

LIVRABLE 2 : CONCEPTION ET ENVIRONNEMENT DE TRAVAIL



BLOC 2 : conception et programmation objet

CESI école d'ingénierie Angoulême

CPI A2 INFORMATIQUE 2022/2023

25/11/2022

Groupe projet N°2 :

Guillaume HOUNKPATI

Youssef AMALLAH

Asna YEWOU KOM

Abderrahmane FANIDI

Table des matières

I-GLOSSAIRE :	3
III-ANALYSE DU PROJET :	4
1-CONTEXTE :	4
2-PROBLEMATIQUE :	4
3-OBJECTIF :	4
IV-REALISATION :	4
1-PRESENTATION GENERALE DU SYSTEME :	4
2- MODELISATION :	5
2-1. MERISE :	5
a. Le Dictionnaire de données :	5
b. Modèle conceptuel de données (MCD) :	6
c. Modèle logique de données (MLD) :	8
2-2. DIAGRAMMES UML :	9
a. Diagramme de cas d'utilisation :	9
b. Diagramme de classes :	11
L'Objectif de ce diagramme de classe est de nous permettre de visualiser la structure de notre application.....	11
c. Diagramme de séquences :	11
d. Diagramme d'activités :	15
V – Résultat :	16
VI – Conclusion :	16
VII-BIBLIGRAPHIE :	16

I-GLOSSAIRE :

UML : (Unified Modeling Language) : c'est le langage de modélisation objet unifié qui décrit la limite, la structure et le comportement du système et des objets.

IHM : signifie interface homme-machine et fait référence à un tableau de bord qui permet à un utilisateur de communiquer avec une machine, un programme informatique ou un système.

Diagramme : Représentation graphique du déroulement et des variations (d'un phénomène).

Architecture client-serveur : représente l'environnement dans lequel des applications de machines clientes communiquent avec des applications de machines de type serveurs.

Système d'information : est un ensemble organisé de ressources qui permet de collecter, stocker, traiter et distribuer de l'information Montant total HT.

II-Introduction :

Dans ce projet nous sommes une start-up informatique qui avons été sollicitée par une entreprise pour l'aider à mieux organiser ses données. Cette entreprise a pour activité principale la vente en ligne de commandes électroniques. En effet, au vu des difficultés rencontrées lors de la gestion des données de cette entreprise, nous avons décidé de mettre sur pieds une architecture de type client-serveur composée d'une application et d'une base de données qui gèrent les données entrantes et sortantes de l'entreprise. Ainsi pour mener à bien ce projet nous allons :

D'une part analyser le système existant et le fonctionnement global de notre entreprise (étape actuelle du projet que nous détaillerons par la suite) et d'autre part apporter des solutions informatiques au problème rencontré(maquette).

III-ANALYSE DU PROJET :

1-CONTEXTE :

Pour remédier au problème de gestion de données auquel cette entreprise fait face nous avons décidé de concevoir une application qui nous permettra de gérer les différentes opérations réalisées au sein de cette entreprise.

2-PROBLEMATIQUE :

La problématique de notre projet est la suivante : Comment gérer les données entrantes et sortantes de cette entreprise ?

3-OBJECTIF :

L'objectif de notre projet est de réaliser une architecture de type client-serveur composée d'une application et d'une base de données.

IV-REALISATION :

1-PRESENTATION GENERALE DU SYSTEME :

Notre client attend de cette application qu'elle réalise les opérations suivantes :

➤ **Gestion du personnel :**

Cette partie de l'application doit pouvoir recenser l'effectif de l'entreprise, donner les caractéristiques de chaque personnel de l'entreprise (Nom et Prénom) et la date à laquelle il a été embauché.

➤ **Gestion des clients :**

Cette partie de l'application doit pouvoir recenser l'ensemble des clients de l'entreprise et donner leurs caractéristiques à savoir leur nom, prénom, adresse, leur date de naissance et la date de leur premier achat afin de bénéficier d'une réduction le jour de leur anniversaire et le jour de leur date d'anniversaire de leur premier achat.

➤ **Gestion des commandes :**

Cette partie de notre application doit pouvoir recenser l'ensemble des commandes de l'entreprise. Il faut noter que chaque commande est caractérisée par une référence composée des deux premières lettres du prénom du client, des deux premières lettres du nom du client, de l'année de la commande, des trois premières lettres de la ville de livraison

et d'un numéro incrémentiel. En effet l'entreprise veut aussi avoir dans ses archives la date d'émission d'une commande, la ou les dates de paiement et le moyen de paiement utilisé.

➤ **Gestion des stocks :**

Cette partie de notre application doit pouvoir recenser l'intégralité des produits à destination du client. Ainsi nous devons gérer le prix du produit HT, sa référence, sa désignation, sa quantité en stock, son seuil de réapprovisionnement et son taux de TVA. Il est important de noter que pour la grande majorité des produits le prix est fixé à l'instant T et pour une minorité le prix est variable en fonction de la quantité commandée par le client, de la nature de l'article et de sa couleur.

➤ **Gestion des statistiques :**

Cette partie de notre application doit pouvoir calculer le panier moyen (après remise), calculer le chiffre d'affaires sur un mois, identifier les produits sous le seuil de réapprovisionnement, calculer le montant total des achats d'un client, identifier les 10 articles les plus vendus, les 10 articles les moins vendus, calculer la valeur commerciale du stock, la valeur d'achat du stock et enfin simuler les variations de valeurs commerciales pour en déduire la valeur du stock.

2- MODELISATION :

- Le fonctionnement d'un système est l'ensembles des fonctionnalités de ce système. Nous avons représenté ces fonctionnalités à travers les diagrammes du langage UML (Unified Modeling Language).

2-1. MERISE :

a. Le Dictionnaire de données :

- Un dictionnaire des données est une collection de données de référence nécessaire à la conception d'une base de données relationnelles.

ENTITE	CODE	SIGNIFICATION	TYPE	OBSERVATIONS
Personnel	IdPersonnel	Identifiant personnel	Numérique	
	nomP	Nom personnel	Alphanumérique (50)	
	prenomP	Prénom personnel	Alphanumérique (50)	
	adresseP	Adresse personnel	Alphanumérique (50)	
	DateEmbauche	Date d'embauche	Date	
	sup	Supérieur personnel	Alphanumérique (50)	
Client	NumClient	Numéro client	Numérique	
	nomC	Nom client	Alphanumérique (50)	
	prenomC	Prénom client	Alphanumérique (50)	
	dateNaissance	Date de naissance	Date	

	nbrCommandes	Nombre commandes	Numérique	
Commande	IdCommande	Identifiant commande	Numérique	Référence : deux premières lettres du prénom du client, deux premières lettres du nom du client, de l'année de la commande, de trois premières lettres de la ville de livraison et d'un numéro incrémentiel
	reference	Référence commande	Alphanumérique (50)	
	dateLivraison	Date livraison	Date	
	dateEmission	Date d'envoi	Date	
	datePaiement	Date de paiement	Date	
	moyenPaiement	Moyen de paiement	Alphanumérique (50)	
	totalArticles	Total des articles	Numérique	
	montantHT	Montant HT	Numérique	
	montantTVA	Montant TVA	Numérique	
	montantTTC	Montant TTC	Numérique	
Produit	IdProduit	Identifiant produit	Numérique	
	refProduit	Référence produit	Alphanumérique (50)	
	prixHT	Prix hors taxe	Numérique	
	Désignation	Désignation produit	Alphanumérique (50)	
	qteStock	Quantité stock	Numérique	
	seuilAppr	Seuil d'approvisionnement	Numérique	
	tauxTVA	Taux TVA	Numérique	
Adresse	idAdresse	Identifiant adresse	Numérique	
	Ville	Ville de l'adresse	Alphanumérique (20)	
	codePostal	Code postal adresse	Numérique	
	Rue	Rue de l'adresse	Alphanumérique (20)	
Facture	idFacture	Identifiant facture	Numérique	
	nomSociete	Nom de la société	Alphanumérique (50)	
	adrSociete	Adresse de la société	Alphanumérique (50)	
	NumSerClient	Numéro service client	Alphanumérique (20)	
	Logo	Logo	Alphanumérique (50)	

Figure1 : dictionnaire de données

b. Modèle conceptuel de données (MCD) :

- Le modèle conceptuel de données (MCD) a pour but d'écrire de façon formelle les données qui seront utilisées par le système d'information. Il s'agit donc d'une

représentation des données, facilement compréhensible, permettant de décrire le système d'information à l'aide d'entités.

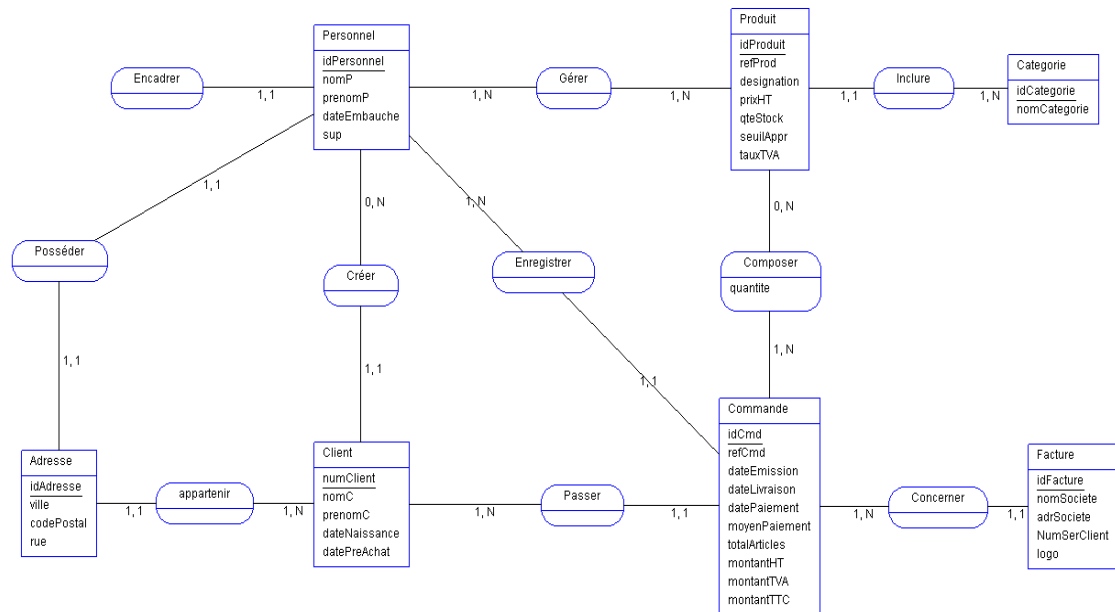


Figure 2 : MCD

RELATION ENTRE PERSONNEL ET COMMANDE

- Un personnel peut enregistrer une ou plusieurs commandes donc l'association possède la cardinalité one to many (1, N) du côté de personnel.
- Une commande peut être enregistrée par une seule personne donc l'association possède la cardinalité one to one (1,1) du côté de commande.

RELATION ENTRE CLIENT ET COMMANDE

- Un client peut effectuer une ou plusieurs commandes donc l'association possède la cardinalité one to many (1, N) du côté de client.
- Une commande peut être effectuée par un et un seul client donc l'association possède la cardinalité one to one (1,1) du côté de commande.

RELATION ENTRE PRODUIT ET COMMANDE

- Une commande est composée d'un ou plusieurs produits donc l'association possède la cardinalité one to many (1, N) du coté de commande.
- Un produit compose 0 ou plusieurs commandes donc l'association possède la cardinalité zéro to many (0, N) du coté de produit.

RELATION ENTRE PERSONNEL ET PRODUIT

- Un personnel peut gérer un ou plusieurs produits donc l'association possède la cardinalité one to many (1, N) du côté de personnel.
- Un produit peut être géré par une ou plusieurs personnes donc l'association possède la cardinalité one to many (1, N) du côté de produit.

RELATION ENTRE PERSONNEL ET CLIENT

- Un personnel de l'entreprise peut ne pas servir de clients ou en servir plusieurs donc l'association possède la cardinalité zéro to many (0, N) du côté de personnel.
- Un client peut être servi par un ou plusieurs personnel(S) de l'entreprise donc l'association possède la cardinalité one to many du côté du client.

RELATION ENTRE ADRESSE ET CLIENT

- Une adresse peut appartenir à un et un seul client donc l'association possède la cardinalité one to one (1,1) du côté d'adresse.
- A un client peut appartenir une ou plusieurs adresses donc l'association possède la cardinalité one to many du côté de client.

RELATION ENTRE PRODUIT ET CATEGORIE

- Un produit est inclus dans une et une seule catégorie donc l'association possède la cardinalité one to one du côté de produit.
- Une catégorie inclut un ou plusieurs produits donc l'association possède la cardinalité one to many du côté de catégorie.

RELATION ENTRE PERSONNEL ET ADRESSE

- Un personnel possède une et une seule adresse donc l'association possède la cardinalité one to one du côté de personnel.
- Une adresse est possédée par une et une seule personne donc l'association possède la cardinalité one to one du côté d'adresse.

RELATION ENTRE PERSONNEL ET PERSONNEL

- Un personnel de l'entreprise peut être encadré par un et un seul autre personnel de l'entreprise qui est son supérieur hiérarchique.

c. Modèle logique de données (MLD) :

Un modèle logique de données (MLD) est la représentation des données d'un système d'information. Les données sont représentées en prenant en compte le modèle technologique qui sera utilisée pour leur gestion.

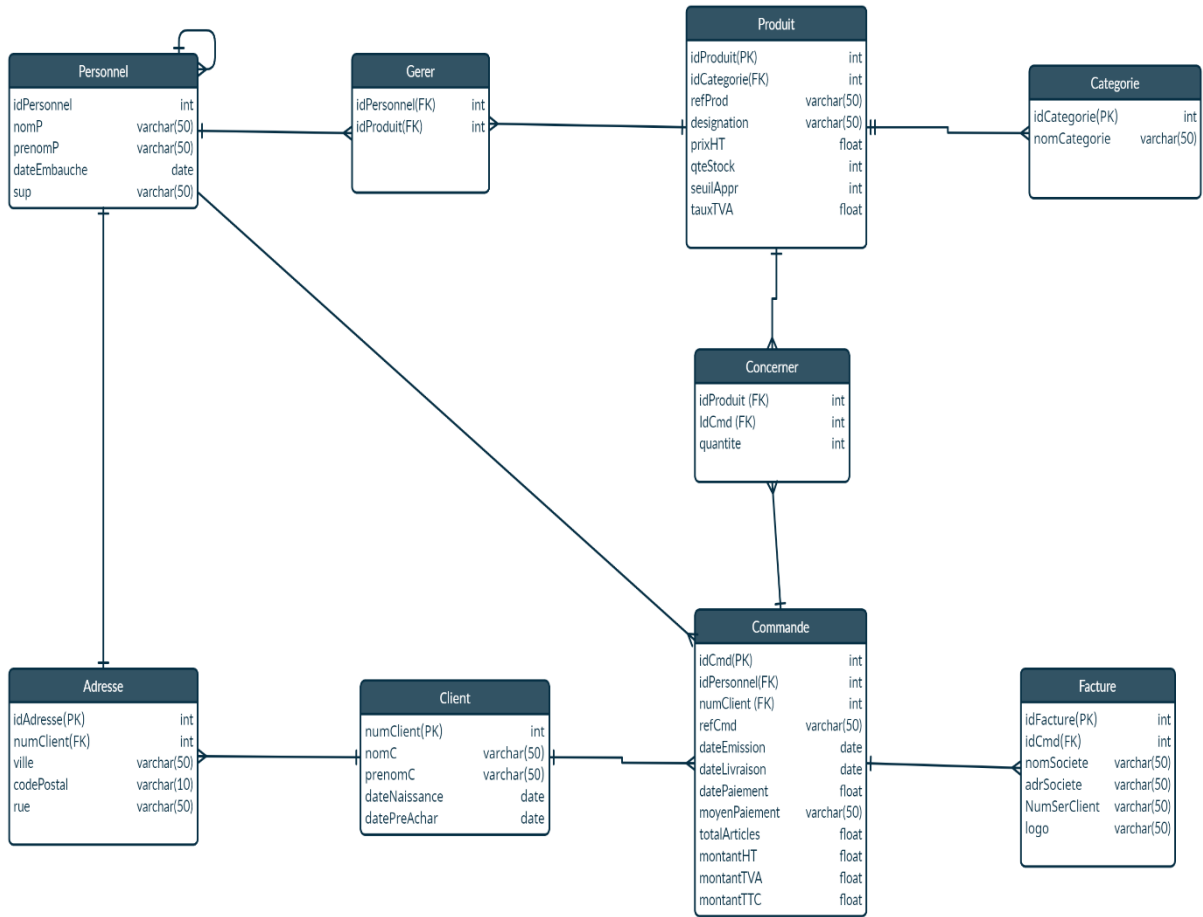


Figure 3 : MLD

2-2. DIAGRAMMES UML :

Le langage UML est langage de modélisation unifié destiné à la modélisation d'un système en orienté objet, Il est destiné aux phases antérieures à la réalisation d'un logiciel. Et on en dénombre quatre à savoir :

a. Diagramme de cas d'utilisation :

Dans notre système nous avons principalement deux acteurs à savoir :

Le personnel : qui est chargé de gérer les clients, les stocks, les commandes et les statistiques de l'entreprise.

Le superviseur : qui fait les taches du personnel en plus de la gestion du personnel.

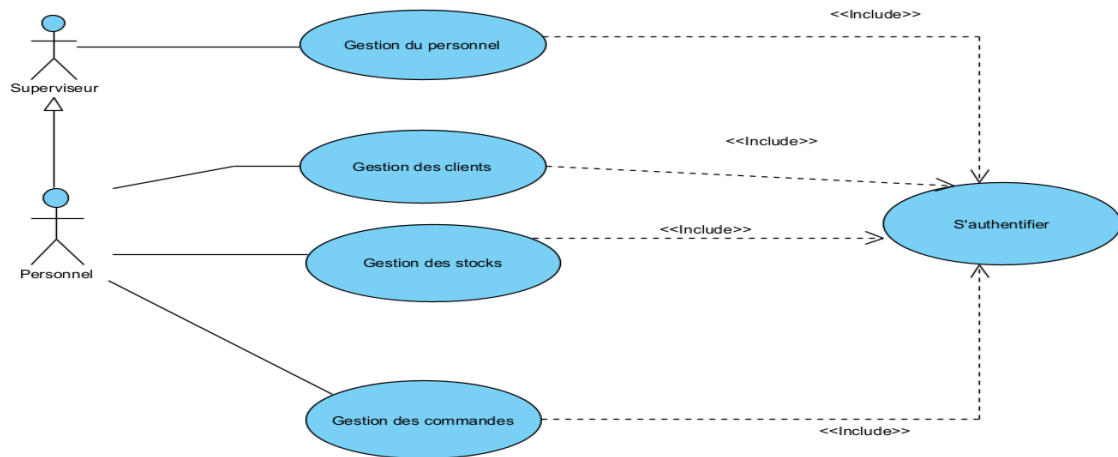


Figure 4 : Diagramme de cas d'utilisation général.

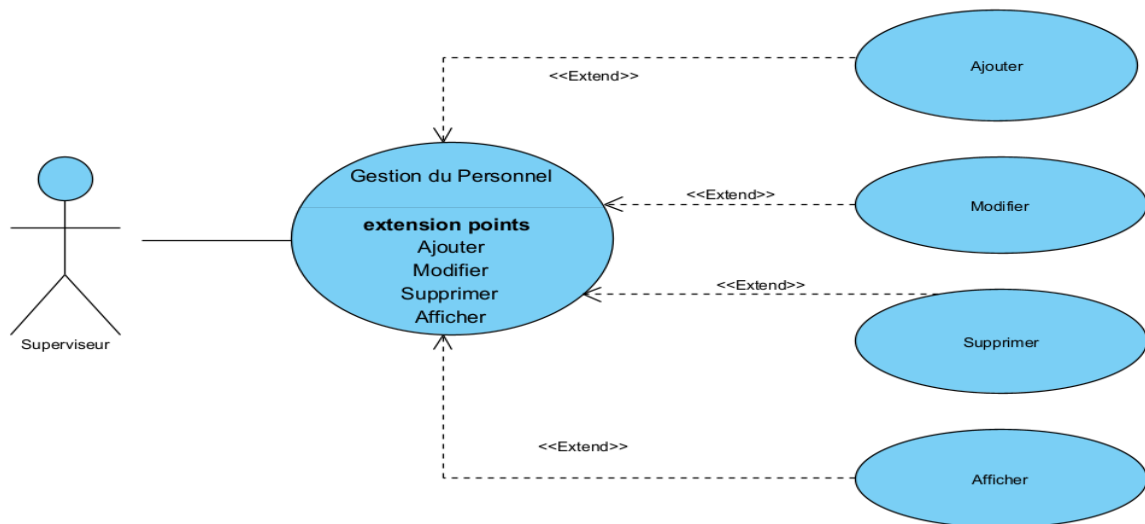


Figure 5 : Diagramme de cas d'utilisation gestion du personnel.

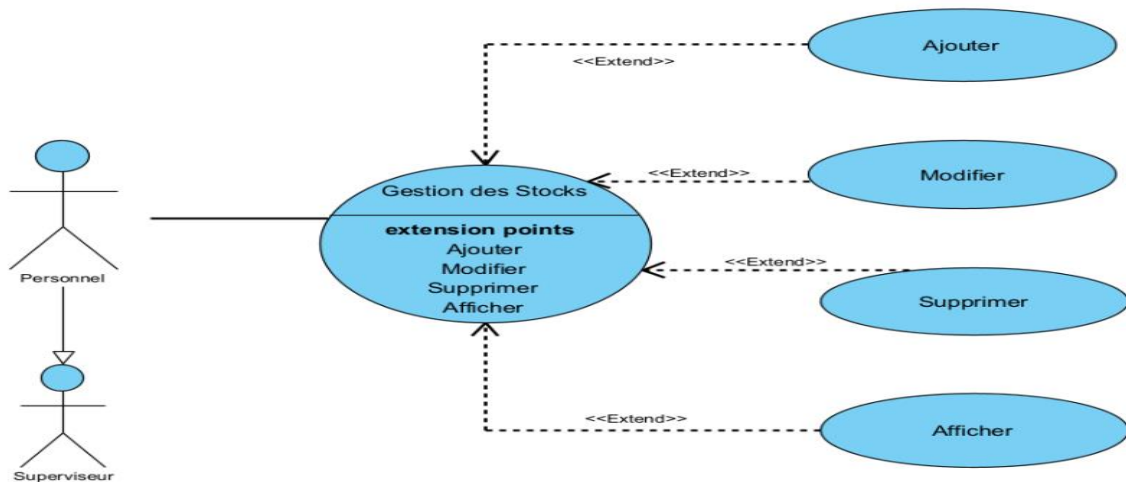


Figure 6 : Diagramme de cas d'utilisation gestion du stock.

b. Diagramme de classes :

Nous avons 6 classes qui composent notre système :

- Les classes Client et Personnel héritent de la classe Personne
- La classe Commande est une composition de la classe Produit d'où le lien d'agrégation
- La classe Statistique dépend des données des classes Commande et Produit

Nous avons des relations d'association entre les classes Client – Commande et les classes Personnel – Produit.

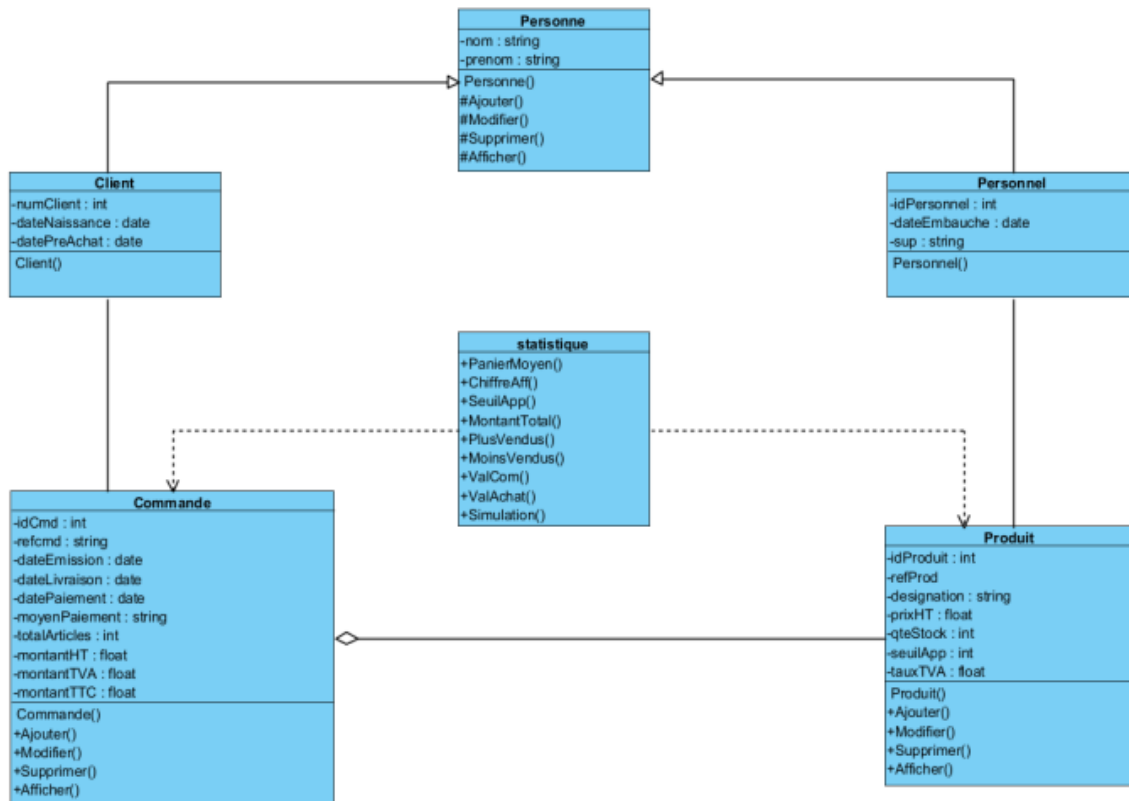


Figure 7 : Diagramme de classe

L'Objectif de ce diagramme de classe est de nous permettre de visualiser la structure de notre application.

c. Diagramme de séquences :

Créer un personnel :

Pour créer un personnel, L'acteur va demander au système d'ajouter un personnel et l'IHM va afficher le formulaire. Le supérieur hiérarchique va entrer les données du nouveau personnel que l'on veut ajouter et les enregistrer afin que l'IHM puisse les vérifier, après la vérification des informations par l'IHM les données sont envoyées dans la base de données qui a son tour va vérifier si celles-ci n'existent pas déjà.

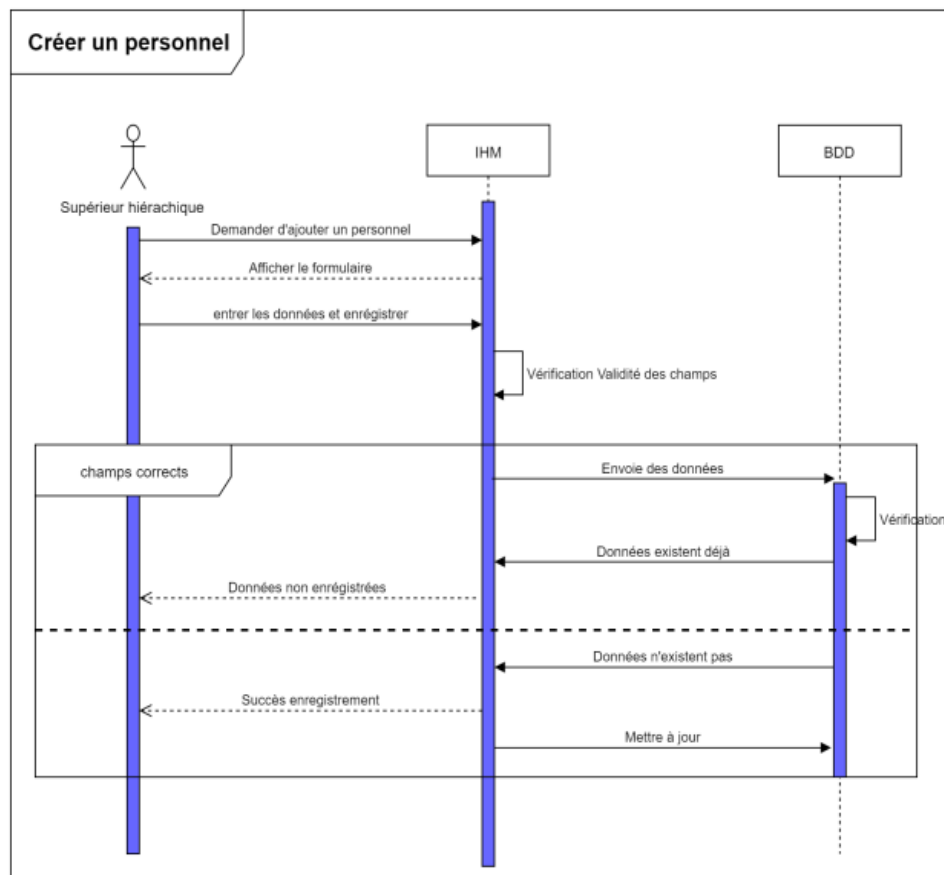


Figure 8 : Diagramme de séquence créer un personnel.

Modifier un personnel :

Pour modifier un personnel, L'acteur va demander la liste des personnels chez l'interface homme-machine, avec l'interaction entre IHM et base de données la liste des personnels chargée dans IHM, l'acteur sélectionner le personnel, puis le base de données fournit toutes les informations et afficher dans l'interface homme-machine, après le succès de la sélection personnel, l'acteur peut modifier les informations par intermédiaire de message de confirmation, Si la réponse oui le utilisateur modifier les informations de personnel, Si la réponse non ça veut dire que il y aucun de modification juste affichages les informations.

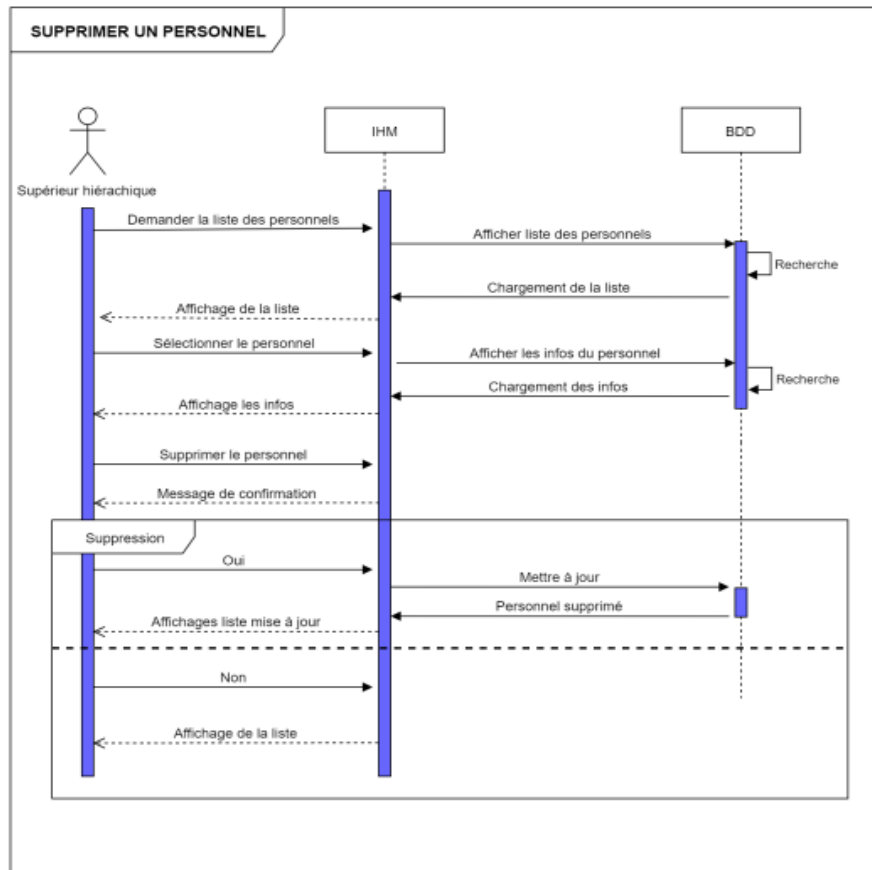


Figure 10 : Diagramme de séquence modifier un personnel.

Afficher un personnel :

- Pour afficher un personnel, L'acteur va demander la liste des personnels chez l'interface homme-machine, avec l'interaction entre IHM et base de données la liste des personnels chargée dans IHM, l'acteur sélectionner le personnel, puis le base de données fournit toutes les informations et afficher dans l'interface homme-machine.

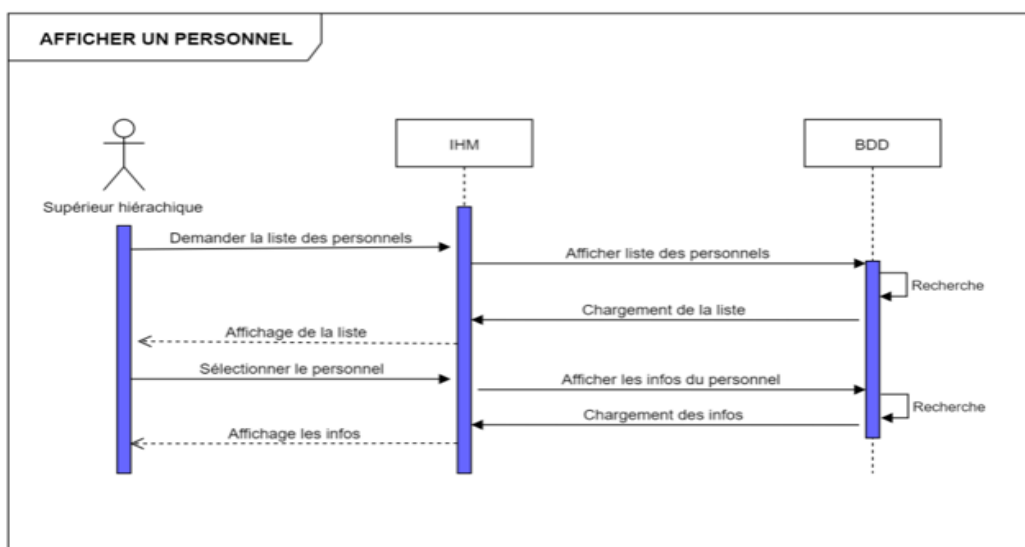


Figure 11 : diagramme de séquence afficher un personnel.

Gestion d'un personnel :

L'utilisateur qui va lancer le démarrage de l'application, saisir l'identifiant et le mot de passe, créer un personnel ou sélectionner le personnel qu'il veut supprimer, modifier ou afficher s'il existe déjà.

La BDD : qui vérifiera chaque opération et nous enverra soit un message d'erreur en cas d'erreur soit le résultat attendu et mettra à jour les informations avant de les enregistrer.



V – Résultat :

Nous avons donc réalisé les différents diagrammes et on a donné une explication pour chacun d'entre eux.

Nous avons donc fait :

- Le dictionnaire de données : est une collection de métadonnées ou de données de référence nécessaire à la conception d'une base.
- Le diagramme d'usage (ou USE-CASE) qui nous permet de d'identifier les possibilités d'interaction entre le système et les acteurs.
- Le diagramme de séquence qui permet de montrer les interactions d'objets/système dans le cadre d'un scénario d'usage.
- Le diagramme d'activité permettant de représenter le déclenchement d'événements en fonction des états du système.
- Le diagramme des classes qui nous a permis d'appréhender la structure de notre système existants et d'en concevoir de nouveaux.

VI – Conclusion :

Pendant ce livrable, nous avons réalisé différents diagrammes : un diagramme d'usage, de séquence, d'activité et de classe. Et aussi un MCD et MLD qui vont nous aider par la suite dans la réalisation de l'application (réalisation de la partie code). Dans l'ensemble, les objectifs que nous étions fixés dans ce livrable ont été atteints.

VII-BIBLIOGRAPHIE :

[https://fr.wikipedia.org/wiki/UML_\(informatique\)#:~:text=Le%20Langage%20de%20Mod%C3%A9lisation%20Unifi%C3%A9,et%20en%20conception%20orient%C3%A9%20objet](https://fr.wikipedia.org/wiki/UML_(informatique)#:~:text=Le%20Langage%20de%20Mod%C3%A9lisation%20Unifi%C3%A9,et%20en%20conception%20orient%C3%A9%20objet)

[https://fr.wikipedia.org/wiki/Merise_\(informatique\)](https://fr.wikipedia.org/wiki/Merise_(informatique))

<https://creately.com/blog/fr/diagrammes/tutoriel-sur-le-diagramme-de-sequence/>

<https://laurent-audibert.developpez.com/Cours-UML/?page=diagramme-activites>

https://www.univ-constantine2.dz/CoursOnLine/Benelhadj-Mohamed/co/grain3_2.html