pour toutes les connaissances qu'ils nous ont inculquées.*

*Le gérant du laboratoire Makhlouf pour son aide très bénéfique.*

*A Mme Maouche Fadhila notre encadreur pour sa Disponibilité, son aide précieuse et ses conseils qui nous ont été d'une Utilité indéniable*

*Un remerciement particulier à nos très chers parents, frères, sœurs, collègues et amies respectives qui nous ont encouragé, soutenu durant tout notre parcours.*



*Merci à tous*



*MECHITOUA AMAL*



*BENASKEUR MANAL*



# Dédicace



*Je dédie ce modeste travail à :*

*Mes très chers parents, pour leur amour, leur patience,  
leur soutien et leur confiance.*

*Ma mère Houria qui m'est la plus chère au monde et dans  
ma vie, merci maman, et à toute sa famille Sekfali.*

*Mon père Mouhamed qui est la plus proche personne pour  
moi, merci papa pour votre sacrifice et votre encouragement.*

*A ma grande familles Mechitoua surtout : mon cher frère  
Elhadji, et le meilleur frère au monde Ramzi, mes oncles et mes  
tantes.*

*A Mon binôme Manal et toute sa famille.*

*A toutes mes amies surtout : (Basma, Siham saïda, aziza  
,Dahbia,*

*Sara....).*

*A tous ceux qui me connaissent.*

*Je dédie ce mémoire ...*



*MECHITOUA AMAL*





# Dédicace



☀️ *chaque fois qu'on achève une étape importante dans notre vie, on fait une pose pour regarder en arrière et se rappeler toutes ces personnes qui ont partagé avec nous tous les bons moments de notre existence, mais surtout les mauvais.* ■

*Je dédie ce modeste travail en signe de reconnaissance et de respect.*

■ *A mon cher père **Nouar** qui m'a donné le courage tout le temps.*

*A ma très chère mère **Malika** la plus chère personne pour moi la meilleure femme du monde, elle est comme une bougie qui brûle et brûle pour ses enfants et sa famille, merci ma maman.*

■ *A mon cher frère **Saïd** et mes cousins **Adel** et **Hilal**.* ☀️

■ *A toute ma famille Benaskeur surtout ma grand-mère **Chamama** et mes tantes **Fouzïa**, **Anaba**, **Yasmïna**, **Zeïneb** et **Barïza**.*

■ *A mes amies: **Rabïa**, **Bouchra**, **Samah**, **Karïma**, **Rahma**, **Fatïma**, **Sïham**, **Nasrïn**, **Sara**, **Souad** et **Sabah**.*

■ *Et aussi à **Hourïya**, **Fares**, **Djalal**, **Amar** et **Hïcham**.*

■ *A Mon binôme **Amal**.*

■ *A tous mes enseignant sans exception.*



**BENASKEUR MANAL**



# *Sommaire*

## *Introduction général*

### *Chapitre I: UML & UP*

Introduction.....	1
I UML.....	1
I.2 UML est une unification des méthodes.....	1
I.3 UML n'est pas une méthode ou un processus !.....	1
I.4 Les inconvénients d'UML.....	2
II Les différents diagrammes d'UML.....	2
II.1 Diagramme de cas d'utilisation.....	3
II.1.1 Les éléments des diagrammes de cas d'utilisation.....	3
II.1.2 Les relations entre les acteurs.....	4
II.1.3 Les relations entre cas d'utilisation.....	4
II.2 Diagramme de classe.....	5
II.2.1 Les Relations entre les classes.....	6
II.3 Les Diagrammes de séquences.....	8
II.3.1 Les éléments du diagramme de séquences.....	8
II.4 Les Diagrammes d'activités.....	10
II.5 Mise en œuvre d'UML.....	10
III Processus de développement UP.....	11
III.1 Définition du processus unifié (Unified Process).....	12
III.2 Les phases et les disciplines d'UP.....	14
III.2.1 Les phases.....	14
III.2.2 Les disciplines.....	15
III.3 Pourquoi utiliser up ?.....	15
Conclusion.....	16

# *Chapitre II: Présentation de l'organisme*

## *d'accueil*

Introduction .....	17
I. Présentation de l'organisme d'accueil (Centre Universitaire De Mila).....	17
II. Représentation du laboratoire de produits chimiques.....	19
III. Définition d'un produit chimique .....	19
III.1 Type de produits chimiques.....	19
III.1.1 Solide.....	19
III.1.2 Liquide.....	19
III.2 Caractéristique d'un produit chimique.....	20
III.3 Choix des produits chimiques.....	21
III.4 Choix de Quantité commandée.....	21
III.5 Installations de stockage des produits chimiques.....	22
IV. L'étude des postes de travail.....	23
V. L'étude des procédures.....	23
V.1 Entrée d'un produit.....	23
V.2 Extraire les produits seuil alerte.....	23
V.3 Extraire les produits périmés.....	23
V.4 Traitement des bons internes (sortie d'un produit).....	24
VI. L'étude des documents.....	24
VII. Critiques et suggestions.....	26
Conclusion.....	26

## **Chapitre III : l'étude préliminaire**

Introduction.....	27
I. Présentation général du projet (système).....	27
II. La définition des grands choix techniques.....	27
III. Spécification des besoins fonctionnels.....	27
III.1 L'ajout d'un produit.....	27
III.2 L'établissement d'un bon de sortie.....	28
III.3 l'extraire les produits seuil alerte.....	28
III.4 l'extraire les produits périmés.....	28
III.5 la suppression d'un produit.....	28
III.6 la modification d'un produit.....	28
III.7 la consultation d'un produit.....	28
IV Spécifications des besoins non fonctionnels.....	28
V Descriptions du contexte du système.....	29
V.1 Identification des acteurs.....	29
V.2 Identification des messages.....	29
V.3 Réalisation du diagramme de contexte.....	29
Conclusion .....	30

## **Chapitre IV: Analyse et conception du système**

Introduction.....	31
I. Diagramme de cas d'utilisation .....	31
II. Description des cas d'utilisation.....	32

II.1 Description des scénarios .....	32
II.1.1 Cas d'utilisation: Authentification.....	32
II.1.2 Cas d'utilisation: Ajouter produit.....	33
II.1.3 Cas d'utilisation: Modifier produit chimique.....	34
II.1.4 Cas d'utilisation: Supprimer produit chimique.....	35
II.1.5 Cas d'utilisation: Extraire produit seuil alerte.....	36
II.1.6 Cas d'utilisation: Extraire produit périmé.....	37
II.1.7 Cas d'utilisation: Etablir bon de sortie.....	38
II.1.8 Cas d'utilisation: Consulter produit.....	39
II.2 Description des scenarios par les diagrammes d'activités.....	40
II.2.1 Cas d'utilisation: Authentification.....	40
II.2.2 Cas d'utilisation: Ajouter produit.....	41
II.2.3 Cas d'utilisation: Modifier produit chimique.....	42
II.2.4 Cas d'utilisation: Supprimer produit chimique.....	43
II.2.5 Cas d'utilisation: Extraire produit seuil alerte.....	44
II.2.6 Cas d'utilisation: Extraire produit périmé.....	45
II.2.7 Cas d'utilisation: Etablir bon de sortie.....	46
II.2.8 Cas d'utilisation: Consulter produit.....	47
II.3 Description des scenarios par les diagrammes de séquences.....	48
II.3.1 Cas d'utilisation: Authentification.....	48
II.3.2 Cas d'utilisation: Ajouter produit.....	49
II.3.3 Cas d'utilisation: Modifier produit chimique.....	50
II.3.4 Cas d'utilisation: Supprimer produit chimique.....	51

II.2.5 Cas d'utilisation: Extraire produit seuil alerte.....	52
II.2.6 Cas d'utilisation: Extraire produit périmé.....	53
II.2.7 Cas d'utilisation: Etablir bon de sortie.....	54
II.2.8 Cas d'utilisation: Consulter produit.....	55
III. Le modèle de domaine.....	56
III.1 Représentation d'une classe UML.....	56
III.2 Identification des classes métier.....	57
III.3 Diagramme des classes.....	58
Conclusion.....	58

## **Chapitre V : Réalisation du système**

Introduction .....	59
I.les outils de travail.....	59
I.1 Pacestar UML Diagrammer.....	59
I.2 Environnement de développement de l'application.....	60
I.2.1 Présentation de Delphi.....	60
I.2.2 Les avantages de Delphi.....	61
I.3 Implémentation de la base de données sous Delphi.....	62
II. Le passage du diagramme de classe au modèle relationnel.....	63
II.1 Transformation des entités/ classes .....	63
II.2 Transformation des associations.....	63
III. Description de l'implémentation du système.....	64
Conclusion.....	70

# ***INTRODUCTION GENERALE***

Le laboratoire de produits chimiques au sein du centre universitaire de Mila joue un rôle très important pour l'institut des sciences de la nature et de la vie et l'institut des sciences et de la technologie. Il gère un grand nombre de produits. Ses missions principales sont la réception de produits chimiques, le stockage et la livraison aux enseignants pour faire les travaux pratiques.

La gestion manuelle du laboratoire rend le travail très long et ardu, Un système de gestion automatique des produits chimiques donnera beaucoup d'efficacité et fiabilité au gestionnaire du laboratoire, et assurera une gestion optimale des produits en stock.

L'objectif de notre travail consiste à la conception et la réalisation d'un système pour la gestion des produits chimiques au sein du laboratoire de centre universitaire de Mila.

Pour atteindre cet objectif nous avons adopté le processus UP comme démarche de développement pour l'analyse et la conception de notre système, lui-même basé sur l'utilisation du langage de modélisation UML. Pour concrétiser notre application, notre choix s'est porté sur le langage Delphi 7 comme langage de programmation et Database Desktop (un composant de delphi) comme SGBD.

Notre mémoire est organisé en cinq chapitres comme suit :

**Le chapitre I:**UML & UP.Dans ce chapitre on a donné une présentation de l'outil de modélisation UML (Unified Modeling Language), ainsi que le processus de développement UP que nous avons adopté pour aboutir à notre système.

**Le chapitre II:** Présentation de l'organisme d'accueil. Dans ce chapitre on a pris en considération le domaine d'étude qui est le laboratoire de produit chimique de centre Universitaire De Mila qui est chargé de la réception des produit et leurs stockage.

**Le chapitre III:** l'étude préliminaire. Dans ce chapitre, on applique la première étape du processus UP sur notre domaine d'étude.

**Le chapitre IV:** Analyse et conception du système.Ce chapitre constitue l'essentiel de notre travail, il décrit de façon détaillée les phases d'analyse et de conception de notre application.

**Le chapitre V** Réalisation du système. Dans ce chapitre nous aborderons le passage de l'orienté objet vers le relationnel, les outils de développement que nous avons utilisé tel que : pcestar UML, Delphi 7, enfin on expose quelques interfaces de l'application.

Enfin, nous terminerons notre mémoire par une conclusion générale.

# *Chapitre I*

## *UML & UP*

**I. La notation UML**

**II. Les différents types de diagrammes d'UML**

**III. Processus de développement UP**

## Introduction

Dans ce chapitre, nous allons présenter quelques généralités du langage de modélisation UML (Unified Modeling language), et le processus de développement UP utilisés pour la réalisation de notre projet.

### I. UML

#### I.1 Définition

UML (Unified Modeling Language) se définit comme un langage de modélisation graphique qui permet la spécification, la représentation et la construction des composantes d'un système informatique. La notation UML est adoptée par L'OMG en 1997 comme une norme (standard).

#### I.2 UML est une unification des méthodes

UML est né de la fusion des trios méthode qui ont le plus influencées la modélisation objet au milieu des années 90.

-OMT : de James Rumbaugh fournit une représentation graphique des aspects statique dynamique et fonctionnel d'un système.

-OOD : de Grady Booch, définie pour le Département of Défense, introduit le concept de paquetage (package).

-OOSE : d'Ivar Jacobson Fonde l'analyse sur la description de besoins des utilisateurs (cas d'utilisation, ou use cases).

#### I.3 UML n'est pas une méthode ou un processus !

UML n'est pas une méthode ou un processus : UML est développé pour permettre la modélisation des systèmes d'une manière standard et pas pour être une méthode de conception ou d'analyse, pour ça il lui manque la démarche.

Nous utilisons UML qui représente un langage formel, normalisé; et vérifier avec les points forts suivants :

- Un outil majeur de communication entre les différents intervenants au sein d'un projet.
- Faciliter la compréhension du système étudié et les couts et les détails.
- Comprendre et décrire les besoins.
- Spécifier et documenter le système.
- Esquisser des architectures logicielles.
- Le modèle est prévu pour arriver à anticiper les résultats du codage.

## I.4 Les inconvénients d'UML

UML contient des points faible comme :

- la mise en œuvre d'UML nécessite un réel apprentissage car sa sémantique est très précise, ce qui amoindrit quelque peu son caractère d'universalité.
- UML n'est pas une méthode il a besoin d'existence d'un processus pour la réussite de ses projets.

## II. Les différents types de diagrammes d'UML

UML s'articule maintenant autour de 13 diagrammes différents, chacun d'eux est dédié à la représentation d'un système logiciel suivant un point de vue particulier :

### -Diagramme Structurel ou diagramme Statique (UML Structures).

- Les Diagrammes de classe (Class diagram).
- Les diagrammes d'objet (Object diagram).
- Les diagrammes de composants (Composant diagram).
- Les diagrammes de déploiement (Deployment diagram).
- Les diagrammes de paquetage (Package diagram).
- Les diagrammes de structure composite (Composite structure diagram).

### -Diagramme comportementaux ou diagramme dynamique

- Les diagrammes cas d'utilisation (Use case diagram).
- Diagramme d'activités (Activity diagram).
- Diagramme d'états-transition (Stat machine diagram).
- **Diagramme d'interaction (Interaction diagram) :**
  - Diagramme de séquence (Séquence diagram).

- Diagramme de communication (communication diagram).
- Diagramme global d'interaction (Interaction overview diagram).
- Diagramme de temps (Timing diagram).

Parmi ses diagrammes, les plus utilisés sont : Les diagrammes d'activités, de cas d'utilisation, de classes, de séquences.

## II.1 Diagramme de cas d'utilisation

Le diagramme des cas d'utilisation représente la structure des grandes fonctionnalités nécessaires aux utilisateurs du système. C'est le premier diagramme du modèle UML, celui qui assure la relation entre l'utilisateur et les objets que le système met en œuvre.

### II.1.1 Les éléments des diagrammes de cas d'utilisation

- **Acteur** : Représente le rôle joué par une entité externe (utilisateur humain, dispositif matériel ou autre système ....) qui interagit directement avec le système étudié. Il se représente par un petit bonhomme, il existe deux types d'acteur: primaire & secondaire.

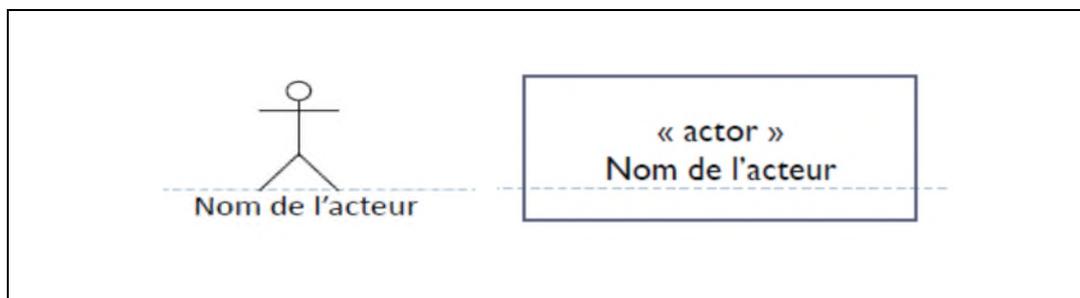


Figure I.1 : Représentation d'acteur.

- **Cas d'utilisation** : Une unité cohérente représentant une fonctionnalité visible de l'extérieur (acteur), il réalise un service bout en bout, avec un déclenchement, un déroulement et une fin, pour l'acteur. Un cas d'utilisation se représente par une ellipse contenant le nom du cas (un verbe à l'infinitif), et optionnellement, au-dessus du nom, un stéréotype.

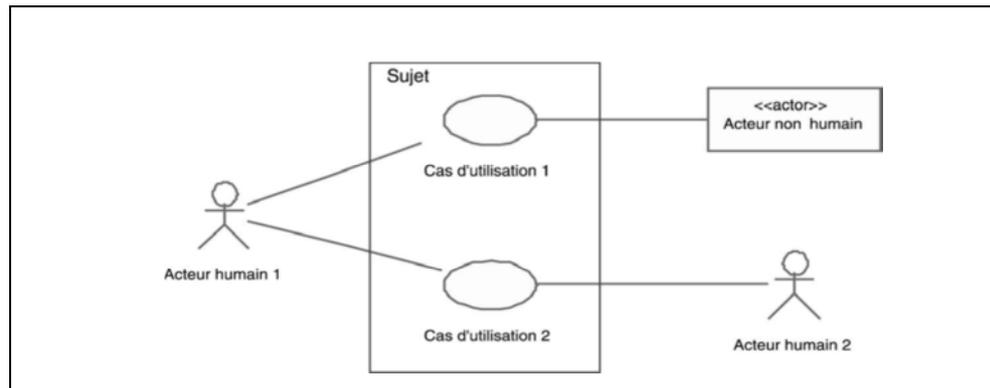


Figure I.2 : Représentation d'un cas d'utilisation.

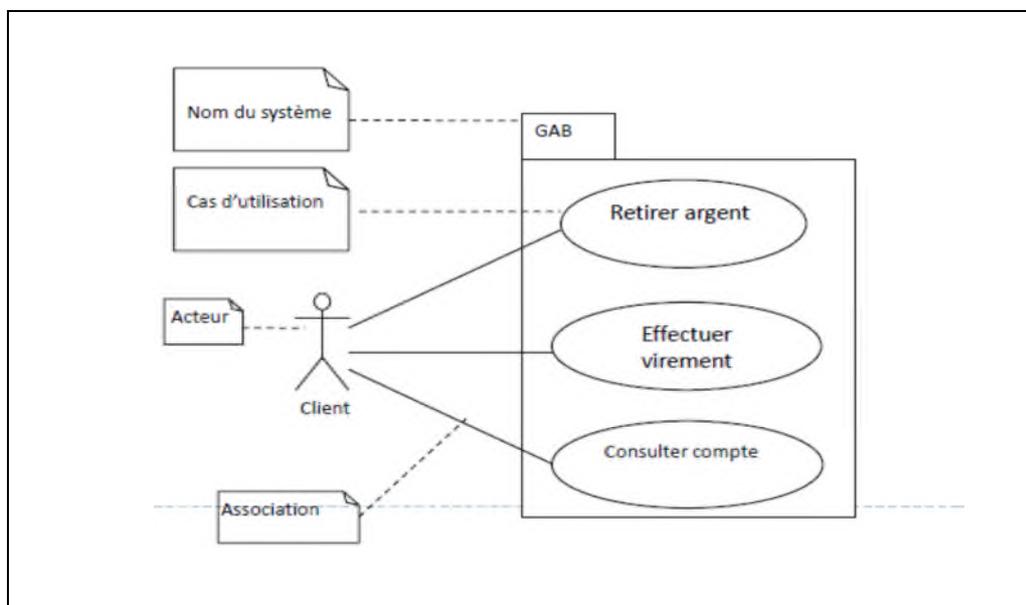


Figure I.3 : Diagramme de cas d'utilisation

## II.1.2 Les relations entre les acteurs

La seule relation entre les acteurs est la relation de **généralisation**. Quand un acteur fils hérite d'un acteur père, il hérite en réalité de toutes les associations du père.

## II.1.3 Les relations entre cas d'utilisation

- **Relation d'inclusion:** Un cas A inclut un cas B si le comportement décrit par le cas A inclut le comportement du cas B (B est une partie de A). La relation d'inclusion est symbolisée par une flèche discontinue stéréotype <<includ>>.

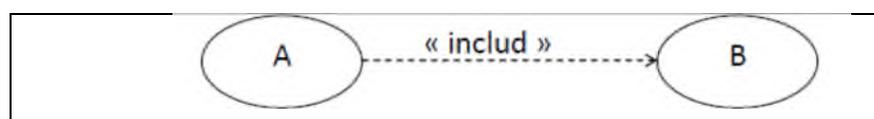


Figure I.4 : Relation d'inclusion.

- **Relation d'extension :** Un cas d'utilisation B étend un cas d'utilisation A si B éventuellement une partie de A. La relation d'extension est symbolisée par une flèche discontinue stéréotype «*extend*».

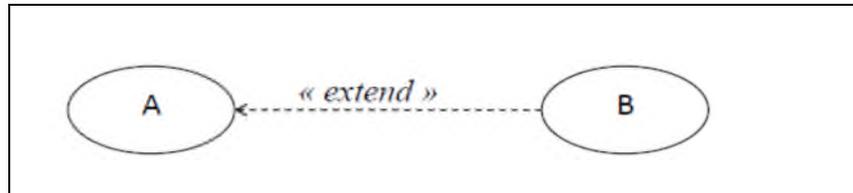


Figure I.5 : Relation d'extension.

- **Relation de généralisation/spécialisation :**

- un cas A est une généralisation d'un cas B si B est un cas particulier de A.
- les cas d'utilisation spécialisés héritent la sémantique de leur parent.
- Cette relation est une flèche dont la pointe (un triangle fermé) est dirigée vers le cas d'utilisation le plus Général.

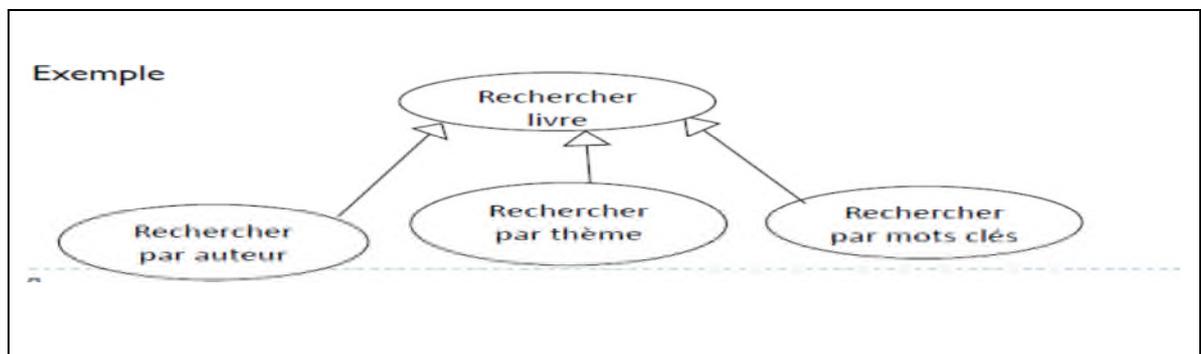


Figure I.6 : Relation de généralisation.

## II.2 Diagramme de classe

Le diagramme de classes est considéré comme le plus important et le plus utilisé dans un développement orienté objet, il fournit une représentation abstraite des objets du système qui vont interagir ensemble pour réaliser les cas d'utilisation.

- **La classe :** est une description abstraite d'un ensemble d'objets possédant les mêmes attributs et méthodes. La classe est représentée par un rectangle divisé en trois compartiments.

- **Un objet** : est une entité aux frontières bien définies, possédant une identité encapsulant un état et un comportement. Un objet est une instance (ou occurrence) d'une classe.
- **Un attribut** : est l'identifiant de la classe, un attribut joue un rôle particulier, car il permet de gérer de façon unique chaque objet, instance de la classe.
- **Les Méthodes** : Représente les éléments de comportement (un service) contenu dans une classe.

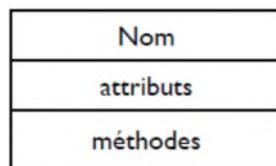


Figure I.7 : Représentation de classe.

## II.2.1 Les Relations entre les classes

➤ **Association** : est une relation entre deux classes qui indique qu'il peut y avoir des liens entre des instances des classes associations.

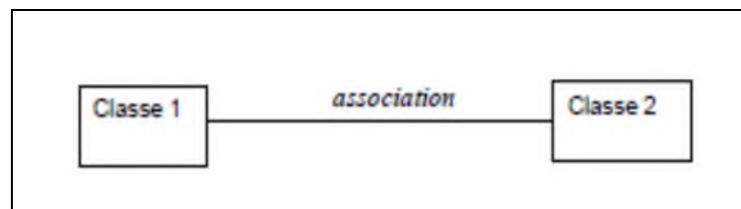


Figure I.8 : Relation d'association.

➤ **Multiplicité (cardinalité)**: est une information caractérisant une association qui spécifie le minimum et le maximum d'objets qui peuvent participer à une relation dans le cadre d'une association :

- Un et un seul : 1 ou 1...1.
- Zéro ou un : 0...1.
- Plusieurs : \* ou 0...\*.
- De M à N (entiers naturels) : M...N.

➤ **Les classes-association** : Possède les caractéristiques des associations et des classes : elle se connecte à deux ou plusieurs Classes et possède également des attributs et des opérations, caractérisées par un trait discontinue entre la classe et l'association qu'elle représente comme suit :

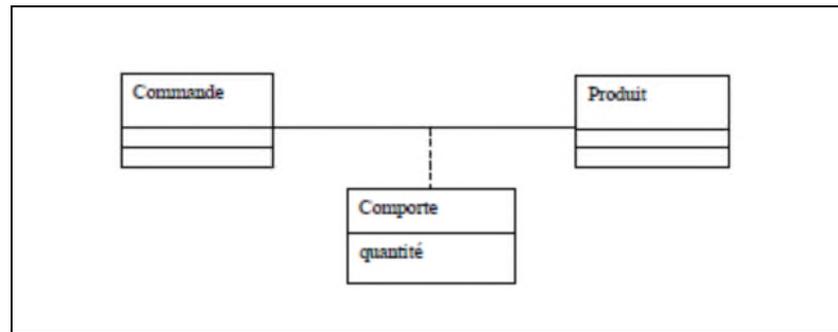


Figure I.9 : Représentation des classes association.

➤ **Agrégation** : Une agrégation est une forme particulière d'association où un tout (appelé classe *agrégat*) est relié à ses parties (appelé classe *agrégées*).

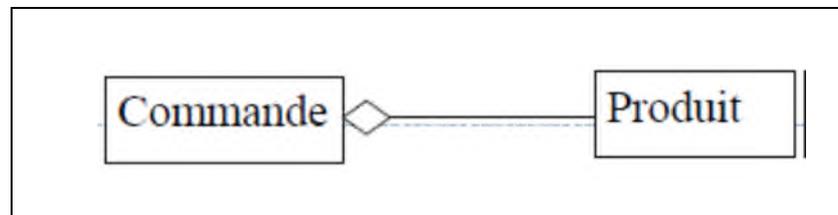


Figure I.10 : Relation d'agrégation.

➤ **Composition** : La composition, également appelée agrégation composite, décrit une contenance structurelle entre instance.

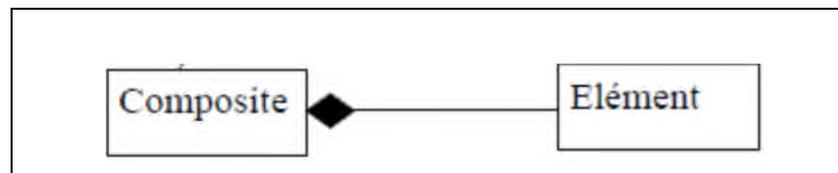


Figure I.11 : Relation de composition.

➤ **L'héritage** : L'héritage décrit une relation entre une classe générale (classe de base parent) et une classe spécialisée (sous-classe) tel que la classe spécialisée (enfant) possède toutes les caractéristiques (parent) et peut comporter des attributs et des méthodes spécifiques supplémentaires, elle est représentée par une flèche avec un trait plein dont la pointe est un triangle fermé désignant le cas le plus général.

➤ **Les interfaces**: Une interface est un ensemble de constantes et de déclaration de méthode, elle est représentée par un classeur stéréotype <<interface>> et un trait discontinu terminé par une flèche triangulaire et le stéréotype <<realize>>. Les classes qui implémentent une interface doivent implémenter toutes ses méthodes.

## II.3 Les Diagrammes de séquences

Les diagrammes de séquence sont la représentation graphique d'un ordre chronologique entre les acteurs et les objets du système.

Les principales informations contenues dans un diagramme de séquence sont les messages échangés entre les lignes de vie, présentés dans un ordre chronologique.

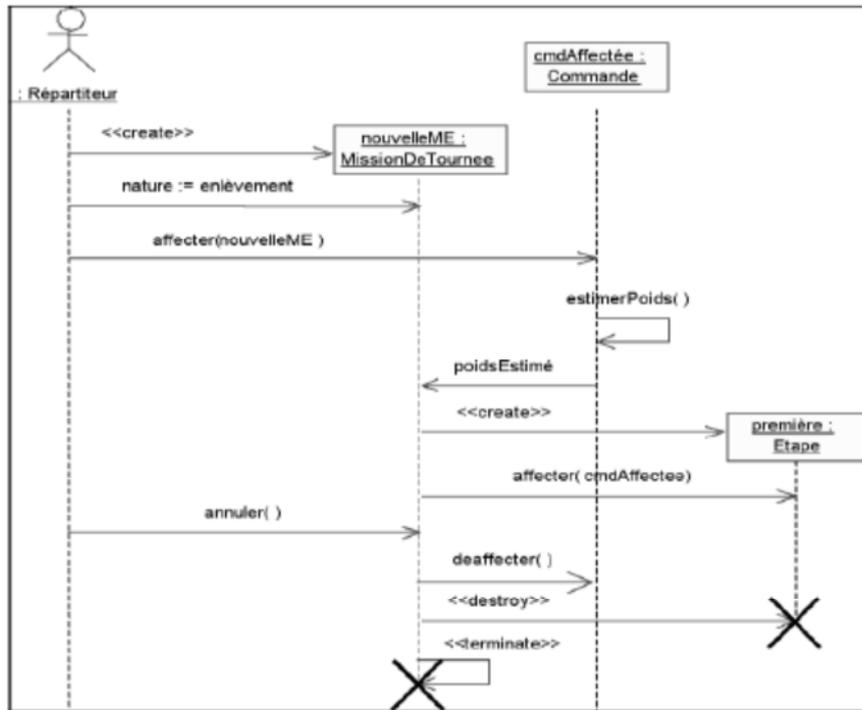


Figure I.12 : Diagramme de séquence.

### II.3.1 Les éléments du diagramme de séquences

- **Les lignes de vie:** Une ligne de vie représente un objet qui participe à l'interaction, une verticale pointillée contenant une étiquette.

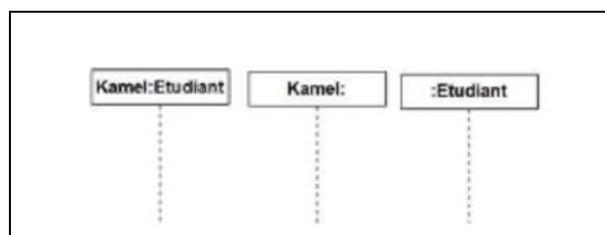


Figure I.13 : Les lignes de vie.

- **les Messages :** Un message représente la spécification d'une communication unidirectionnelle entre les objets qui transportent de l'information avec l'intention de

déclencher une activité chez le récepteur. Plusieurs types de messages existent, les plus communes sont :

- ✓ **Message synchrones** : Dans ce cas l'émetteur est bloqué le temps que le récepteur traite le message envoyé et envoie la réponse, avant de poursuivre ses actions, le message peut être suivi d'une réponse représentée par une flèche en discontinue.

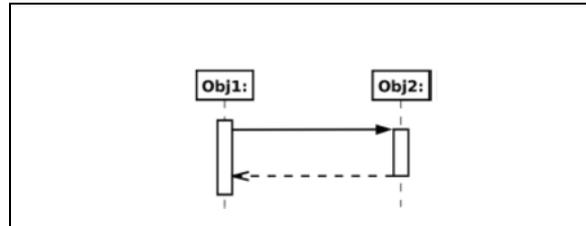


Figure I.14 : Message synchrone.

- ✓ **Message asynchrones** : Dans ce cas l'émetteur n'est pas bloqué le récepteur traite le message envoyé, le message peut être suivi d'une réponse représentée par une flèche en trait plein.

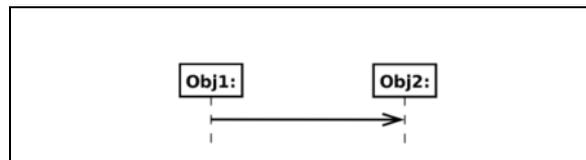


Figure I.15 : Message asynchrones.

- ✓ **Message de création et destruction d'instance** : la création d'un objet est matérialisée par une flèche qui pointe sur le sommet d'une ligne de Vie. La destruction d'un objet est matérialisée par une croix qui marque la fin de la ligne de vie de l'objet, la destruction d'un objet n'est pas nécessairement consécutive à la réception d'un message.

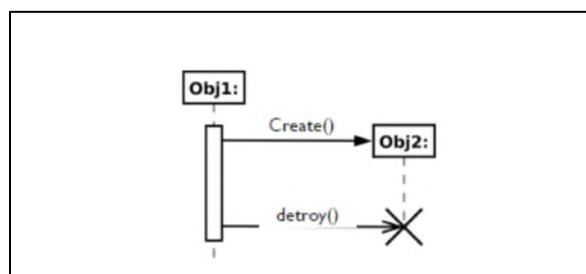


Figure I.16 : Message de création.

## II.4 Les Diagrammes d'activités

Le diagramme d'activités permet de représenter la dynamique du système d'information. Les diagrammes d'activités permettent de spécifier des traitements très proche des langages de programmation objet : spécifier des actions de base, structure de contrôle (condition, boucle), Ils sont donc bien adapter à la spécification détaillée des traitements en phase de réalisation. On peut aussi les utiliser de façon plus informelle pour décrire des enchainements d'actions de haut niveau, en particulier pour la description détaillée des cas d'utilisation.

- **Activité:** Une activité définit un comportement décrit par un séquencèrent organisé d'unités dont les éléments simples sont les actions. Une activité est un comportement et à ce titre peut être associée à des paramètres. De la gauche vers la droite, on trouve : le nœud représentant une action, un nœud objet, un nœud de décision ou de fusion, un nœud de bifurcation ou d'union, un nœud initial, un nœud final et un nœud final de flot.
- **Transition :** Le passage d'une activité vers une autre est matérialisé par une transition. Graphiquement les transitions sont représentées par des flèches en traits pleins qui connectent les activités entre elles. Les transitions spécifient l'enchaînement des traitements et définissent le flot de contrôle.
- **La synchronisation :** Les flots de contrôle parallèles sont séparés par des barres de synchronisation qui peuvent être des :
  - **Débranchements:** les transitions qui partent d'un débranchement ont lieu en même temps.
  - **Jonctions:** on ne franchit une jonction qu'après la réalisation de toutes les transitions qui s'y rattachent.

## II.5 Mise en œuvre d'UML

UML n'est pas une méthode et ne propose pas une démarche de modélisation explicitant et en cadrant toutes les étapes d'un projet, le modèle sert donc des objectifs différents suivant l'étape de développement et sera construit avec des points de vue de plus en plus détaillés.

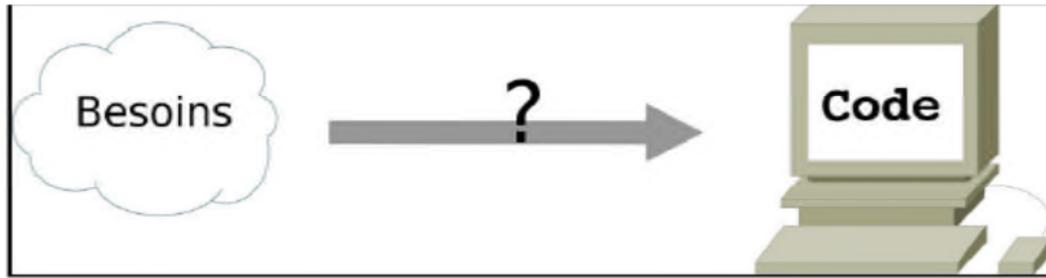


Figure I.17 : Quelle méthode pour passer de l'expression des besoins au code.

**Dans les activités de capture des besoins**, il convient premièrement de considérer le système comme une boîte noire à part entière afin d'étudier sa place dans le système métier plus global qu'est l'entreprise. On développe pour cela un modèle de niveau contexte, afin de tracer précisément les frontières fonctionnelles du système ; ses activités sont réalisées à l'aide de : Diagramme de cas d'utilisation, de séquence système et maquette IHM.

L'objectif de diagramme de cas d'utilisation est l'identification et la présentation des besoins à partir des informations recueillies lors des rencontres entre informaticiens et utilisateurs. Les **Diagrammes de séquence système** utilisés pour une Spécification détaillée des besoins. Les **maquettes d'IHM (Interface Homme-Machine)** permettent aux utilisateurs d'avoir une vue concrète mais non définitive de la future interface de l'application.

**Dans les activités d'analyse**, le modèle représente le système vu de l'intérieur. Il se compose d'objets représentant une abstraction des concepts manipulés par les utilisateurs. Le modèle comprend le modèle du domaine, diagramme de classe participant et diagramme de navigation. Le modèle du domaine décrit les classes qui modélisent les concepts du domaine d'application (on utilise aussi le terme de métier), c.à.d. des objets du monde réel. Le diagramme de classes participantes est particulièrement important puis qu'il effectue la jonction entre, d'une part, les cas d'utilisation, le modèle du domaine. Le but d'un diagramme de classe de conception et de Produire le diagramme de classe qui servira pour l'implémentation.

### III. Processus de développement UP

UML facilite la compréhension et la spécification des composant d'un système d'information donc UML n'est pas une méthode c'est pour ca qu'on utilise une démarche, un processus.

Un **processus** définit une séquence d'étapes, partiellement ordonnées. L'objet d'un processus de développement est de produire des logiciels de qualité qui répondent aux besoins de leurs utilisateurs dans des temps et des coûts prévisibles. Le processus peut se décomposer suivant deux axes de contrôle sur le développement :

- l'axe de développement technique, qui se concentre principalement sur la qualité de la production ;
- l'axe de gestion du développement, qui permet la mesure et la prévision des coûts et des délais.

## III.1 Définition du processus unifié (Unified Process)

Un processus unifié est un processus de développement logiciel construit sur UML ; il est itératif et incrémental, centré sur l'architecture, conduit par les cas d'utilisation et piloté par les risques. La gestion d'un tel processus est organisée d'après les 4 phases suivantes:

Pré étude, élaboration, construction et transition.

Ses activités de développement sont définies par 6 disciplines fondamentales qui décrivent la modélisation métier, la capture des besoins, l'analyse et la conception, l'implémentation, le test et le déploiement.

Le processus unifié doit donc être compris comme une trame commune des meilleures pratiques de développement, et non comme l'ultime tentative d'élaborer un processus universel. La définition d'un processus UP est donc constituée de plusieurs disciplines d'activité de production et de contrôle de cette production. Tout processus UP répond aux caractéristiques ci-après.

- **Il est itératif et incrémental** : le projet est découpé en itérations de courte durée (environ 1 mois) qui aident à mieux suivre l'avancement global. À la fin de chaque itération, une partie exécutable du système final est produite, de façon incrémentale.

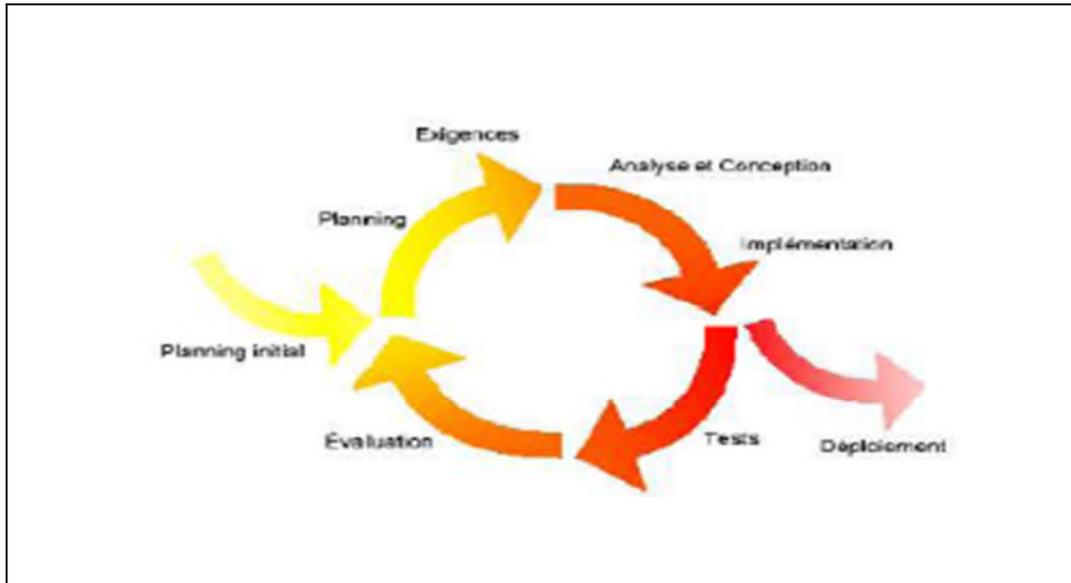


Figure I.18: UP est itératif et incrémental.

- **Il est piloté par les risques :** Dans ce cadre, les causes majeures d'échec d'un projet logiciel doivent être écartées en priorité. Nous identifions une première cause provenant de l'incapacité de l'architecture technique à répondre aux contraintes opérationnelles, et une seconde cause liée à l'inadéquation du développement aux besoins des utilisateurs.
- **Centré sur l'architecture :** tout système complexe doit être décomposé en parties modulaires afin de garantir une maintenance et une évolution facilitées. Cette architecture (fonctionnelle, logique, matérielle, etc.) doit être modélisée en UML et pas seulement documentée en texte.
- **Il est orienté utilisateur :** car la spécification et la conception sont construites à partir des modes d'utilisation attendus par les acteurs du système.

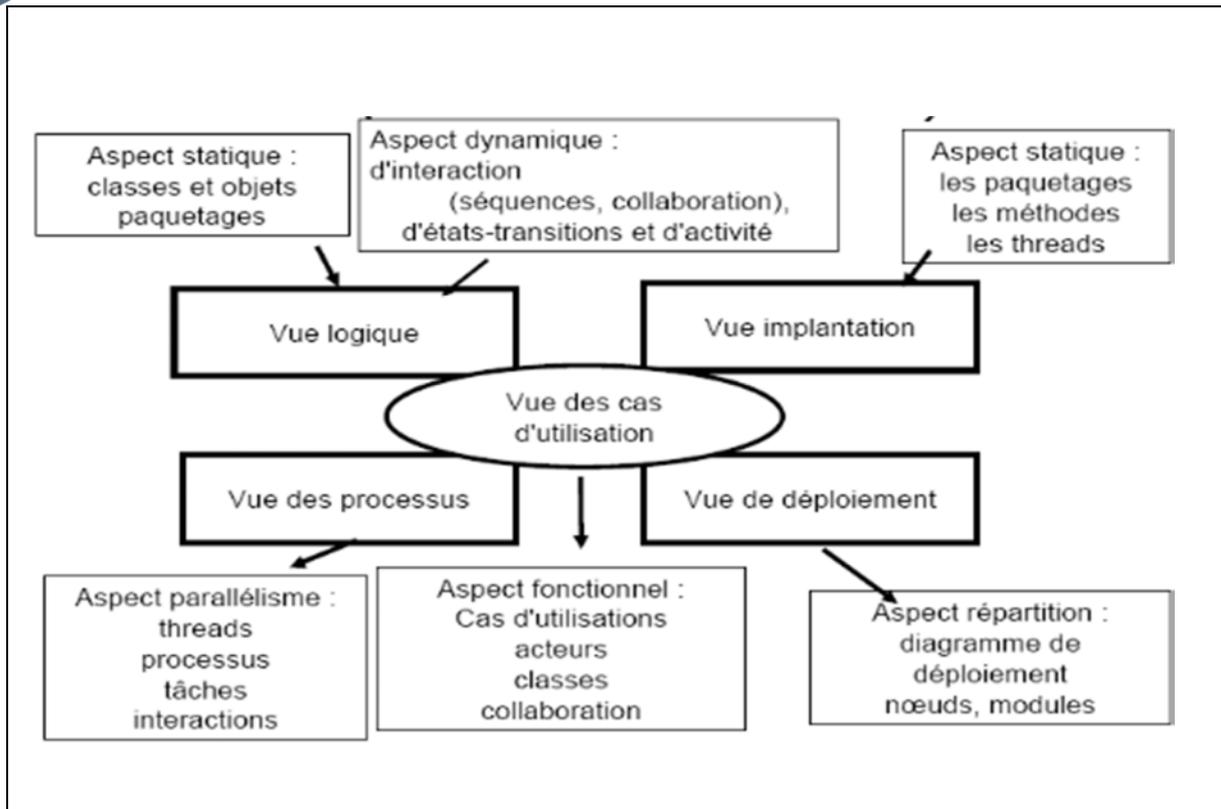


Figure I.19: UP est centré sur l'architecture.

## III.2 Les phases et les disciplines d'UP

### III.2.1 Les phases

La gestion d'un tel processus est organisée suivant les quatre phases suivantes : Initialisation, élaboration, construction et transition.

➤ **La phase d'initialisation:** Conduit à définir la « vision » du projet, sa portée, sa faisabilité, son business case, afin de pouvoir décider au mieux de sa poursuite ou de son arrêt.

➤ **La phase d'élaboration:** Poursuit trois objectifs principaux en parallèle : identifier et décrire la majeure partie des besoins des utilisateurs, construire (et pas seulement décrire dans un document !) l'architecture de base du système, lever les risques majeurs du projet.

➤ **La phase de construction:** consiste surtout à concevoir et implémenter l'ensemble des éléments opérationnels (autres que ceux de l'architecture de base). C'est la phase la plus consommatrice en ressources et en effort.

➤ **La phase de transition:** permet de faire passer le système informatique des mains des développeurs à celles des utilisateurs finaux. Les mots-clés sont : conversion des données, formation des utilisateurs, déploiement, bêta-tests.

Chaque phase est elle-même décomposée séquentiellement en itérations limitées dans le temps (entre 2 et 4 semaines). Le résultat de chacune d'elles est un système testé, intégré et exécutable. L'approche itérative est fondée sur la croissance et l'affinement successifs d'un système par le biais d'itérations multiples, feedback et adaptation cycliques étant les moteurs principaux permettant de converger vers un système satisfaisant.

Le système croît avec le temps de façon incrémentale, itération par itération, et c'est pourquoi cette méthode porte également le nom de développement itératif et incrémental. Il s'agit là du principe le plus important du Processus Unifié.

## III.2.2 Les disciplines

- **L'analyse:** Détaille les cas d'utilisation et procède à une première répartition du comportement du système entre divers objets.
- **La conception:** Définit la structure statique du système sous forme de sous système, classes et interfaces; Définit les cas d'utilisation réalisés sous forme de collaborations entre les sous systèmes les classes et les interfaces.
- **L'implémentation:** Intègre les composants (code source) et la correspondance entre les classes et les composants.
- **Le déploiement:** Définit les nœuds physiques des ordinateurs et l'affectation de ces composants sur ces nœuds.
- **Le test :** Décrit les cas de test vérifiant les cas d'utilisation.

## III.3 Pourquoi utiliser up ?

L'objectif d'un processus unifié est de maîtriser la complexité des projets informatiques en diminuant les risques. Il existe plusieurs processus de développement (la méthode séquentielle merise) et d'autre adaptation d'up comme (RUP, XUP, AUP, EUP, 2TUP, EssUP, scrum) mais, nous utilisant up car UP est un ensemble de principes génériques et simples adapté en fonctions des spécificités des projets. UP répond aux préoccupations suivantes :

- **QUI** participe au projet ?

- **QUOI**, qu'est-ce qui est produit durant le projet ?
- **COMMENT** doit-il être réalisé ?
- **QUAND** est réalisé chaque livrable ?

Cette préoccupation répond aux besoins d'UML pour son projet.

## Conclusion

Dans ce chapitre, nous avons présenté UML & UP simplifier pour pouvoir les utilisés dans le reste de notre travail.

# ***Chapitre II***

## ***Présentation de l'organisme d'accueil***

**I. Présentation de l'organisme d'accueil (Centre  
Universitaire De Mila)**

**II. Représentation du laboratoire de produits  
chimiques**

**III. Définition d'un produit chimique**

**IV. L'étude des postes de travail**

**V. L'étude des procédures**

**VI. L'étude des documents**

**VII. Critiques et suggestions**

# Chapitre II: Présentation de l'organisme d'accueil

## Introduction

Nous avons réalisé notre stage au sein du laboratoire des produits chimiques du centre universitaire de Mila. Le monde de la gestion des laboratoires, est un monde très vaste puisqu'il englobe une multitude de domaines. Les questions qui doivent être traitées sont par exemple: Quels sont les formats de documents importants, quelles sont les fonctionnalités exactes d'un système de gestion de laboratoire, etc. Dans ce chapitre nous allons présenter le laboratoire des produits chimiques, puis nous discuterons l'importance de ce domaine dans la gestion générale de laboratoire, ainsi que les étapes à suivre pour cette gestion, la typologie des systèmes de cette gestion, et enfin, les enjeux de cette gestion.

## I. Présentation de l'organisme d'accueil (Centre Universitaire De Mila)

Le Centre Universitaire de Mila est un établissement public à caractère scientifique, culturel et professionnel doté de la personnalité morale et de l'autonomie financière. Il a été créé par le décret exécutif N°204-08 du 09 juillet 2008. Donnant ainsi naissance à l'enseignement Supérieur dans la wilaya. Il est situé à 5km du centre-ville de Mila sur la route nationale N°79 allant vers les communes de Zeghaia et de Ferdjioua et s'étend sur une superficie d'environ 87 hectares. Le Centre Universitaire de Mila est composé de trois instituts et propose une formation supérieure dans plusieurs domaines :

- Institut des sciences et de la technologie :
  - ✓ Département math et informatique.
  - ✓ Département science de la nature et de la vie.
  - ✓ Département science et technologie.
- Institut des sciences économiques, commerciales et des sciences de gestion
- Institut des langues et littérature Arabe.

# Chapitre II: Présentation de l'organisme d'accueil

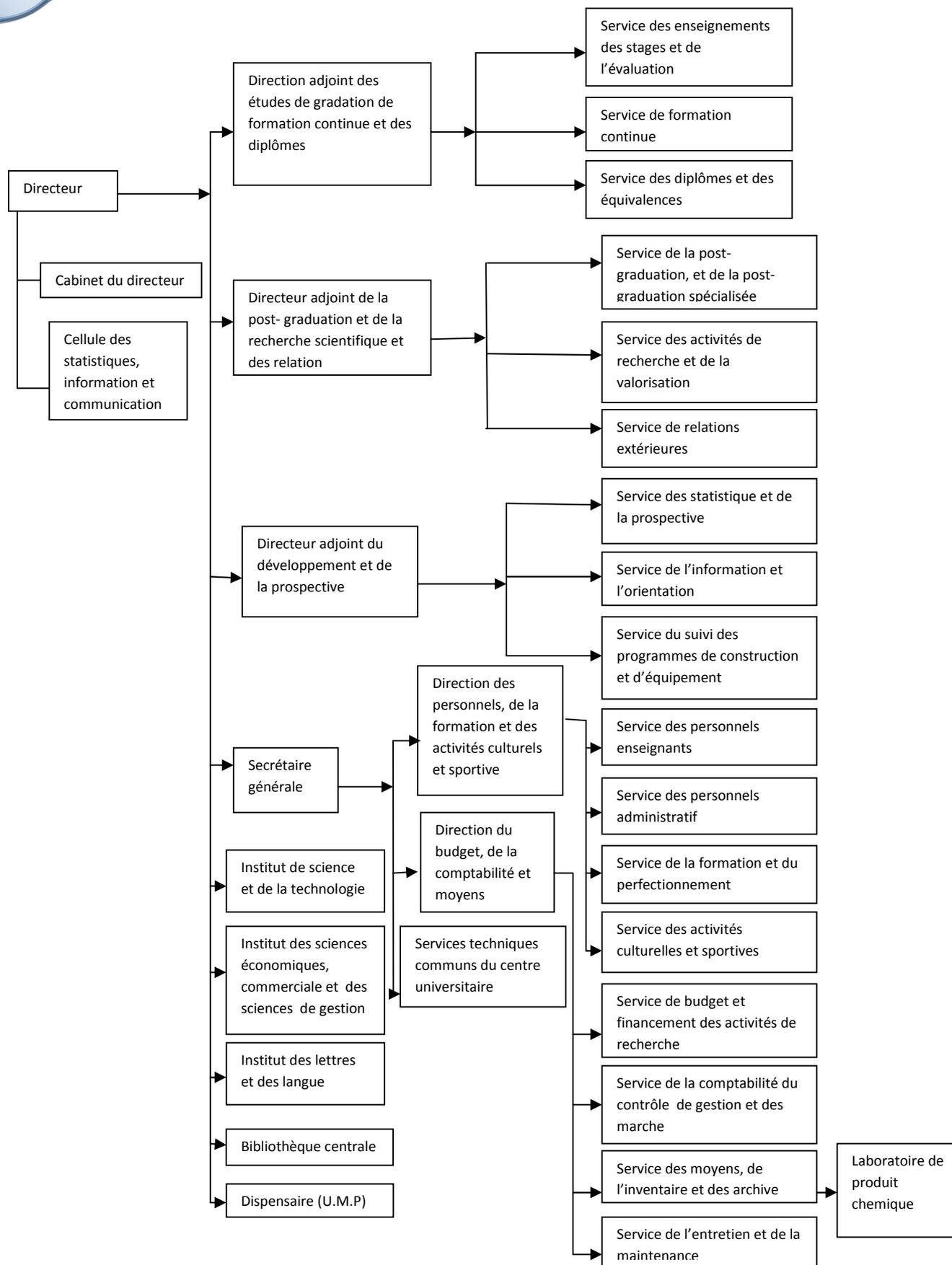


Figure II.1 Organigramme du centre universitaire de Mila.

# Chapitre II: Présentation de l'organisme d'accueil

## II. Représentation du laboratoire de produits chimiques

Pour couvrir tous les besoins des étudiants en département science de la nature et de la vie et département de sciences et technologies afin de réaliser leurs travaux pratiques, l'université a créé le laboratoire de produits chimiques le 12 -10-2010, le stock de produits à débiter son travail le 28-10 -2010. Le laboratoire possède un seule gérant qui gère un nombre important de produits chimiques et le matériel nécessaire pour faire les travaux pratiques des étudiants.

-Le stock est composé d'une seule chambre pour le moment, et il est situé au rez-de-chaussée par mesure de sécurité.

## III. Définition d'un produit chimique

Les produits chimiques ont plusieurs définitions et il n'existe pas une définition précise. On peut les définir comme :

- Tous ce qu'on trouve dans le monde physique (un solide, un liquide, un gaz).
- Substances provenant de la transformation d'une ressource naturelle par un traitement chimique, telles que les produits d'entretien, les produits de beauté, les matières plastiques.

### III.1 Type de produits chimiques

Il existe deux types de produits chimiques dans le laboratoire du centre universitaire de Mila:

**III.1.1 Solide** : Sous forme de poudre.

**III.1.2 Liquide** : Soit sous forme d'acide ou sous forme de base.

➤ **Un acide** : est une molécule ou un ion qui en solution donne des ions  $H^+$  ( $H_3O^+$ ). Il y a deux catégories :

- **Les acides forts** : qui sont ionisés totalement ou presque totalement en solution aqueuse diluée.
- **Les acides faibles** : sont ionisés légèrement.

## Chapitre II: Présentation de l'organisme d'accueil



Dans cet exemple,  $\text{HNO}_3$  est un acide et  $\text{H}_2\text{O}$  agit en tant que base.

$\text{NO}_3^-$  est appelé la **base conjuguée** de l'acide  $\text{HNO}_3$  et  $\text{H}_3\text{O}^+$  est l'**acide conjugué** de la base  $\text{H}_2\text{O}$ .

➤ **Une base:** est une molécule ou un ion qui en solution capte des ions  $\text{H}^+$ .



Dans cet exemple, le  $\text{NH}_3$  est une base de  $\text{H}_2\text{O}$  et agit comme un acide.  $\text{NH}_4^+$  est l'acide conjugué de la base de  $\text{NH}_3$ , et  $\text{OH}^-$  est la base conjuguée de l'acide  $\text{H}_2\text{O}$ .

Un composé qui peut agir comme un acide ou une base, telle que le  $\text{H}_2\text{O}$  dans les exemples ci-dessus, est appelée **amphotère**.

### III.2 Caractéristique d'un produit chimique :

1. La désignation qui possède deux formes alphabétique et code chimique.
2. Un type (solide, liquide).
3. Date fabrication, date d'expiration.
4. Le nom commercial.
5. Degré de pureté.
6. Composants.
7. Poids monoculaires.
8. Degré de risque.
9. Densité.
10. Symbole de danger.

**Exemple** de symbole de danger d'étiquetage de produit chimique dans le centre universitaire de Mila:

## Chapitre II: Présentation de l'organisme d'accueil

Matières inflammables et combustibles	
Matières comburantes	
Matière toxiques et infectieuses ayant des effets toxique immédiats et graves	

**Tableau II.1** : de symbole de danger d'étiquetage de produit chimique dans le centre universitaire de Mila.

### III.3 Choix des produits chimiques

La sélection des produits chimiques à utiliser dans les laboratoires du centre universitaire de Mila est basée sur plusieurs éléments :

- Les besoins du programme;
- L'importance des expériences qui fournies aux étudiants.
- Les risques chimiques;
- La probabilité d'utiliser des produits chimiques dans plusieurs activités.
- La maturité, les connaissances et les compétences des étudiants;
- La disponibilité d'autres activités et matériels;
- Les installations de stockage et l'équipement de laboratoire disponibles;
- Les considérations environnementales et les coûts liés à l'élimination.

### III.4 Choix de Quantité commandée

Lorsque vous déterminez la quantité d'un produit chimique particulier à commander, tenez compte des facteurs suivants :

- Le taux de consommation.
- La stabilité du produit chimique (la plupart des sels inorganiques et des acides et bases dilués stockés dans les écoles ne se détériorent pas au fil du temps);
- L'utilisation future du produit chimique;
- L'espace de stockage disponible;

## Chapitre II: Présentation de l'organisme d'accueil

- Les ressources financières.

### III.5 Installations de stockage des produits chimiques

Les risques associés à l'utilisation de produits chimiques peuvent être considérablement réduits en stockant tous les produits chimiques dans des installations adéquates. Une zone de stockage de produits chimiques idéale est une zone séparée à l'extérieur des classes (magasin au rez-de-chaussée).

En plus il faut respecter quelque règle d'installation pour évité le risque comme protège les produits chimiques des rayons du soleil et des températures extrêmes; équipé le stock par des lampes, d'interrupteurs et de ventilateur anti explosion afin d'éviter les incendies provoqués par les courts-circuits électriques ou des étincelles d'interrupteurs défectueux.

Les produits sont stockés dans un placard de fer avec plusieurs étagères tel que les articles solides sont dans les 'étagères supérieures, les étagères inférieures sont pour les articles liquides.

Le stock est accessible uniquement au personnel autorisé (Le gérant).



Figure II.2 Placard d'installation de produits chimiques.

## Chapitre II: Présentation de l'organisme d'accueil

### IV L'étude des postes de travail

Il existe un seul poste de travail : le gérant de labo.

**Fiche d'étude du poste de travail: le gérant.**

<b>Fiche d'étude de poste de travail</b>
<b>Désignation:</b> Gestion du laboratoire
<b>Responsable:</b> Le gérant
<b>Responsabilité:</b> La gestion des produits chimiques.
<b>Moyenne utilisé:</b> Ordinateur, téléphone, imprimante
<b>Les tâches à accomplir</b>
La réception des produits chimiques
La tenue de stock (ajout, modification, suppression des articles).
Rangement des produits dans le stock.
Distribution des produits selon les besoins des enseignants (Le bon de commande), établissement des bons de sortie.

**Tableau II.2 :** Fiche d'étude du poste de travail gérant.

### V L'étude des procédures

#### V.1 Entrée d'un produit

Dès l'arrivée des articles, le gérant décharge la livraison dans la salle de réception et commence la vérification de la quantité, la qualité plus les contraintes du cahier de charges).

- Enfin, il range les produits dans le magasin et inscrit ces derniers dans le registre du stock.
- Pour le suivi de l'état du stock, il procède à la mise à jour de la fiche de stock (Quantité entrée, prix, date d'entrée...etc. Pour chaque produit).

#### V.2 Extraire les produits seuil alerte

Chaque quinzaine, le gérant contrôle l'état du stock pour recenser les articles en rupture (dont la quantité a atteint le seuil d'alerte), A la fin il établit un bon de commande destiné au magasin central.

#### V.3 Extraire les produits périmés

Cette opération permet d'obtenir Les articles qui ont atteint la date d'expiration.

## Chapitre II: Présentation de l'organisme d'accueil

A la fin il établit une fiche des produits périmés et il diminue les quantités des produits expirés.

### V.4 Traitement des bons internes (sortie d'un produit)

Lors de la réception d'un bon interne, le gérant vérifie la disponibilité des produits et la possibilité de satisfaire la commande. Enfin, il fait la mise à jour de la fiche de stock et établit un bon de sortie.

## VI. L'étude des documents

Durant notre stage, nous avons étudié les documents utilisés par le gérant, et nous avons collecté une liste des données les plus importantes pour notre application. Les documents utilisés sont:

- Registre du stock.
- Registre exemplaire ou registre de stock.
- Fiche de situation de produits chimiques.
- Bon de commande.
- Bon de sortie.
- Décharge.

Désignation	Code	Signification	Type
Fiche de stock	Codep	Code du produit	N
	Désignation	Désignation du produit	A
	Date	Date d'entrée ou de sortie de produit	D
	N-b	Numéro de bon entrée ou sortie	N
	Q-e	La quantité entée au magasin	N
	Q-s	La quantité sortie du magasin	N
	Q-r	La quantité restée dans magasin	N
Bon de sortie	Dates	Date de sortie du produit	D
	Nbons	Le numéro de bon de sortie	N
	Codep	Le code de produit sorti	N
	Qtéf	La quantité fournie	N
	Nom et prenom	Le nom et prénom de l'enseignant)	N
	obsevation	Observation de gérant	AN
Bon de commande	Nbc	Le numéro de bon de commande	N
	Datec	La date du bon de commande	D

## Chapitre II: Présentation de l'organisme d'accueil

	Codep	Le code de produit demandé	N
	NomEn	Le nom de l'enseignant	A
	PrénomEn	Le prénom de l'enseignant	A
	Qtéc	La quantité commandée	AN
	observation	L'observation	A
Le gérant	login	Le login de gérant	A
	Nomg	Le nom de gérant	A
	Prénomg	Le prénom de gérant	N
	Numtélg	Le numéro de téléphone de gérant	N
	Adrg	L'adresse de gérant	A
	password	Le mot de passe	N
L'enseignant	CodeEn	Le code de l'enseignant	N
	NomEn	Le nom de l'enseignant	A
	PrénomEn	Le prénom de l'enseignant	N
	adrEn	L'adresse de l'enseignant	N
	numtelEn	Le numéro de téléphone d'enseignant	
Fiche de situation De produit chimique.	Codep	Code de produit	N
	Désignation	Désignation de produit	A
	Q-e	La quantité entrée au magasin	AN
	Q-s	La quantité sortie du magasin	AN
	Q-r	La quantité restée dans magasin	AN
Décharge	Codep	Code de produit	N
	Désignation	Désignation de produit	A
	Q-e	La quantité entrée au magasin	AN
	Unité	Unité de produit chimique	AN
Registre exemplaire de registre de stock	Codep	Code de produit	N
	Désignation	Désignation de produit	A
	Date	Date d'entrée ou de sortie de produit	D
	N-b	Numéro de bon entrée ou sortie	N
	Nom	Le nom de l'utilisateur	A
	Prénom	Le prénom de l'utilisateur	A
Fournisseur	Code-f	Le code du fournisseur	N
	Nom-f	Le nom du fournisseur	A
	Prénom-f	Le prénom du fournisseur	A
	Adr-f	L'adresse du fournisseur	A
	Num-tel-f	Le numéro de téléphone du fournisseur	N

**Tableau II.3:** les documents utilisés par le gérant dans le centre universitaire de Mila.

## Chapitre II: Présentation de l'organisme d'accueil

**Légende des types :**

**A** : Alphabétique. **N** : Numérique. **AN** : Alpha Numérique.

### VII. Critiques et suggestions

À la cour de notre stage, nous avons constaté quelques défaillances dans la gestion des produits chimiques au sein du laboratoire:

Toute la gestion au sein du laboratoire est manuelle, la recherche d'un produit pour une demande est très longue qui cause une perte de temps. Il faut faire la demande au minimum deux jours à l'avance.

Lorsqu'un produit est absent ou sa quantité est insuffisante, l'enseignant est le responsable d'achat de produit afin de pouvoir réaliser les TP pour ses étudiants.

Pour résoudre ces problèmes, nous suggérons d'automatiser la gestion des produits chimiques.

### Conclusion

Dans ce chapitre, nous avons présentés l'organisme où nous avons effectué notre stage (laboratoire des produits chimiques du centre universitaire de Mila).

# Chapitre III

## L'étude préliminaire

**I. Présentation général du projet (système)**

**II . La définition des grands choix techniques**

**III. Spécification des besoins fonctionnels**

**IV. Spécifications des besoins non fonctionnels**

**V. Descriptions du contexte du système**

## Introduction

Pour faire un projet en suivant le processus de développement UP simplifié, il existe plusieurs étapes, le premier est l'étude préliminaire (pré étude) qui consiste à spécifier les besoins fonctionnels et non fonctionnels du système sans se préoccuper des détails du système (on considère le système comme une boîte noire).

### I. Présentation général du projet (système)

L'objectif du projet est la conception et la réalisation d'un système pour gérer les produits chimiques au sein du laboratoire du centre universitaire de Mila.

Les grandes fonctionnalités de ce système sont :

- L'ajout d'un nouveau produit.
- La consultation des produits.
- La gestion (traitement) des commandes interne.
- La suppression des produits.
- La modification d'un produit.

### II. La définition des grands choix techniques

Pour assurer la fiabilité de notre système d'information et pouvoir répondre aux besoins des utilisateurs, nous avons utilisé :

- Un langage de modélisation standard pour la conception orienté objet (UML).
- Un processus de développement (UP simplifié).
- L'interface et la création de la Base de données avec le langage Delphi version 7.
- Windows 7 comme système d'exploitation.

### III. Spécification des besoins fonctionnels

#### III.1 L'ajout d'un produit

Lors de la réception d'un nouveau produit, le gérant ajoute ce produit dans la fiche de stock et le range dans l'endroit adéquat.

## III.2 L'établissement d'un bon de sortie

Lorsque le gérant reçoit un bon de tirage (commande), il vérifie la disponibilité et la quantité des produits demandés, diminue la quantité en stock des produits concernés, et enfin il imprime un bon de sortie.

## III.3 L'extraire les produits seuil alerte

Chaque quinzaine, le gérant contrôle l'état du stock pour recenser les articles en rupture (dont la quantité a atteint le seuil d'alerte), A la fin il établit un bon de commande destiné au magasin central.

## III.4 L'extraire les produits périmés

Le gérant vérifier l'état de stock des articles qui ont atteint la date d'expiration.

A la fin il établit une fiche de produit périmé et il diminue les quantités des produits expirés.

## III.5 La suppression d'un produit

Lorsqu'un produit n'est plus utilisable par le laboratoire, le gérant procède à son élimination de la base de données.

## III.6 La modification d'un produit

La modification consiste à faire des changements sur les informations concernant un produit. La modification la plus courante est celle de la quantité en stock (lors de la réception de nouvelles quantités d'un produit).

## III.7 La consultation d'un produit

Cette opération utilisé Pour obtenir un produit existe dans la base de donné.

## IV. Spécifications des besoins non fonctionnels

Il existe des critères utilisés pour faciliter et sécuriser l'utilisation du système :

- ✓ Utilisation des mots de passe et login.
- ✓ Les fiches de saisies doivent être claires et conviviales.

## V. Descriptions du contexte du système

### V.1 Identification des acteurs

Un acteur représente un rôle joué par une entité externe (utilisateur humain, dispositif matériel ou autre système) qui interagit directement avec le système étudié. Un acteur peut Consulter et/ou modifier directement l'état du système, en émettant et/ou recevant des messages éventuellement porteurs de données.

**Le gérant du laboratoire** est le seul acteur dans notre système.

### V.2 Identification des messages

Un message représente la spécification d'une communication entre objets qui transporte l'information avec l'intention de déclencher une activité chez le récepteur. Cette notion de message est également tout à fait applicable pour décrire les interactions de plus haut niveau entre les acteurs et le système.

**Pour chaque acteur** on identifie les messages qui déclenchent une activité du système.

**Pour le système** on identifie les messages émis vers un acteur particulier et qui portent une information utilisée par l'acteur.

Les messages du gérant de laboratoire sont :

- Ajouter produit.
- Etablir bon de sortie.
- Extrait produit seuil alerte.
- Extrait produit périmé.
- Supprimer produit.
- Modifier produit.
- consulter produit.

### V.3 Réalisation du diagramme de contexte

Tous les messages échangés entre les acteurs et le système vont être représentés sur un diagramme que nous appelons diagramme de contexte ou le système étudié est représenté par un objet central entouré par d'autres objets. Des liens relient le système à chacun des acteurs et sur chaque lien, nous montrons les messages en entrée et en sortie du système.

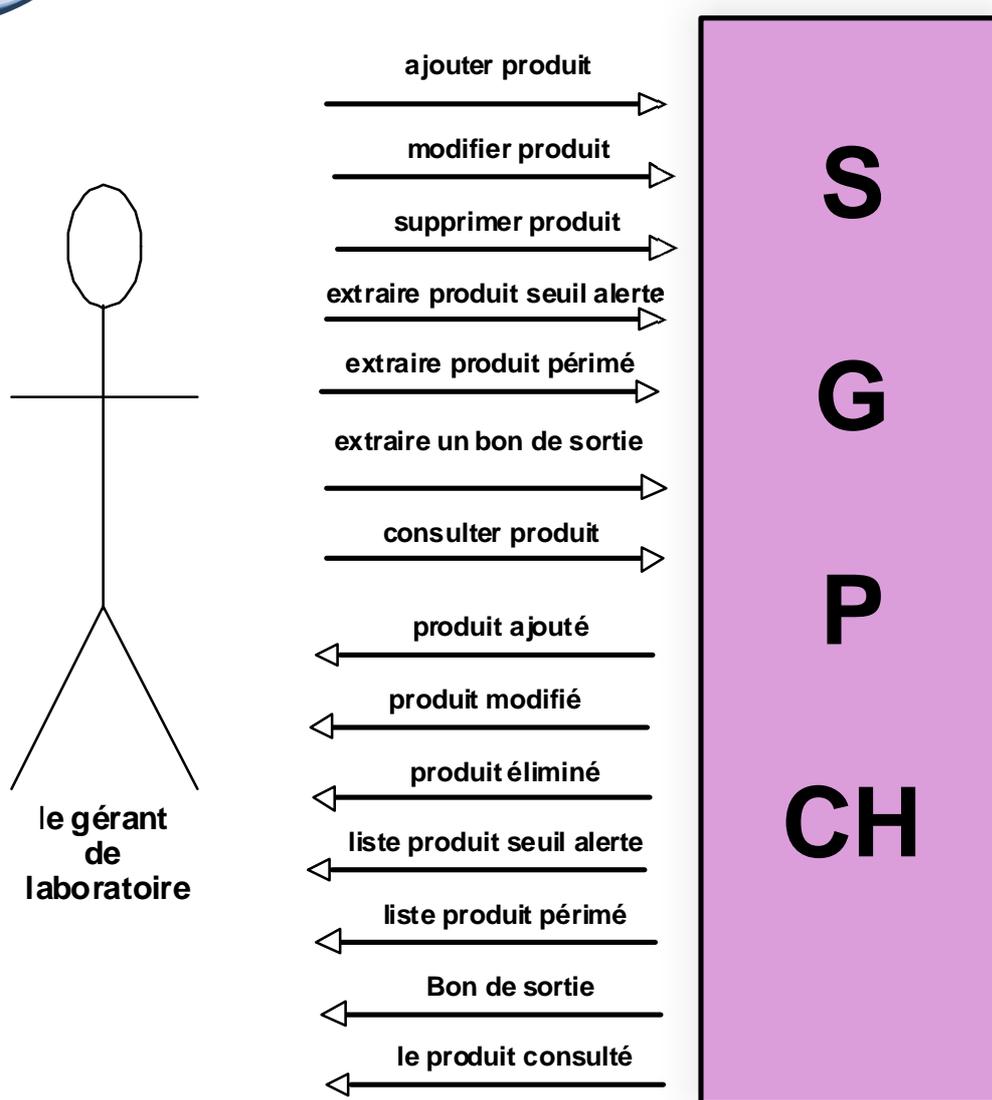


Figure III.1 : Diagramme de contexte

## Conclusion

Dans ce chapitre, nous avons spécifié les besoins fonctionnels et non fonctionnels du système, à la fin nous avons modélisé le contexte du système par le diagramme de contexte.

# *Chapitre IV*

## *Analyse et*

## *Conception du système*

**I. Diagramme de cas d'utilisation**

**II. Description des cas d'utilisation**

**III. Le modèle de domaine**

## Introduction

Dans ce chapitre, nous allons détailler l'étude du contexte fonctionnel du système, en écrivant les différentes façons qui auront les acteurs d'utiliser le future système.

Les tâches réalisées dans ce chapitre sont:

- Décrire les cas d'utilisation
- La description d'un cas d'utilisation, sera consolidée par un diagramme d'activités.
- Réaliser le diagramme de séquence.
- Réaliser le diagramme de classe.

### I Diagramme de cas d'utilisation

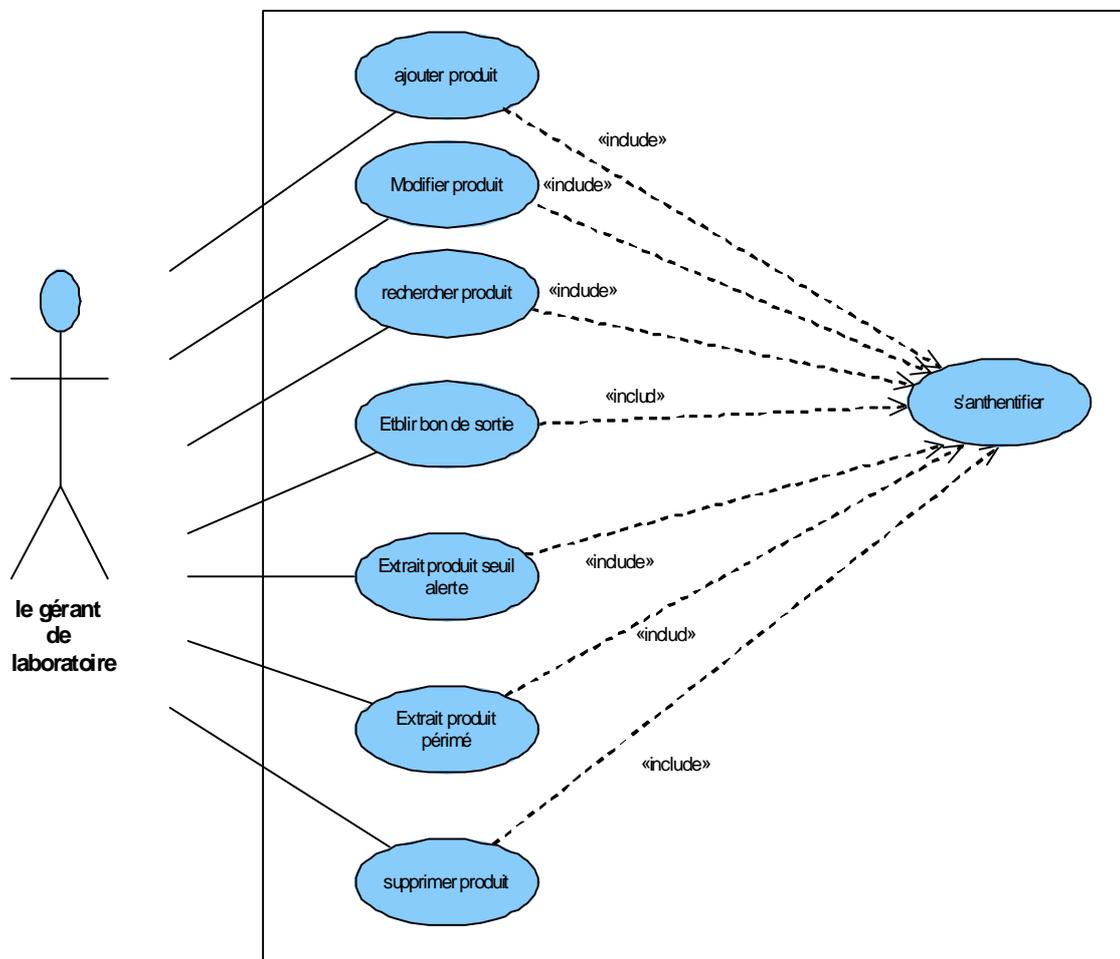


Figure IV.1: Diagramme de cas d'utilisation

## II Description des cas d'utilisation

### II.1 Description des scénarios

#### II.1.1 Cas d'utilisation: Authentification.

**Titre : Authentification**

**Finalité :** Ce cas permet au gérant d'utiliser le système en toute sécurité.

**Acteur :** Le gérant du laboratoire.

**Pré condition :** Le gérant du laboratoire doit avoir un mot de passe et login.

**Post conditions :** Le menu de l'application est accessible (accéder au système).

#### Scénario nominal

1. Le système affiche une fenêtre pour qu'il s'identifie.
2. Le gérant du laboratoire saisi son login et son mot de passe.
3. Le système vérifie leur validité puis lance le menu principal de l'application.

#### Scénario alternatif :

1. Login ou mot de passe erroné, l'authentification est demandée à nouveau au maximum trois fois:
  - le système afficher un message d'erreur.
  - Reprendre le scénario au point 2.

#### Scénario exceptionnel:

Le gérant ne saisit pas le vrai mot de passe. L'application se ferme.

**II.1.2 Cas d'utilisation: Ajouter produit.**

**Titre :** Ajouter produit.

**Finalité :** Ce cas permet d'ajouter un nouveau produit à la base.

**Acteurs :** Le gérant du laboratoire.

**Prés condition :** Le gérant s'authentifie. L'arrivée d'une décharge pour ce produit.

**Post condition :** Un nouveau produit est ajouté.

**Scénario nominal:**

1. Le système affiche le formulaire « ajouter produit chimique ».
2. Le gérant du laboratoire saisit le code du produit chimique.
3. Le gérant du laboratoire saisit les informations de produit chimique.
4. Le système vérifie les informations saisies.
5. Le système demande la validation de l'ajout.
6. Le système ajoute le produit chimique à la base de données.

**Scénario alternatif :**

1. Le code du produit saisit est erronée :
  - le système afficher un message d'erreur.
  - Reprendre le scénario au point 2.
2. Les informations saisit sont incomplètes ou erronés (incorrectes) :
  - le système afficher un message d'erreur.
  - Reprendre le scénario au point 3.

**Scénario exceptionnel :**

Le gérant du laboratoire ne valide pas l'ajout donc l'opération est annulée.

**II.1.3 Cas d'utilisation: Modifier produit chimique**

**Titre :** Modifier produit chimique.

**Finalité :** Modifier les propriétés d'un produit chimique.

**Acteur :** Le gérant du laboratoire.

**Prés condition :** Le gérant s'authentifie. Le produit chimique existe déjà.

**Post condition :** Les propriétés du produit chimique sont modifiées.

**Scénario nominal :**

1. Le système affiche le formulaire « modifier produit chimique ».
2. Le gérant de labo saisit le code de produit chimique.
3. le système affiche les autres informations.
4. Le gérant du laboratoire modifier les propriétés de produit chimique (désignation, quantité,...).
5. Le système demande de valider la modification.
6. Le gérant valide la modification.
7. le système enregistre la modification.

**Scénario alternatif :**

1. Le code du produit saisit est erronée :
  - le système afficher un message d'erreur.
  - Reprendre le scénario au point 2.
2. Les informations saisit sont incomplètes ou erronés (incorrectes) :
  - le système afficher un message d'erreur.
  - Reprendre le scénario au point 4.

**Scénario exceptionnel :**

Le gérant du laboratoire ne valide pas l'opération, la modification est annulée.

**II.1.4 Cas d'utilisation:** Supprimer produit chimique**Titre :** Supprimer produit chimique.**Finalité :** Éliminé un produit chimique existant.**Acteur :** Le gérant du laboratoire.**Pré condition:** Le produit chimique existe déjà.**Post condition:** Le produit chimique est éliminé.**Scénario nominal:**

1. Le système affiche le formulaire de suppression.
2. Le gérant du laboratoire saisit le code du produit chimique.
3. le système demande la validation de suppression.
4. Le système supprime le produit chimique.

**Scénario alternatif :**

1. Les codes saisis sont erronés (incorrectes) :
  - le système afficher message erreur.
  - Reprendre le scénario au point 2.

**Scénario exceptionnel :**

Si le gérant du laboratoire ne valide pas l'élimination alors l'opération est abandonnée.

**II.1.5 Cas d'utilisation:** Extraire produit seuil alerte.**Titre :** Extraire produit seuil alerte.**Finalité :** Etablir une liste des articles dont la quantité approche la quantité minimale (le seuil d'alerte)**Acteurs :** Le gérant.**Prés condition :** Le gérant doit s'authentifier.**Post condition :** Un bon de commande des articles en seuil d'alerte est imprimé.**Scénario nominal :**

1. Le gérant demande au système d'afficher la liste des produits qui ont atteint le seuil minimum en stock.
2. Le système affiche tous les articles dont la quantité en stock est inférieure ou égale au seuil minimal.
3. Le gérant imprime un bon de commande au magasin.

**Scénario alternatif :**

1. La liste ne contient aucun produit:  
-Le gérant n'établit pas un bon de commande.

**II.1.6 Cas d'utilisation:** Extraire produit périmé.**Titre :** Extraire produit périmé**Finalité :** Ce cas permet d'obtenir les produits périmés.**Acteur principal :** Le gérant.**Pré condition:** Le gérant s'est authentifié. Le contrôle des produits en stock.**Post condition :** La quantité des produits périmés est diminuée.

Un bon de commande des articles périmé est imprimé.

**Scénario nominaux:**

1. Le gérant demande de visualiser la liste des produits périmés.
2. Le système affiche tous les articles dont la date d'expiration est inférieure ou égale à la date système (courant).
3. Le gérant modifie la quantité en stock des produits périmés.
4. Le gérant imprime un bon de commande.

**Scénario alternatif :**

1. La liste ne contient aucun produit.
2. Le gérant ne modifie rien.

**II.1.7 Cas d'utilisation:** Etablir bon de sortie

**Titre:** Etablir bon de sortie.

**Finalité:** Etablir un bon de sortie pour les produits demandés.

**Acteur :** Le gérant du laboratoire.

**Pré condition :** Le gérant du laboratoire reçoit un bon de commande (bon de tirage).

**Post condition :** Etablir un bon de sortie.

**Scénario nominal :**

1. Le gérant demande le formulaire d'un bon de sortie.
2. Le système affiche le formulaire de bon de sortie.
3. Le gérant remplit ce formulaire.
4. Le système demande la validation des informations.
5. Le système imprime le bon de sortie.
6. Le gérant modifie la quantité en stock.

**Scénario alternatif :**

1. Le formulaire incomplet:
  - le système afficher un message d'erreur.
  - Reprendre le scénario au point 3.

**Scénario exceptionnel :**

Si le gérant du laboratoire ne valide pas, les opérations diminuer quantité en stock et imprimer sont annulées.

## II.1.8 Cas d'utilisation: Consulter produit

**Titre:** Consulter produit.

**Finalité:** Consulter les produits en stock selon leur type, désignation,...

**Acteur :** Le gérant du laboratoire.

**Pré condition :** Le gérant doit s'authentifier.

**Post condition :** Aucun changement sur la base de données.

### Scénario nominal :

1. Le système affiche le formulaire de consultation.
2. Le gérant choisit les critères de recherches.
3. Le système affiche la liste de produits qui vérifient les critères choisis.

## II.2 Description des scénarios par les diagrammes d'activités

## II.2.1 Cas d'utilisation &lt;&lt;Authentification&gt;&gt; :

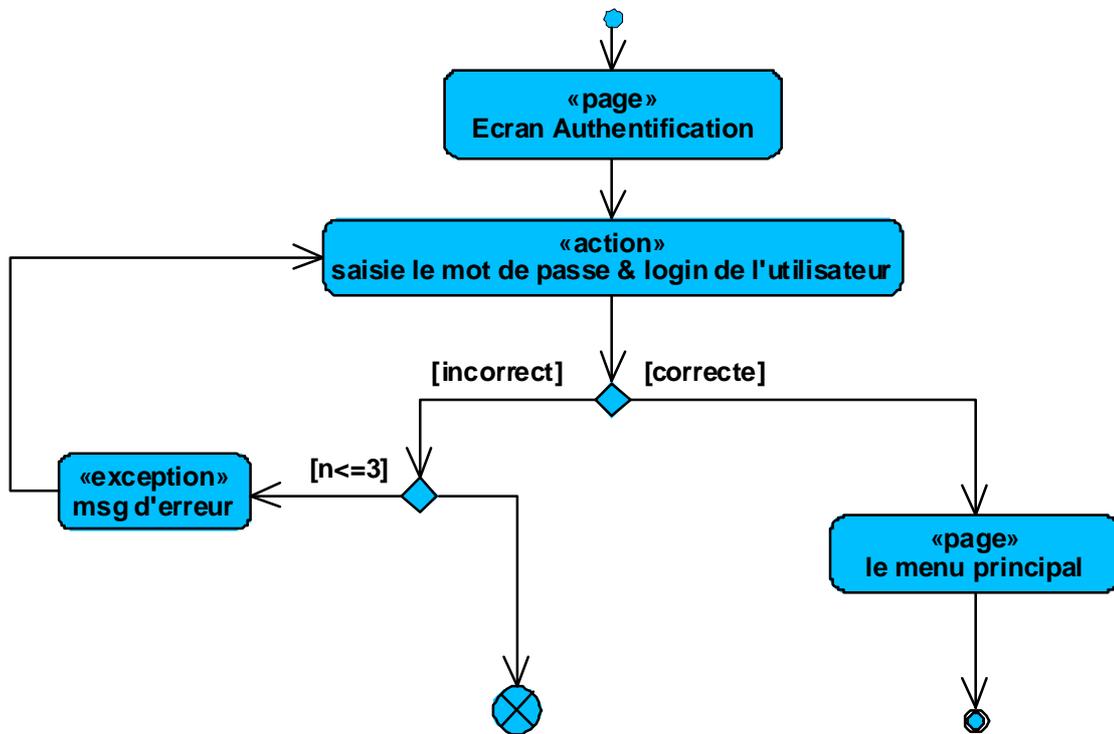


Figure IV.2: Diagramme d'activité « S'authentifier »

## II.2.2 Cas d'utilisation « Ajouter produit » :

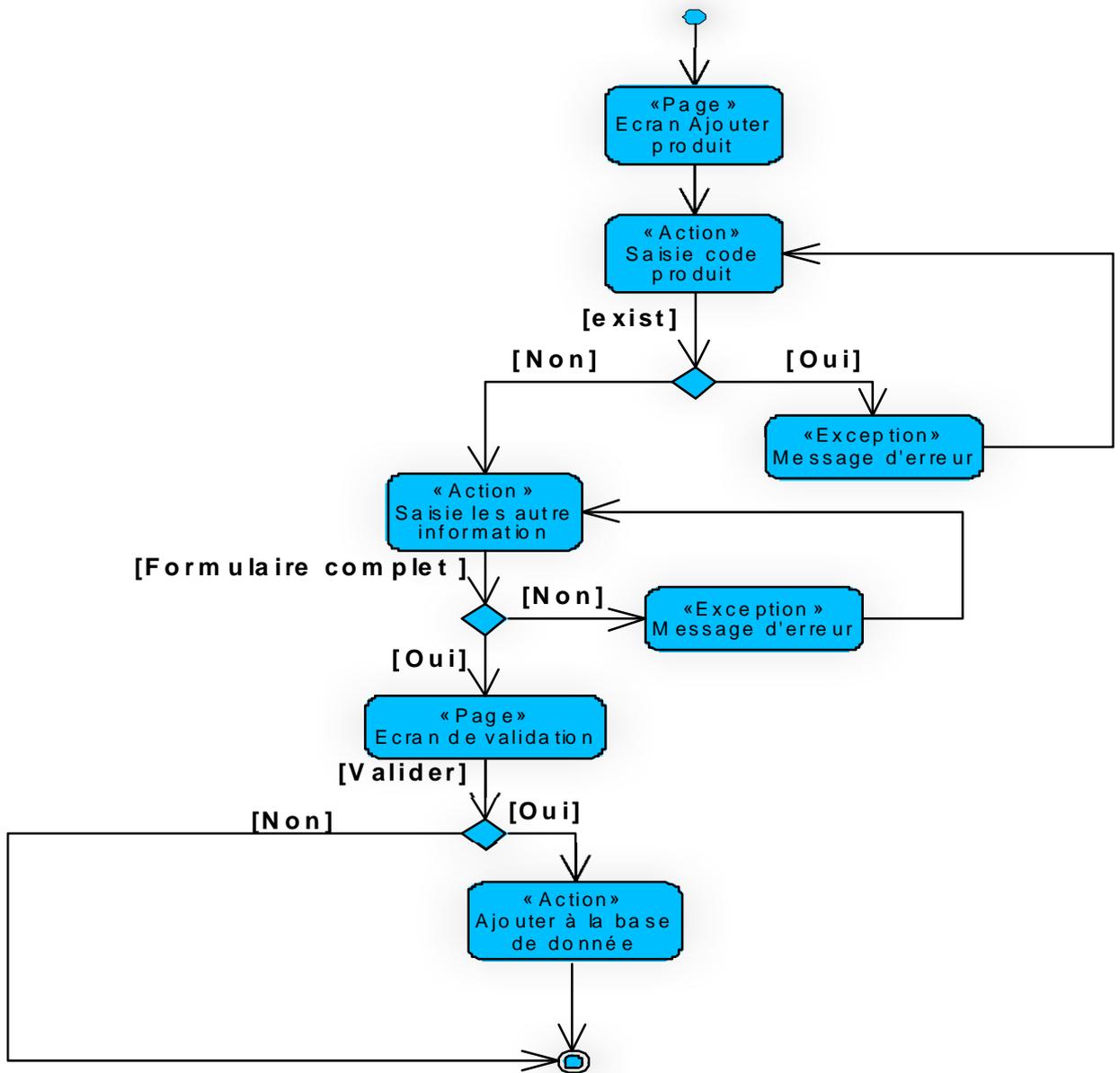


Figure IV.3: Diagramme d'activité « Ajouter produit »

## II.2.3 Cas d'utilisation « Modifier produit chimique » :

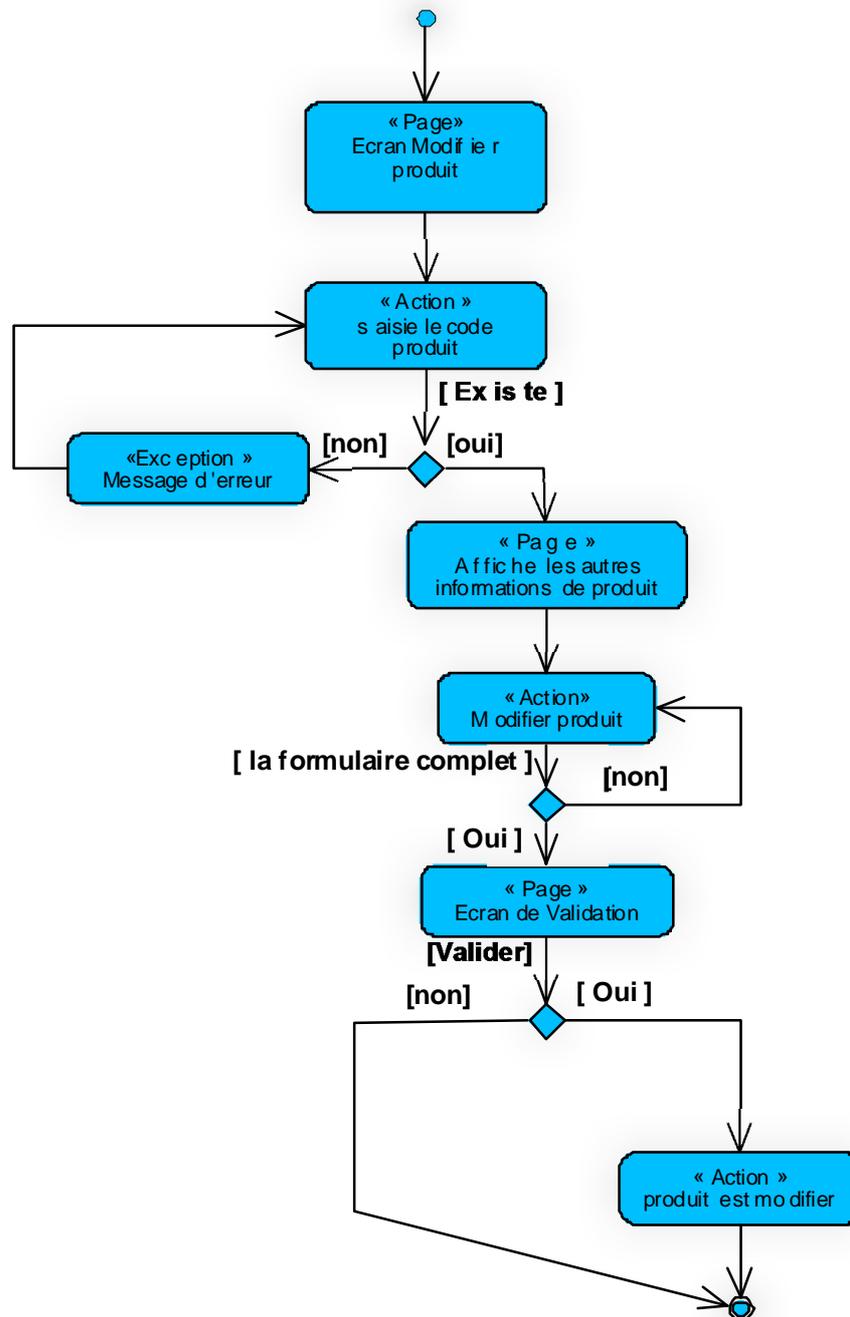


Figure IV.4: Diagramme d'activité « Modifier produit »

## II.2.4 Cas d'utilisation « Supprimer produit chimique »:

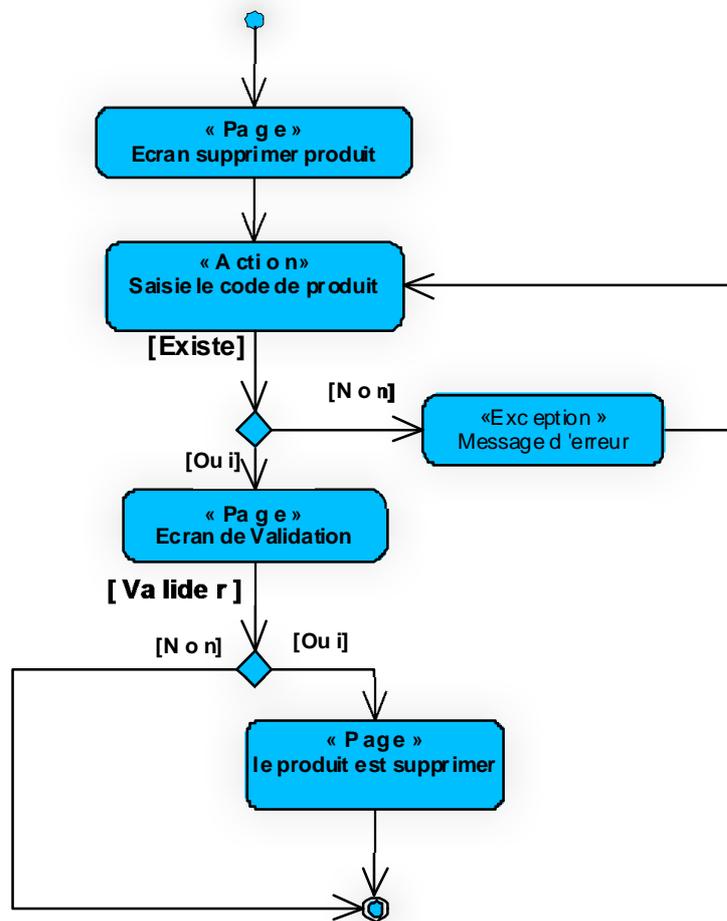


Figure IV.5: Diagramme d'activité « supprimer produit »

## II.2.5 Cas d'utilisation «Extraire produit seuil alerte» :

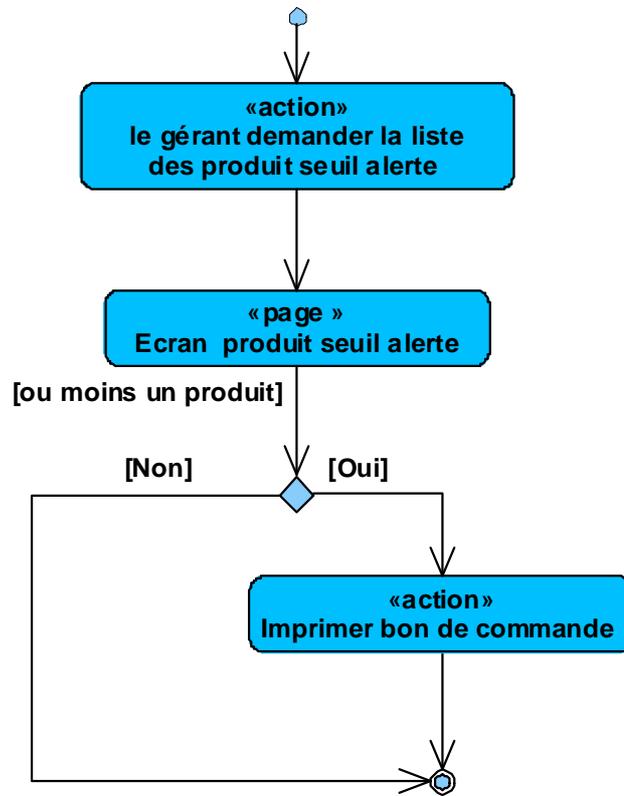


Figure IV.6: Diagramme d'activité «extraire produit seuil alerte»

## II.2.6 Cas d'utilisation «Extraire produit périmé» :

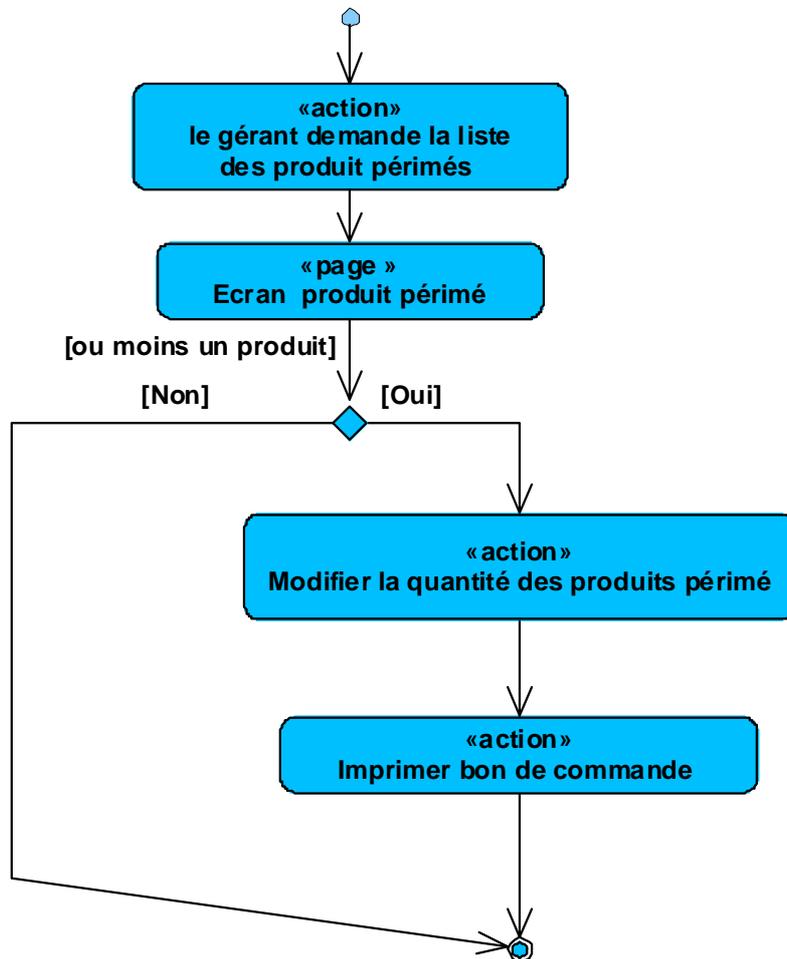


Figure IV.7: Diagramme d'activité «Extraire produit périmé»

## II.2.7 Cas d'utilisation «Etablir bon de sortie» :

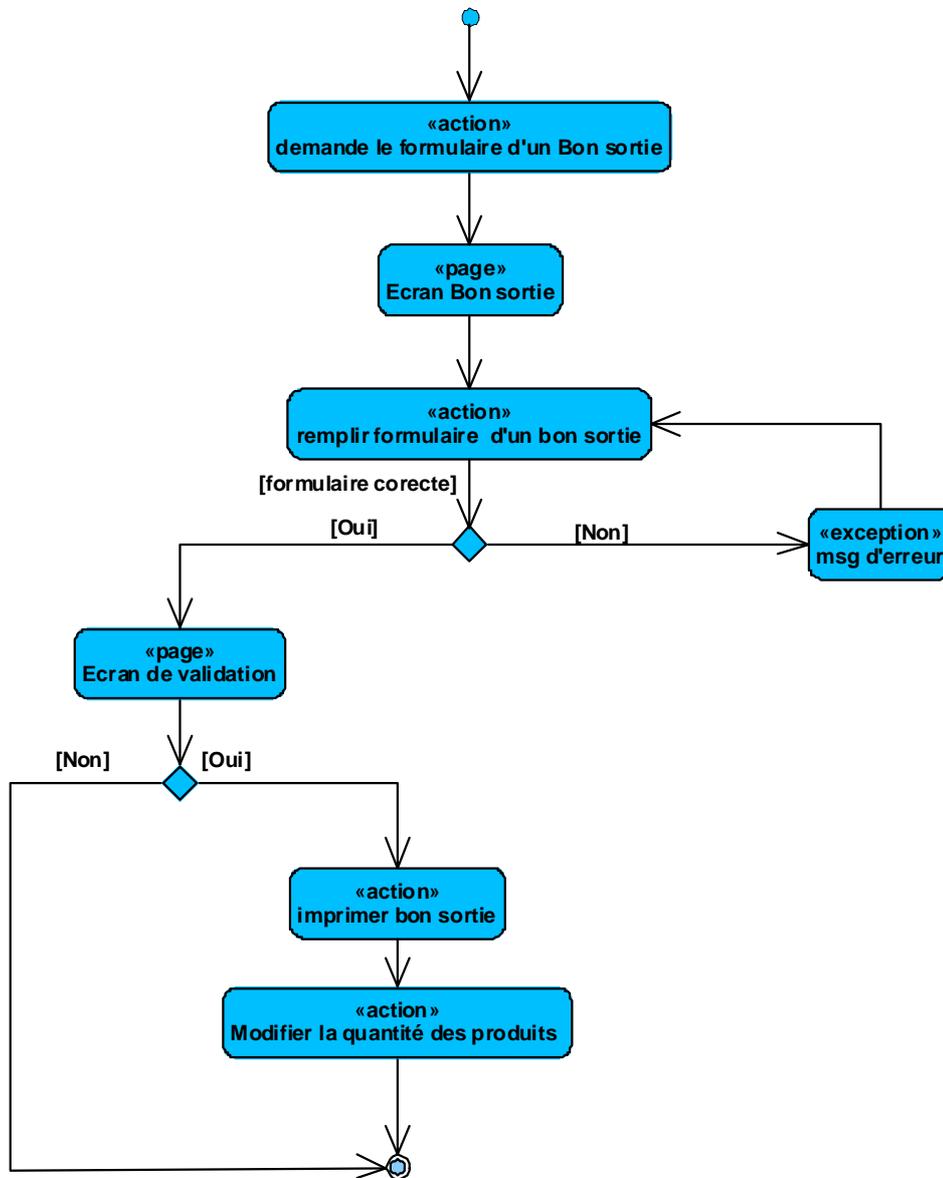


Figure IV.8: Diagramme d'activité «Etablir bon de sortie»

## II.2.8 Cas d'utilisation «Consulter produit» :

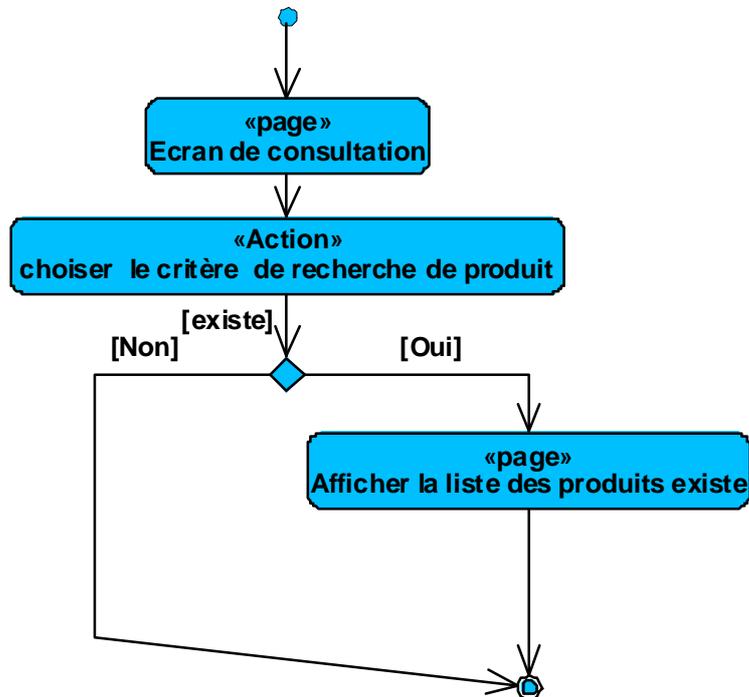


Figure IV.9: Diagramme d'activité «Consulter produits»

II.3 Description des scénarios par les diagrammes de séquences

II.3.1 Cas d'utilisation <<S'authentifier>> :

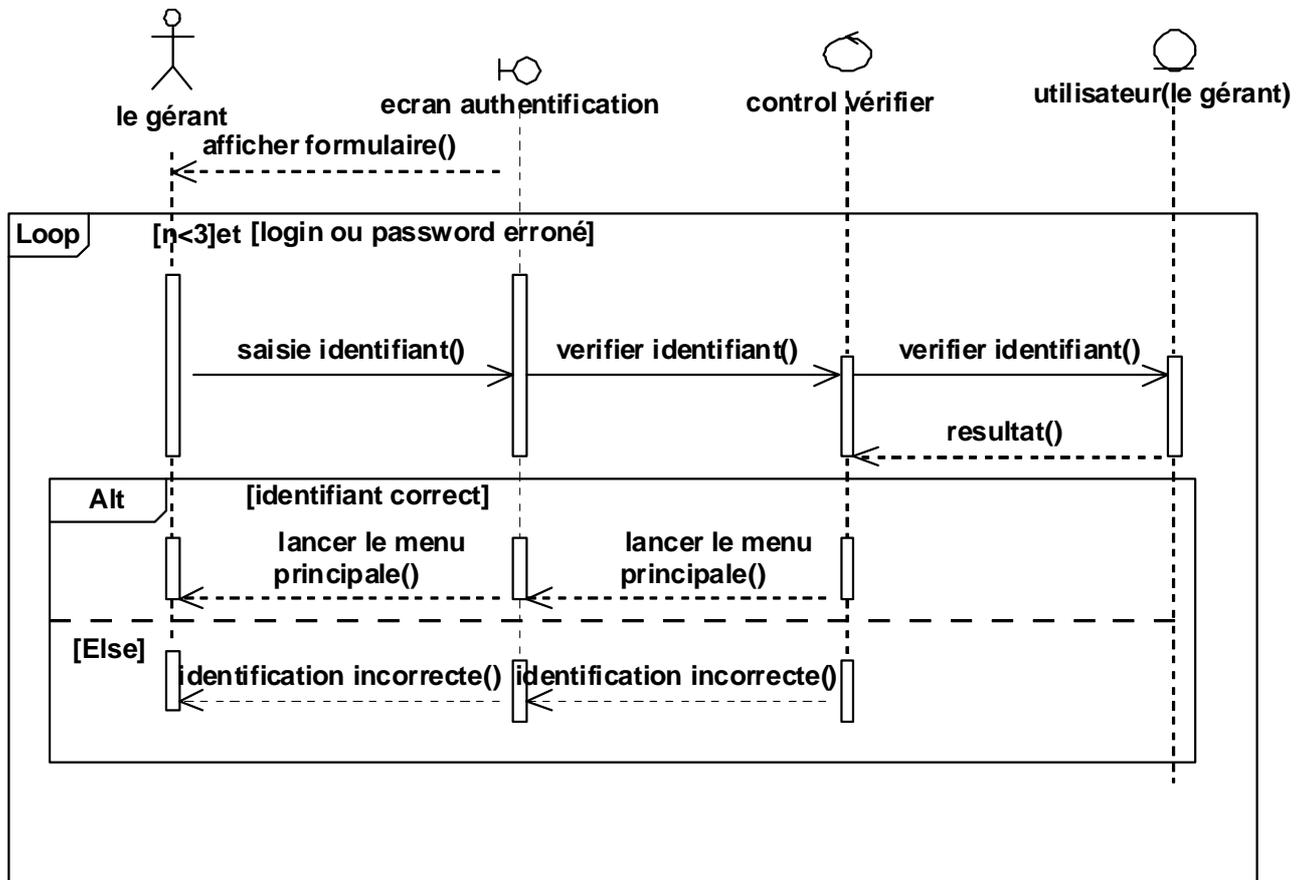


Figure IV.10: Diagramme de séquences <<S'authentifier>>.

II.3.2 Cas d'utilisation «Ajouter produit »:

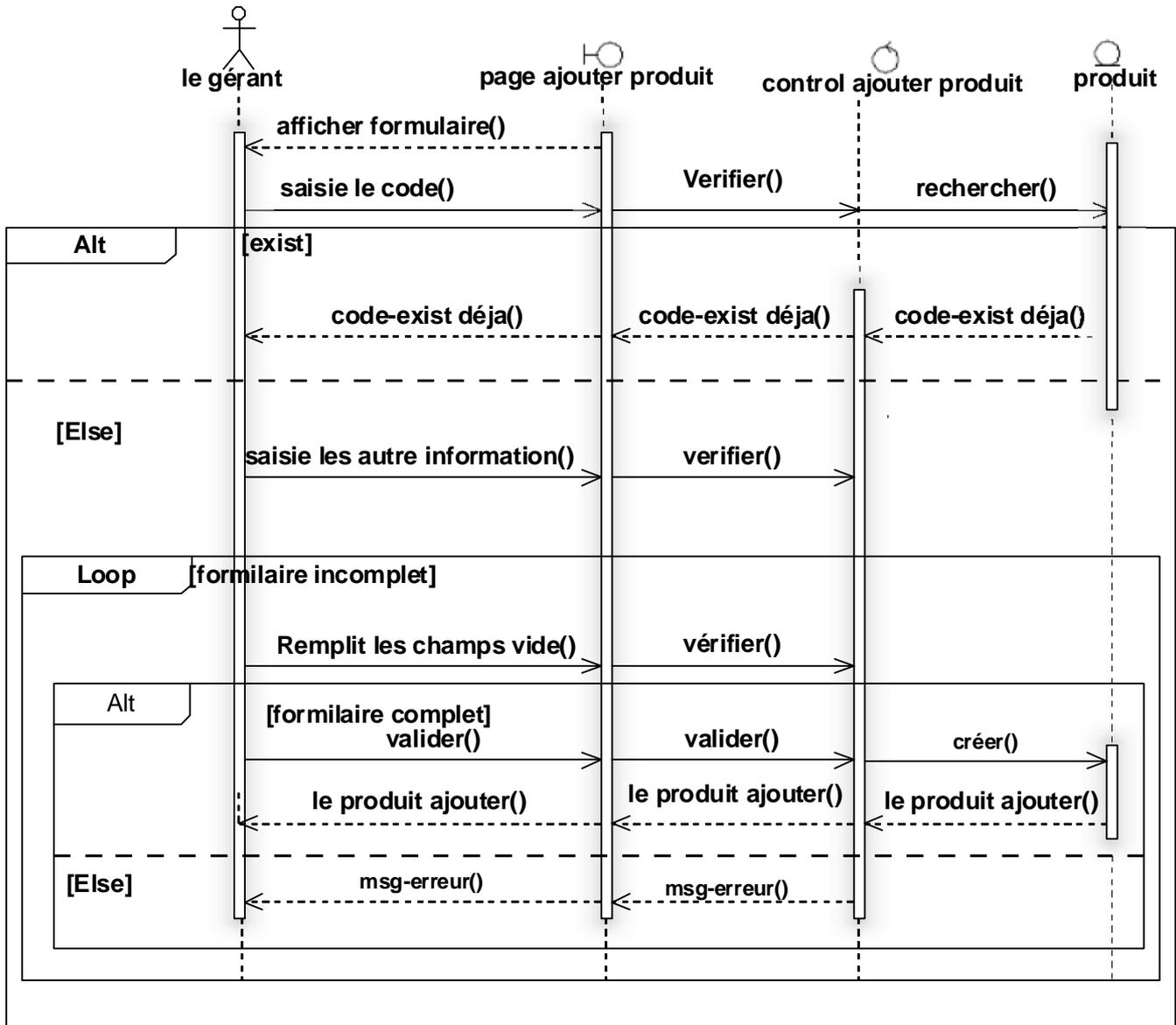


Figure IV.11: Diagramme de séquences « Ajouter produit »

II.3.3 Cas d'utilisation <<Modifier produit chimique>> :

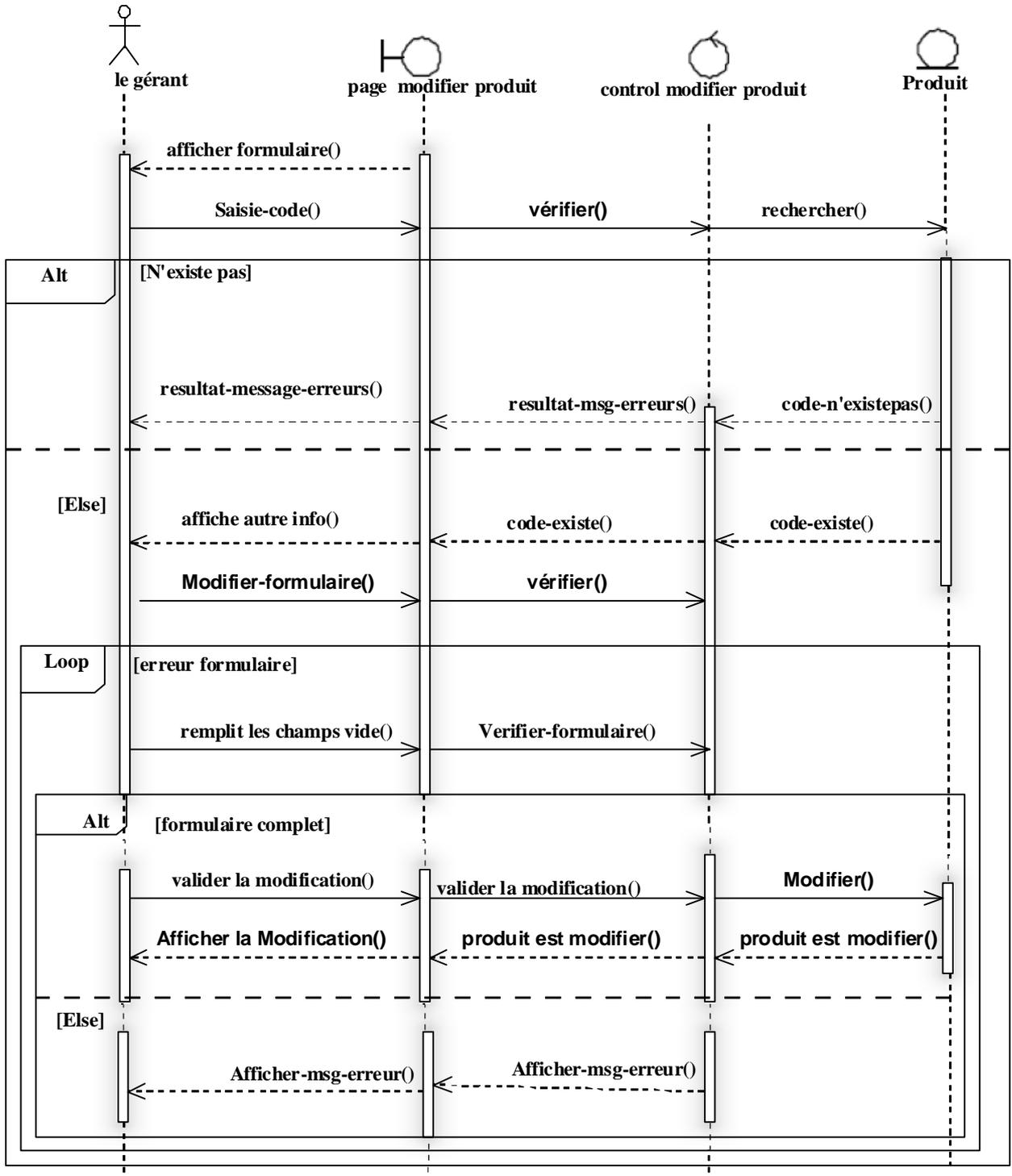


Figure IV.12: Diagramme de séquences « Modifier produit »

## II.3.4 Cas d'utilisation «Supprimer produit chimique» :

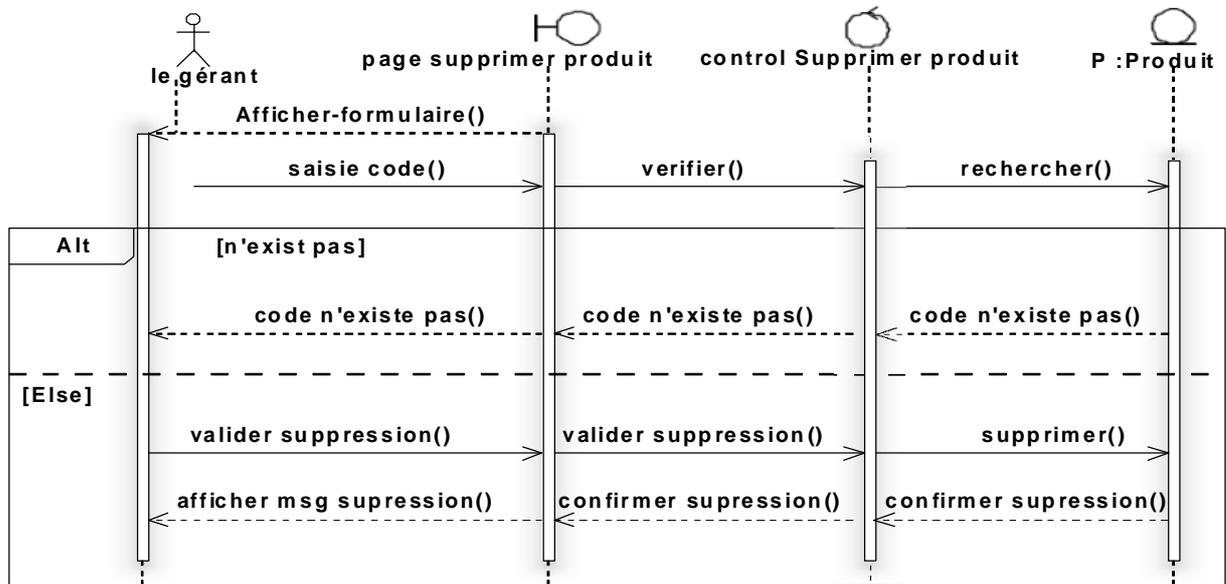


Figure IV.13: Diagramme de séquences « supprimer produit »

## II.3.5 Cas d'utilisation «Extraire produit seuil alerte» :

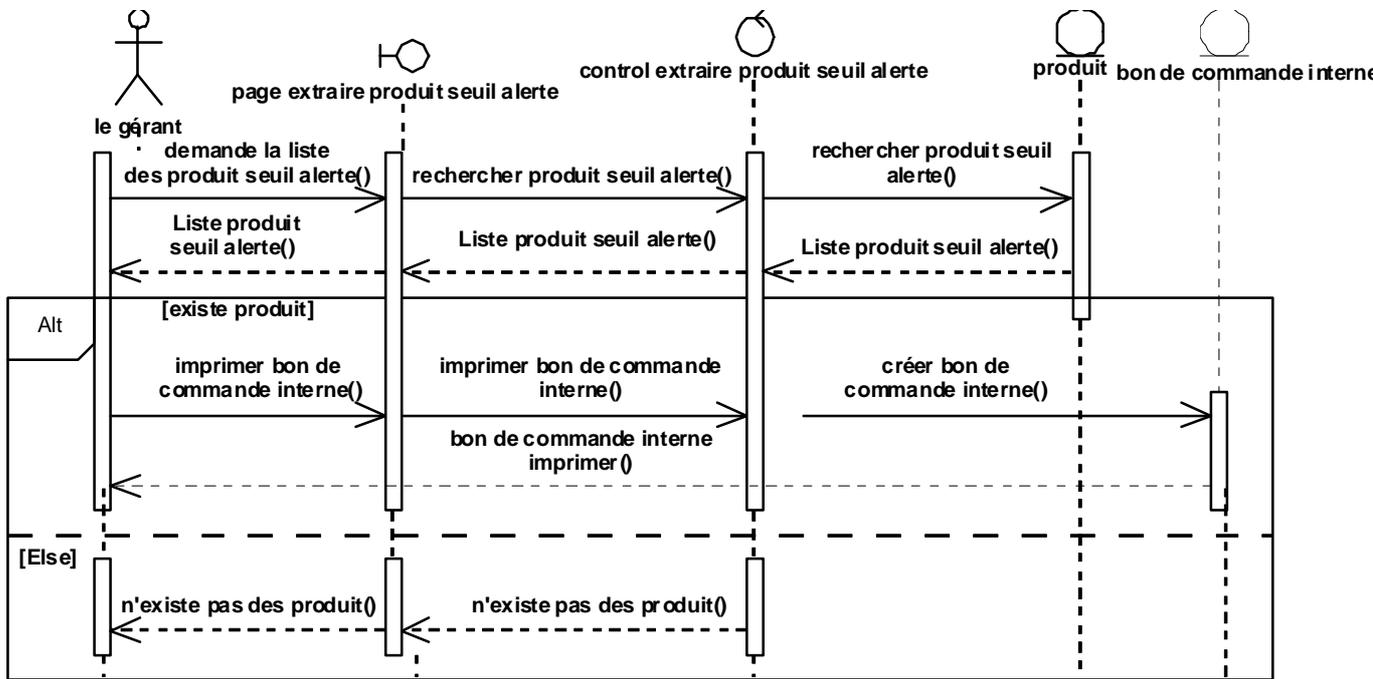


Figure IV.14: Diagramme de séquences «extraire produit seuil alerte»

II.3.6 Cas d'utilisation «Extraire produit périmé» :

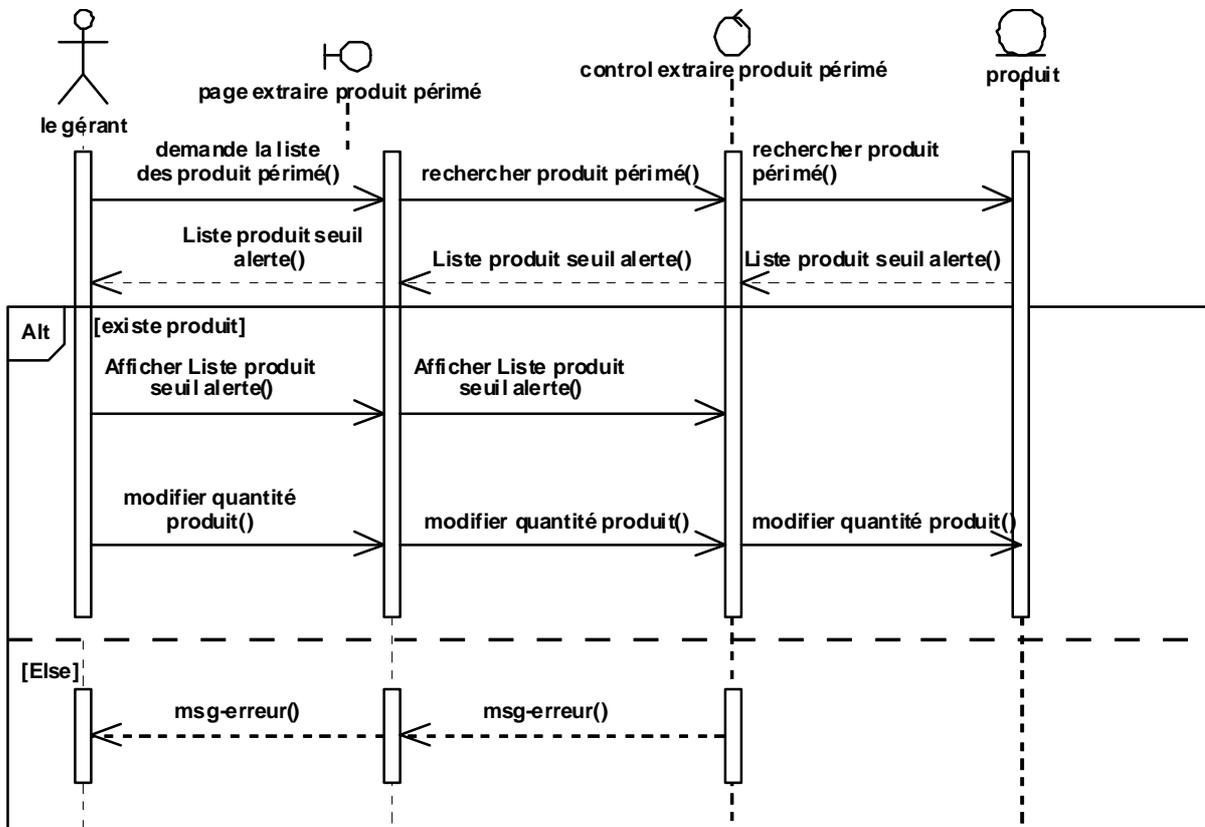


Figure IV.15: Diagramme de séquences «Extraire produit périmé»

## II.3.7 Cas d'utilisation « Etablir bon de sortie » :

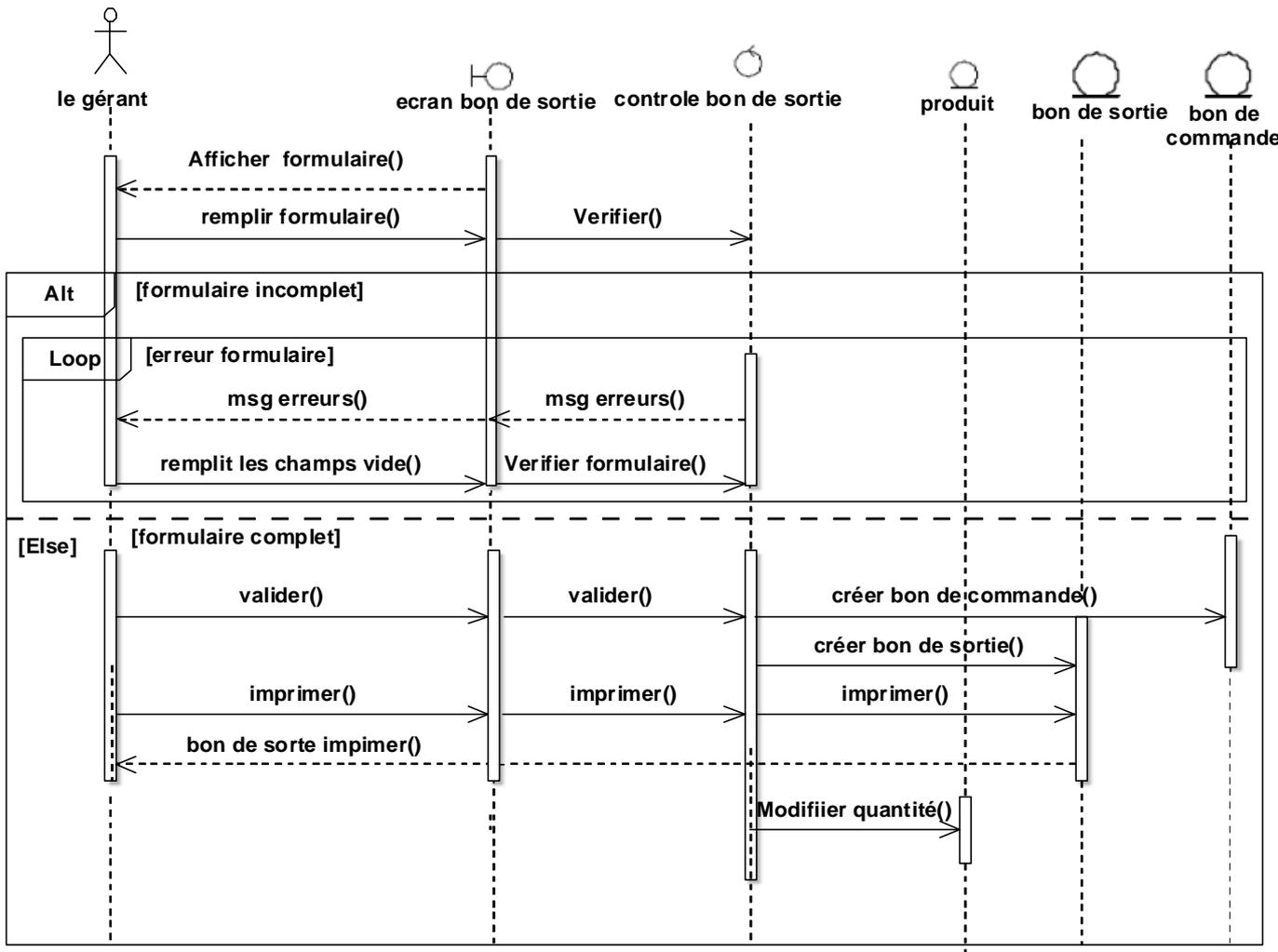


Figure IV.16: Diagramme de séquences «Etablir bon de sortie»

## II.3.8 Cas d'utilisation « Consulter produit » :

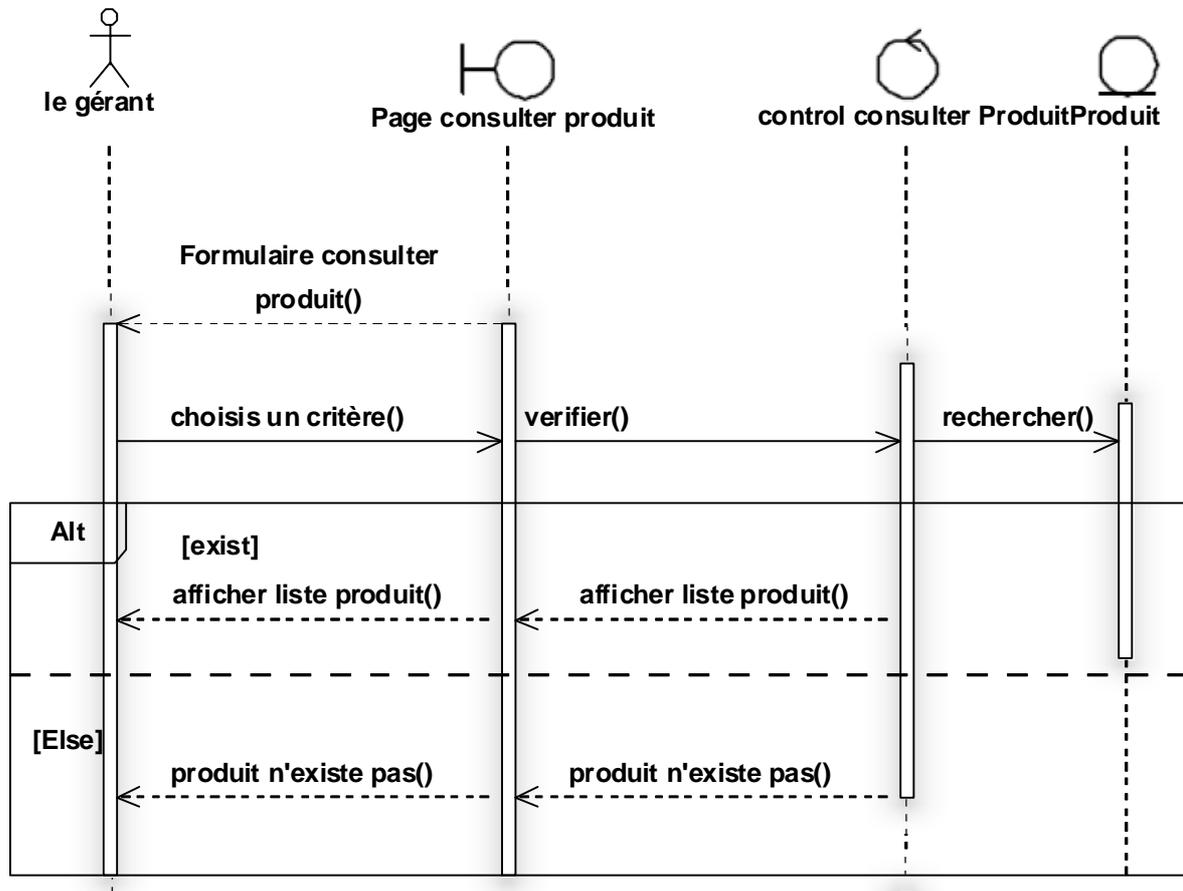


Figure IV.17: Diagramme de séquences « Consulter produits »

## III. Le modèle de domaine

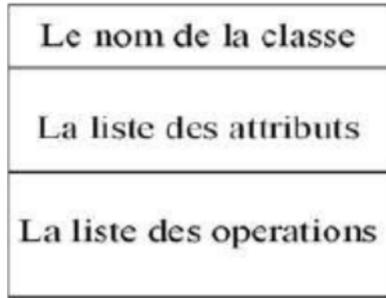
Dans ce chapitre nous allons identifier un certain nombre de classes issues du domaine d'étude ces classes sont appelées classes métier et l'ensemble de ces classes métier constitue le modèle de domaine.

## III.1 Représentation d'une classe UML

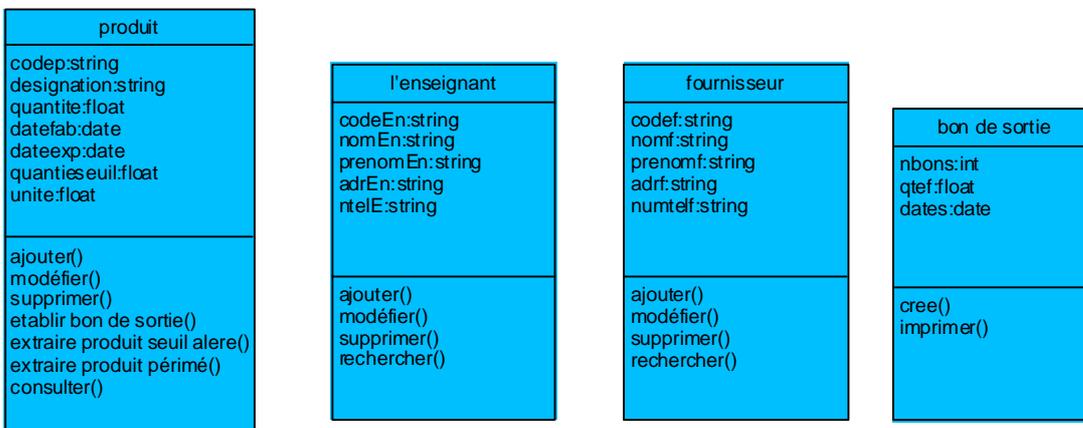
Puisque UML est un langage de modélisation objet, il faut qu'il puisse représenter une classe.

Comme le montre la figure, une classe est décrite par un rectangle composé de trois compartiments:

1. le nom de la classe;
2. la liste des attributs;
3. la liste des Operations.



## III.2 Identification des classes métier



III.3 Diagramme des classes

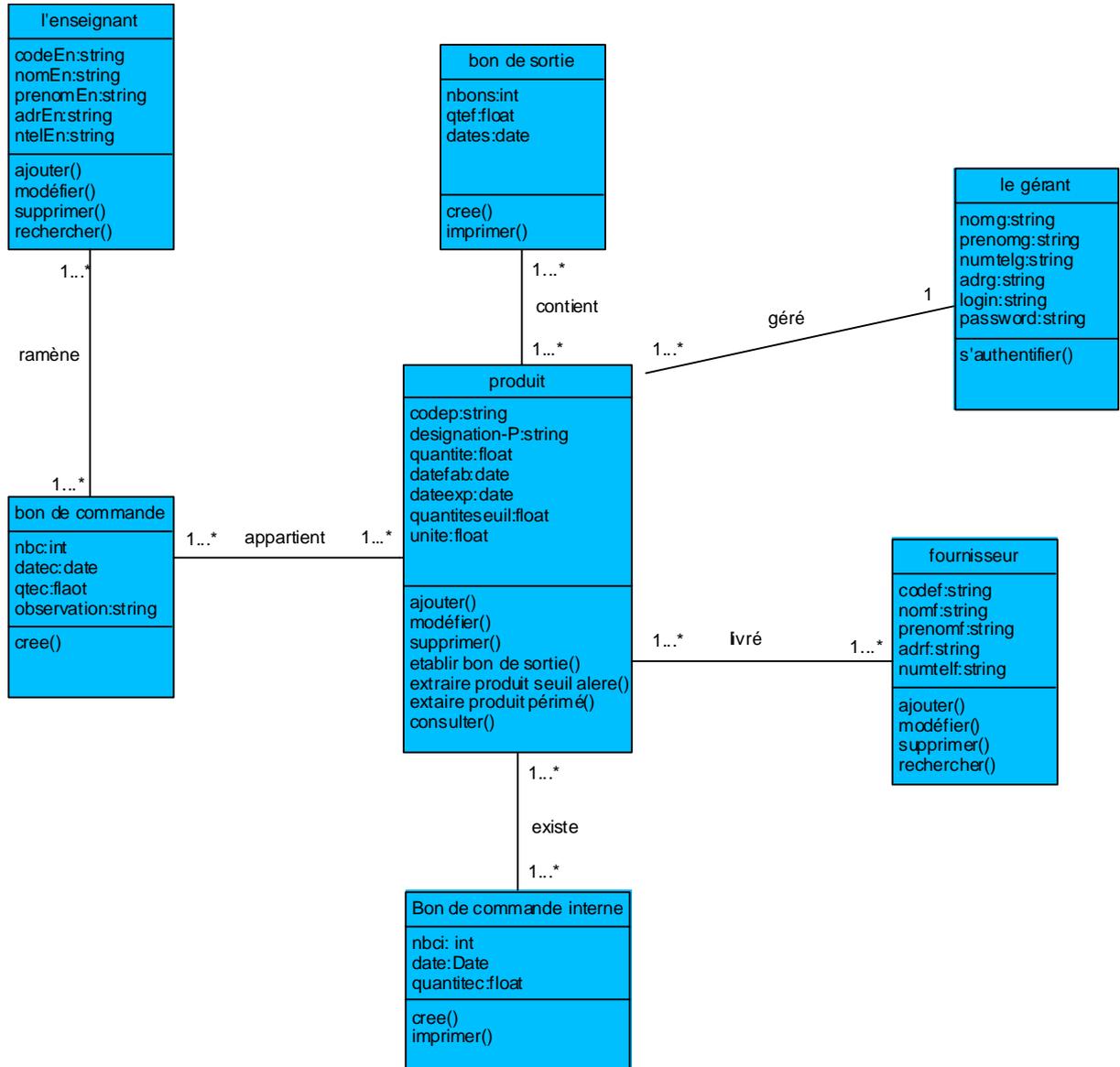


Figure IV.18: Diagramme de classes

## Conclusion

Dans ce chapitre, nous avons utilisé le langage de modélisation UML pour réaliser l'analyse et la conception de notre application.

# *Chapitre V*

## *Réalisation du système*

**I. les outils de travail**

**II. Le passage du diagramme de classe au modèle relationnel**

**III. Description de l'implémentation du système**

## Introduction

Après avoir terminé l'étude conceptuelle, nous pouvons alors passer à l'étape finale de ce mémoire à savoir l'implémentation. Elle a comme objectif d'aboutir à un produit final exploitable par les utilisateurs. Dans cette phase nous allons préparer notre base de données en convertissant le diagramme de classe que nous avons obtenue dans le chapitre précédent en une base de données relationnelle. Par la suite, nous présenterons les différents outils que nous avons utilisés pour le développement de notre application, puis nous décrivons quelques interfaces de l'application, afin d'illustrer les grandes et principales fonctionnalités réalisées.

## I.les outils de travail

### I.1 Pacestar UML Diagrammer 6.02

Un programme qui fournit un ensemble complet d'outils de modélisation graphique, d'analyse et de conception dans le développement de logiciels basés sur les modèles UML, COM,OMT

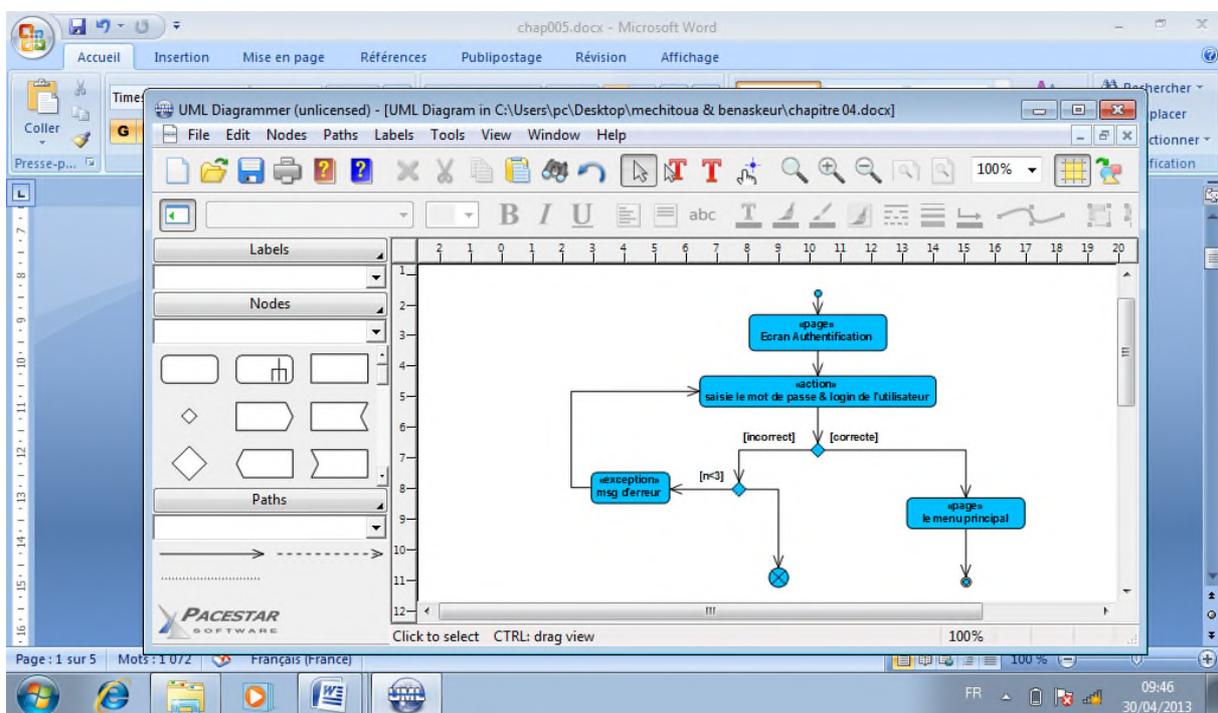


Figure V.1: Pacestar UML.

## I.2 Environnement de développement de l'application

### I.2.1 Présentation de Delphi

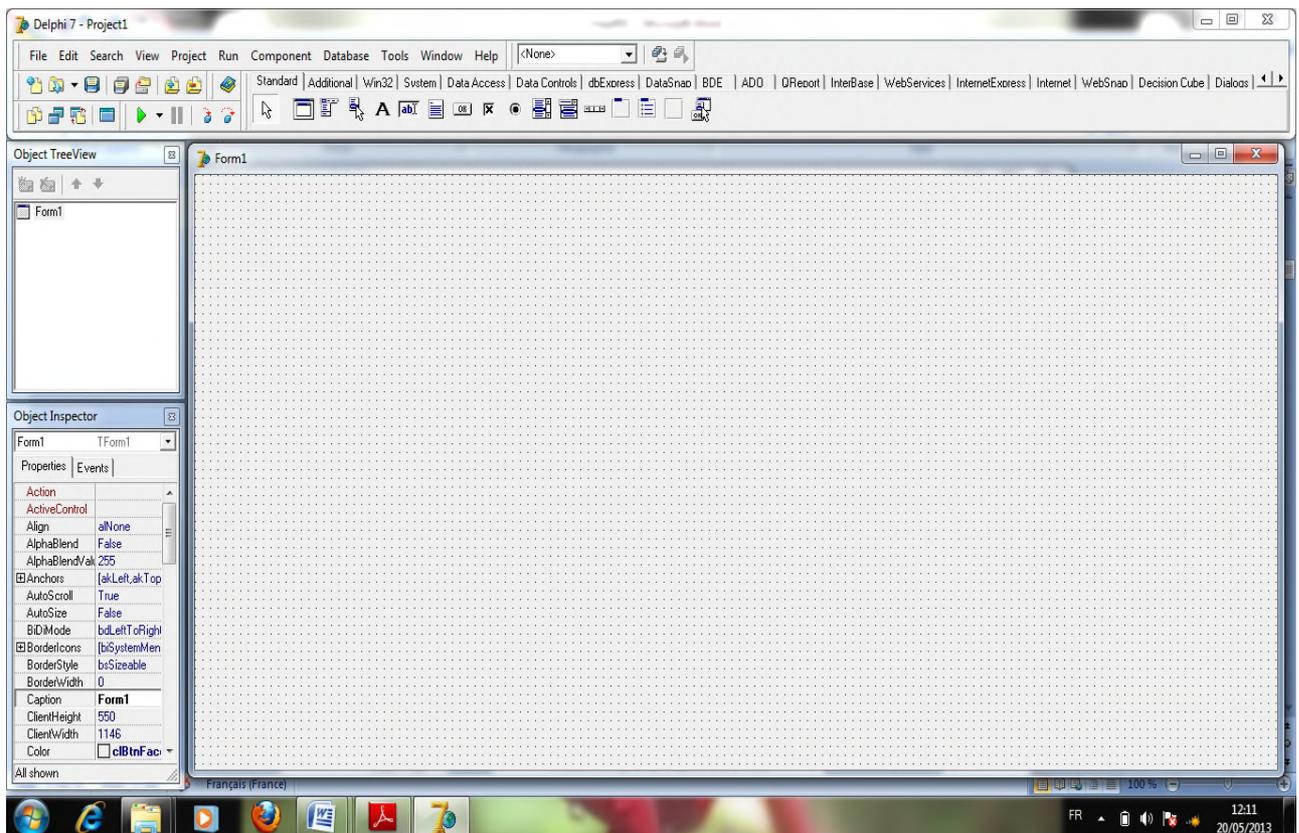
Delphi est un environnement de programmation visuel orienté objet permettant de développer des applications sous Windows. Il représente la suite logique de la famille turbo Pascal avec ses nombreuses versions (précisément le pascal objet).

Delphi est un outil moderne, puissant, faisant appel à une conception visuelle des applications, à la programmation orientée objet, à une bibliothèque de composants très riche (la VCL: Visual Components Library), aux fichiers DLL (Dynamic Link Library) et API (Application Programming Interface) de Windows; Delphi se classe comme l'un des meilleurs environnements de développement rapide des applications (RAD) dans le monde informatique.

Delphi, c'est aussi un produit qui évolue avec le temps. Chaque nouvelle version possède son lot de nouveautés. Voici les versions successives :

- Delphi 1 est le premier environnement pascal RAD sous Windows 3.11 (visuel).
- Delphi 2 marque le passage au mode 32 bits avec Windows 95.
- Delphi 3 supporte déjà les Active X, ainsi que l'aide à la saisie dynamique.
- Delphi 4 propose les tableaux dynamiques.
- Delphi 5 ouvre l'accès natif aux bases de données ADO et interbase.
- Delphi 6 introduit dbExpress et les composants portables de la CLX.

Et en fin, Delphi 7, Delphi 8 et récemment Delphi 2006 et 2007 qui sont sans doute les versions les plus enrichies de toute l'histoire du produit.



**Figure V.2:** l'interface Delphi

## I.2.2 Les avantages de Delphi

L'environnement DELPHI a plusieurs avantages dont on peut citer :

- Il est bien structuré, d'une difficulté moyenne et il donne des applications rapides ;
- Il est disponibles en plusieurs version : version 1 pour windows 3.x, en version 2 et 3 pour Windows 95 et NT ;
- Delphi est un environnement de développement de type rad (rapide application développement) basé sur le langage pascal. Il permet de réaliser rapidement et simplement des applications Windows.
- Delphi propose un ensemble très complet de composants visuels prêts à l'emploi incluant la quasi-totalité des composants Windows (boutons, boîtes de dialogue, menus, barres d'outils...) permettant de créer facilement divers types d'applications et de bibliothèques ;
- Delphi est un outil moderne, qui fait appel à une conception objet.
- Il prend en charge le maintien automatique d'une partie du code source.

- Il permet de créer facilement de nouveaux composants qui peuvent être intégrés dans la palette des composants déjà existants ;
- Son compilateur intégré permet une application rapide et efficace car les erreurs éventuelles du code sont immédiatement détectées. L'utilisateur est alors informé précisément des erreurs de son programme ;
- Il permet également d'utiliser des formats images, textes, sons, grâce à certains composants ;
- Delphi n'est pas lié à un format de données spécifiques. Il peut en effet utiliser des tables, dbase, access, ou paradox et accéder à des bases de données SQL serveur à travers un odbc(open data base connectivity).
- Il faut mentionner que la version utilisé pour le développement de ce projet est DELPHI 7.

### I.3 Implémentation de la base de données sous Delphi

Delphi contient un composant permettant de créer une base de donnée : Module Base de donnée (DBD32.exe) vous permet de créer, de voir et de modifier les tables des bases de données Paradox et dBase dans divers formats.

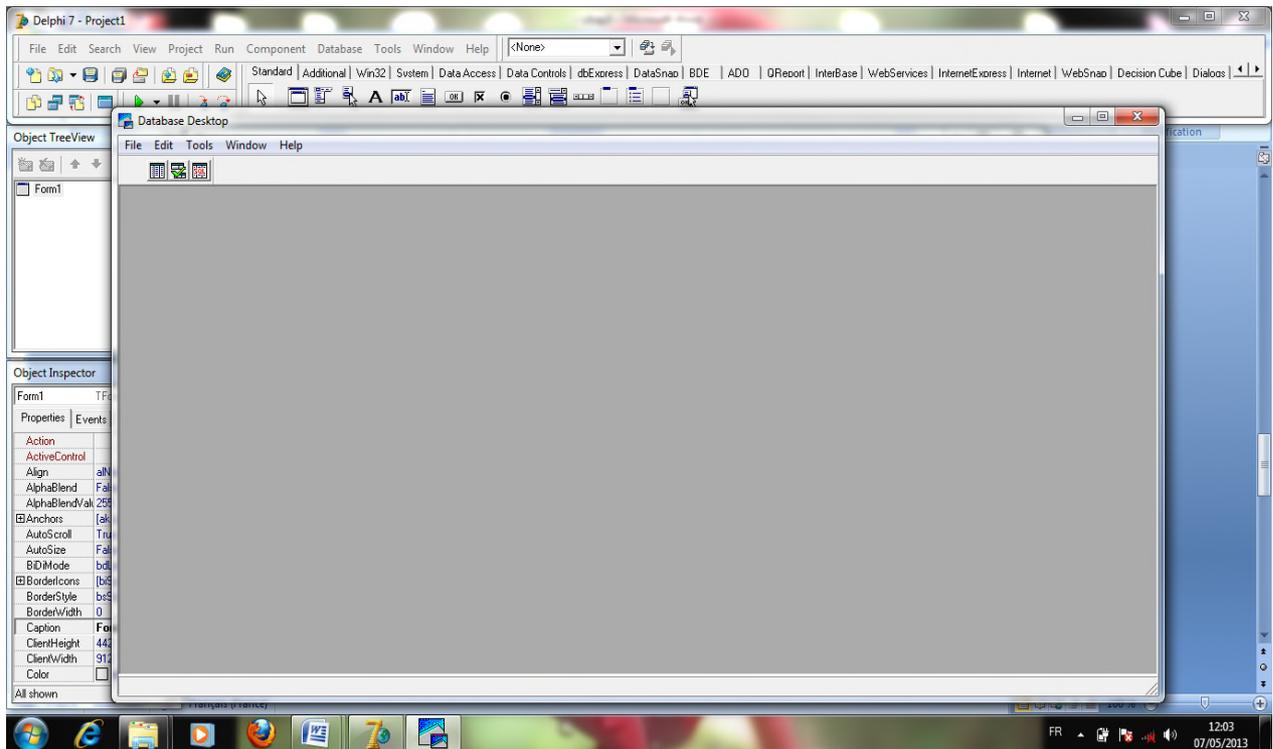


Figure V.3 : Module Base de donnée (Database Desktop).

## II. Le passage du diagramme de classe au modèle relationnel

Nous donnons ci-après quatre règles (de R1 à R4) pour traduire un schéma conceptuel entité association ou UML en un schéma relationnel équivalent. Il existe d'autres solutions de transformation mais ces règles sont les plus simples et les plus opérationnelles :

### II.1 Transformation des entités/ classes :

La règle est simple R1 : Chaque entité devient une relation, identifiant de l'entité devient clé primaire pour la relation. Chaque classe du diagramme UML devient une relation. Il faut choisir les attributs de la classe pouvant jouer le rôle d'identifiant.

### II.2 Transformation des associations

Les règles de transformation que nous allons voir dépendent des cardinalités / multiplicités maximale des associations. Nous distinguons trois familles d'association :

a- **Association 1..\*** : La règle est la suivante :

R2 : Il faut ajouter un attribut de type clé étrangère dans la relation fils de l'association. L'attribut porte le nom de la clé primaire de la relation père de l'association.

b- **Association \*.\*** : La règle est la suivante :

R3 : L'association / classe- association devient une relation. La clé primaire de cette relation est la concaténation des identifiants des identités connectées à l'association. Chaque attribut devient clé étrangère si l'entité / classe connectée dont il devient une relation en vertu de la règle R1. Les attributs de l'association / classe- association doivent être ajoutés à la nouvelle relation. Ces attributs ne sont ni clé primaire, ni clé étrangère.

c - **Association 1..1** : La règle est la suivante :

R4 : Il faut ajouter un attribut de type clé étrangère dans la relation dérivée de l'entité ayant la cardinalité minimal égale à zéro. Dans le cas du diagramme UML il faut ajouter un attribut de type clé étrangère dans la relation dérivée de la classe ayant la multiplicité minimale égale à un. L'attribut porte le nom de la clé primaire de la relation dérivée de l'entité / classe connectée à l'association. Si les deux cardinalités minimales égale à zéro, le choix est donné entre les deux relations dérivées de la R1. Si les deux cardinalités minimales égale à un, il est préférable de fusionner les deux entités / classe en une seule.

## II-3 Les relations de la base de données

Authentification (Password, Login).

Produit (**Codep**, Designation, Type, Quantité, Quantiteseuil, Datefab, Dateexp, unite, # Login).

Enseignant (**CodeEn**, NomEn, PrenomEn, AdrEn, NumtelEn).

Fournisseur (**Codef**, Nomf, Prenomf, Adrf, Aumtelf).

gérant (**Login**, Password, Nomg, prenomg, Numtelg, Adrg).

Bon de sortie (**Nbons**, Dates, Qtéf, Observations).

Bon de commande (**Nbc**, Datec, Qtéc, Observationc).

Bon de commande interne (**Nbci**, Datei, Qtéci).

Contient (**Nbons**, **Codep**, Qtéf ).

Existe (**Nbci**, **Codep**, Qtéci).

Livré (**Codep**, **Codef**, observation).

Appartient(**Codep**, **Nbci**, dateA ).

Ramène (**Nbc**, **CodeEn**, dateR).

## III. Description de l'implémentation du système

Le contenu de cette partie est une présentation qui définit la manière d'utilisation de notre système, enrichie par quelques formulaires de l'application réalisée.

The screenshot shows a web browser window titled 'Form1' with the following elements:

- Title:** laboratoire de produit chimique
- Login:** A text input field containing the name 'amel'.
- Password:** A text input field with masked characters 'xxxxxx'.
- Buttons:** An 'Ok' button with a green circular icon and an 'exit' button with a red power icon.
- Decorative Elements:** A magnifying glass on the left and a yellow smiley face keychain on the right.

Figure V.4: Formulaire s'authentifier

The screenshot shows a web browser window titled 'Form2' with the following elements:

- Header:** A banner image showing various laboratory glassware and chemicals.
- Navigation:** A 'produit' button with a multi-colored bar icon.
- Title:** Laboratoire de produit chimique du Centre Universitaire De Mila
- Buttons:** An 'exit' button with a red power icon.
- Main Content:** Two large images of laboratory glassware. The left image shows three flasks with red, yellow, and green liquids. The right image shows three flasks with red liquid on a red background.
- Footer:** Two buttons labeled 'Enseignant' and 'Fournisseur'.

Figure V.5: Formulaire menu principal

Codep	Design
S2	s
S3	Q
S4	a
S5	a

Figure V.6: Formulaire ajouter produit

Codep	Design
S2	s
S3	Q
S4	a
S5	a

Figure V.7: Formulaire supprimer produit.



Figure V.8: Formulaire modifier produit.



Figure V.9: Formulaire consulter produit.



Figure V.10: Formulaire extraire produit périmé.



Figure V.11: Formulaire extraire produit seuil alerte.



Figure V.12: Formulaire établir bon sortie.

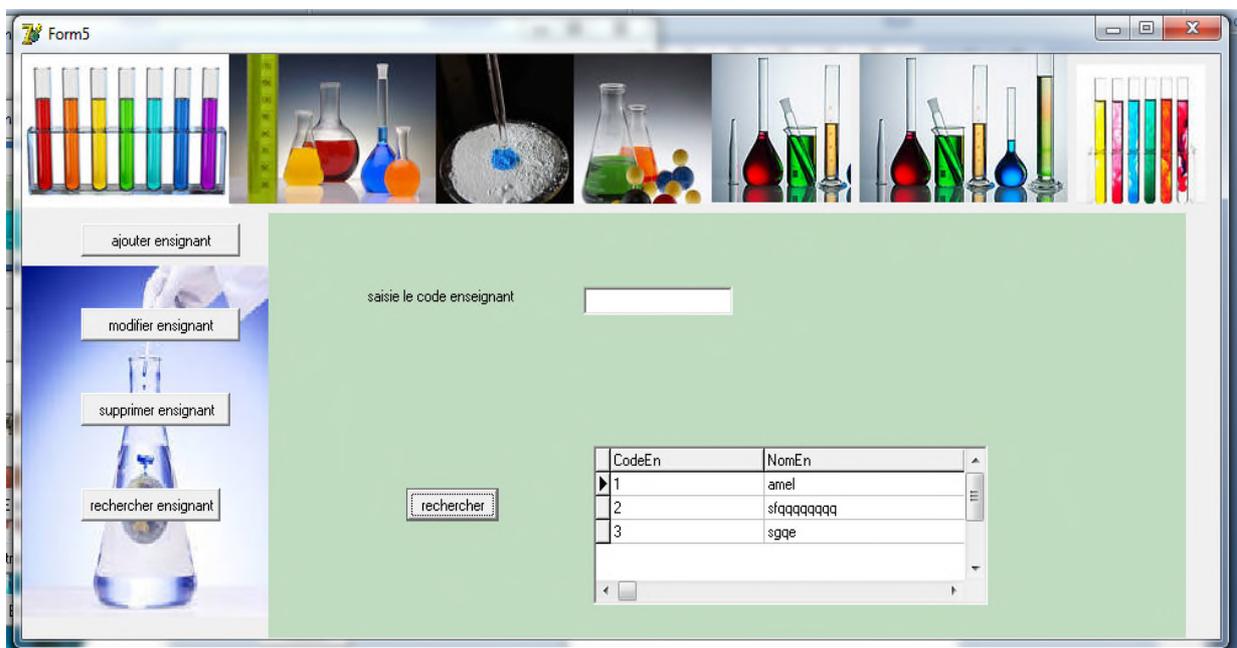
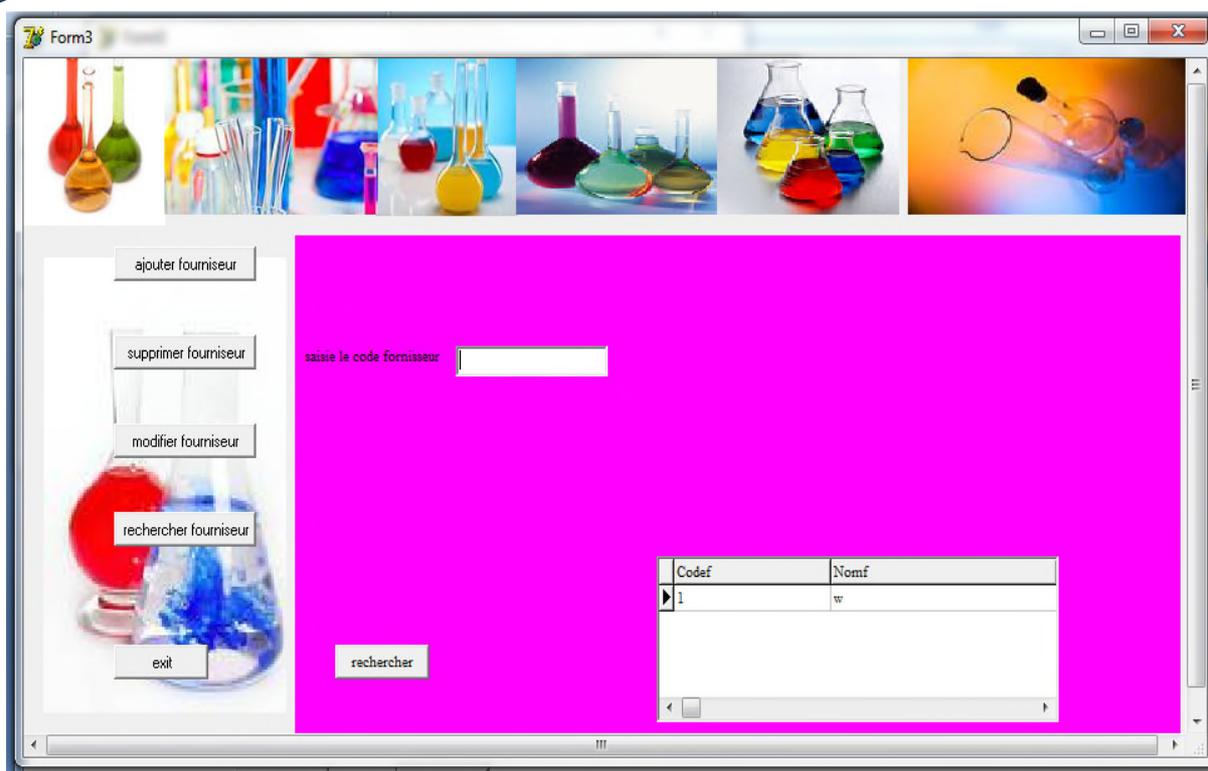


Figure V.13: Formulaire enseignant.



**Figure V.14:** Formulaire fournisseur.

## Conclusion

Dans ce chapitre on a défini les langages de programmation et les outils de travail que nous avons utilisé pour implémenter et réaliser notre système.

## ***CONCLUION GE'NE'RAL***

L'objectif de notre stage au sein du laboratoire de centre universitaire de Mila était la réalisation d'un système pour la gestion des produits chimiques.

Le logiciel développé est opérationnel pour effectu  les op rations de gestion de mani re automatique et facile et qui offre les services suivants :

- ✓ Gestion des produits: Ajout, Modification, Suppression, Consultation.
- ✓ Extraire les produits en seuil d'alerte.
- ✓ Extraire les produits p rim s
- ✓ Traitement des bons de commande et de sortie.

Pour r aliser ce syst me, nous avons utilis  le langage de mod lisation UML, le processus de d veloppement UP et d'autres outils de d veloppement comme : Delphi7, Pamestar UML.....

En r alisant ce projet, nous avons acquit de nouvelles connaissances comme nous avons appliqu  les connaissances que nous avons appris durant notre cursus universitaire.

## *Liste des figures*

<b>Chapitre</b>	<b>figure</b>	<b>Page</b>
Chapitre I	<b>Figure I.1</b> : Représentation d'acteur.	3
	<b>Figure I.2</b> : Représentation d'un cas d'utilisation.	4
	<b>Figure I.3</b> : Diagramme de cas d'utilisation	4
	<b>Figure I.4</b> : Relation d'inclusion.	4
	<b>Figure I.5</b> : Relation d'extention.	5
	<b>Figure I.6</b> : Relation de généralisation.	5
	<b>Figure I.7</b> : Représentation de classe.	6
	<b>Figure I.8</b> : Relation d'association.	6
	<b>Figure I.9</b> : Représentation des classes association.	7
	<b>Figure I.10</b> : Relation d'agrégation.	7
	<b>Figure I.11</b> : Relation de composition.	7
	<b>Figure I.12</b> : Diagramme de séquence.	8
	<b>Figure I.13</b> : Les lignes de vie.	8
	<b>Figure I.14</b> : Message synchrone.	9
	<b>Figure I.15</b> : Message asynchrones.	9
	<b>Figure I.16</b> : Message de création.	9
	<b>Figure I.17</b> : Quelle méthode pour passer de l'expression des besoins au code.	11
	<b>Figure I.18</b> : UP est itératif et incrémental.	13
	<b>Figure I.19</b> : UP est centré sur l'architecture.	14
Chapitre II	<b>Figure II.1</b> : Organigramme du centre universitaire de Mila.	18
	<b>Figure II.2</b> : Placard d'installation de produits chimiques.	22
Chapitre III	<b>Figure III.1</b> : Diagramme de contexte	31
Chapitre IV	<b>Figure IV.1</b> : Diagramme de cas d'utilisation	32
	<b>Figure IV.2</b> : Diagramme d'activité « S'authentifier »	41
	<b>Figure IV.3</b> : Diagramme d'activité « Ajouter produit »	42
	<b>Figure IV.4</b> : Diagramme d'activité « Modifier produit »	43
	<b>Figure IV.5</b> : Diagramme d'activité « supprimer produit»	44

	<b>Figure IV.6:</b> Diagramme d'activité «extraire produit seuil alerte»	45
	<b>Figure IV-7:</b> Diagramme d'activité «Extraire produit périmé»	46
	<b>Figure IV.8:</b> Diagramme d'activité «Etablir bon de sortie»	47
	<b>Figure IV.9:</b> Diagramme d'activité «Consulte produits»	48
	<b>Figure IV.2:</b> Diagramme de séquences « S'authentifier »	49
	<b>Figure IV.11:</b> Diagramme de séquences «Ajouter produit».	50
	<b>Figure IV.12:</b> Diagramme de séquences « Modifie produit »	51
	<b>Figure IV.13:</b> Diagramme de séquences «supprimer produit»	52
	<b>Figure IV.14:</b> Diagramme de séquences «extraire produit seuil alerte»	53
	<b>Figure IV.15:</b> Diagramme de séquences «Extraire produit périmé»	54
	<b>Figure IV.16:</b> Diagramme de séquences «Etablir bon de sortie»	55
	<b>Figure IV.17:</b> Diagramme de séquences «Consulter produits»	56
	<b>Figure IV.18:</b> Diagramme de classes	58
Chapitre V	<b>Figure V .1:</b> Pcestar UML	59
	<b>Figure V.2:</b> l'interface Delphi	61
	<b>Figure V.3 :</b> Module Base de donnée (Database Desktop)	61
	<b>Figure V.4:</b> Formulaire s'authentifier	65
	<b>Figure V.5:</b> Formulaire menu principal	65
	<b>Figure V.6:</b> Formulaire ajouter produit	66
	<b>Figure V.7:</b> Formulaire supprimer produit	66
	<b>Figure V.8:</b> Formulaire modifier produit	67
	<b>Figure V.9:</b> Formulaire consulter produit	67
	<b>Figure V.10:</b> Formulaire extraire produit périmé	68
	<b>Figure V.11:</b> Formulaire extraire produit seuil alerte	68
	<b>Figure V.12:</b> Formulaire établir bon sortie	69
	<b>Figure V.13:</b> Formulaire enseignant	69
	<b>Figure V.14:</b> Formulaire fournisseur	70

## *Liste des tableaux*

<b>chapitre</b>	<b>tableau</b>	<b>page</b>
Chapitre II	<b>Tableau II.1:</b> de symbole de danger d'étiquetage de produit chimique dans le centre universitaire de Mila.	21
	<b>Tableau II.2 :</b> Fiche d'étude du poste de travail gérant.	23
	<b>Tableau II.3:</b> les documents utilisés par le gérant dans le centre universitaire de Mila.	25

# Bibliographie

## ***1-Livre:***

1. **UML** en action de pascal roque & Franck vallée.
2. **ULM** de DI GALLO Frédéric.
3. **UML2** modéliser une application web.
4. **Livre** delphi.7\_Ebooks-land.net .

## ***2-Sites :***

5. [www.mémoireOnling.Com](http://www.mémoireOnling.Com)

## ***3-Travaux universitaire(Mémoires) :***

6. **Mémoire** « Conception et réalisation d'un système pour la gestion du matériel électrique et gazier au sein de la CAMEG de Mila (Filiale du groupe Sonelgaz).

