



République Algérienne Démocratique et Populaire
Université Abou Bakr Belkaid– Tlemcen
Faculté des Sciences
Département d'Informatique

Mémoire de fin d'études

pour l'obtention du diplôme de Licence en Informatique

Thème

Application de gestion de magasin d'optique médicale (GMOM)

Réalisé par :

- Mégaiz Samia.

Présenté le 27 Juin 2013 devant la commission d'examination composée de MM.

- Mr Benamar Abdelkarim (Examineur)
- Mme Berramdane (Examineur)
- Mme Belhabi Amel (Examineur)

Sommaire :

Introduction générale	6
-----------------------------	---

Chapitre I : Etude conceptuelle des données

I . Introduction.....	9
II. Modélisation des données.....	10
II .1. Niveau conceptuel.....	10
II.1.a. Modèle conceptuel de données.....	10
II.1.b. Modèle conceptuel de traitements.....	16
II .2. Niveau logique/organisationnel.....	19
II.2.a. Modèle logique de données.....	19
II.2.b. Modèle organisationnel de traitement.....	20
II.3. Niveau physique/opérationnel.....	22
II.3.a. Modèle physique de données.....	22
II.3.b. Modèle opérationnel de traitements.....	23
III. Conclusion	23

Chapitre II : Réalisation du projet

I. Introduction.....	25
II. Introduction au SGBD Oracle.....	25
III. Introduction au Lazarus.....	26
IV. Les interfaces de l'application.....	27
IV.1 Principale.....	27

IV.2 Entrée.....	28
IV.3 Ajout	29
IV.4 Modification.....	32
IV.5 Recherche.....	32
IV.6 Stock de monture.....	32
V. Conclusion.....	34
Conclusion générale.....	36
Bibliographie et webgraphie	38

Liste des figures

Figure I.1 : le MCD.....	15
Figure I.2 : Synchronisation d'une opération dont la condition.....	16
Figure I.3 : Schéma d'une opération.....	17
Figure I.4 : le MCT.....	18
Figure I.5 : Le formalisme complet d'un MOT.....	20
Figure I.6 : le MOT.....	21
Figure II.1 : Principale.....	27
Figure II.2 Erreur user ou password.....	27
Figure II.3 : Fenetre d'entrée.....	28
Figure II.4 : Fenetre client.....	29
Figure II.5 : Fenetre correction.....	30
Figure II.6 : Fenetre message d'achat monture.....	30
Figure II.7 : Fenetre achat monture.....	31
Figure II.8 : Fenetre lentille.....	31
Figure II.9 : Fenetre Recherche.....	32
Figure II.10 : Fenetre stock.....	32
Figure II.11 : etat de stock.....	33
Figure II.12 : fenetre modification d'une monture.....	33
Figure II.13 : fenetred'ajout d'une monture.....	34

Liste des tables

Table I.1 : l'approche par niveau.....	9
Table I.2 : dictionnaire des données.....	12

Introduction

Générale :

Le monde connaît une avance technologique considérable dans tous les secteurs et cela grâce à l'informatique qui est une science qui étudie les techniques du traitement automatique de l'information ; elle joue un rôle important dans le développement de l'entreprise et d'autres établissements.

Avant l'invention de l'ordinateur on enregistrait toutes les informations manuellement sur des supports en papier ce qui engendrait des problèmes tels que la perte de temps considérable dans la recherche de ces informations ou la dégradation de ces dernières.

Ainsi jusqu'à présent l'ordinateur reste le moyen le plus sûr pour le traitement et la sauvegarde de l'information cette invention a permis d'informatiser les systèmes de données des entreprises cette machine devient un outil indispensable qui aide à la prise de décision et se trouve présent dans tous les domaines d'activité. Au début l'absence de conception globale a généralement induit à la mise en place d'applications répondant chacune à un besoin spécifique rapidement ; ces besoins se sont croisés, utilisant les mêmes données l'un déclenchant l'autre aboutissant à la fin à une incohérence et une redondance de données.

Les magasins font partie intégrante des entreprises commerciales que l'informatique pourra beaucoup aider.

L'objectif de mon projet présenté dans ce mémoire est la conception et la réalisation d'une application permettant la gestion de magasin d'optique médicale, se fait dans le but de faciliter aux opticiens de gérer leurs magasins et d'enregistrer les données des clients et aussi permettant la gestion de stock.

L'opticien-lunetier est un professionnel qui, sur prescription médicale, réalise, adapte et vend des articles destinés à corriger les défauts ou déficiences de la vue, et conseille les utilisateurs des matériels fournis. Il commercialise également des fournitures telles que lunettes de soleil, étuis, produits d'entretien, instruments d'optique et de mesure. Le caractère commercial de son métier est important. Il est chargé, entre autres, du contact avec la clientèle, de la gestion des stocks, de la définition de l'équipement adéquat, du management des salariés.

L'application de GMOM (Gestion de Magasin d'Optique Médicale) comporte des modules essentiels :

- ✚ Enregistrement d'un client : le client a son propre identifiant
- ✚ Recherche d'un client : en introduisant le nom du client
- ✚ Modification des données d'un client : la possibilité de changer toutes les données sauf l'identifiant
- ✚ Stockage des montures : l'ajout ou la suppression de la monture

Ce mémoire est organisé de la façon suivante :

- ✚ Après cette introduction générale, le premier chapitre consacré à la présentation de la méthode d'analyse Merise et la conception de la BDD

- ✚ Le deuxième chapitre concerne la réalisation d'application GMOM en définissant l'oracle SGBD pour la création de la BDD et Lazarus IDE

Chapitre I

Etude Conceptuelle De Données

I-Introduction :

La méthodologie permet de décrire, d'analyser, de spécifier et de concevoir d'une façon rigoureuse et normale les systèmes d'information existants.

Parmi les méthode de conception, nous avons opté pour la méthode MERISE « Méthode d'Etude de Réalisation Informatique par Sous Ensemble », elle est avant tout un état d'esprit, en effet, utiliser MERISE implique un investissement personnel qui comporte beaucoup de rigueur, aujourd'hui elle est largement utilisée dans les administrations et les entreprises de toute taille, le choix de cette méthode revient aux avantages qu'elle offre tout particulièrement. (1)

Cette approche vise à concevoir le futur système automatisé d'information de chaque domaine de l'entreprise en modélisant les données et les traitements sur trois niveaux :

- Le niveau **conceptuel**,
- Le niveau **logique** ou **organisationnel**,
- Le niveau **physique**.

Le principe de l'approche par niveau est basé sur les point suivant :

-chaque niveau répondra à des préoccupations bien précises

-les règles de fonctionnement du système d'information seront mises en évidence à travers ces trois niveaux

-à chaque niveaux correspondront deux modèles, un modèle pour les données et un autre pour les traitements. (2)

Les niveaux de la méthode Merise (2)

Niveau \ Aspect	Règles	Modèles	
		Données	traitement
Conceptuel	Gestion	Modèle Conceptuel de Données (MCD).	Modèle Conceptuel de Traitement (MCT).
Organisationnel/Logique	Organisation	Modèle Logique de Données (MLD).	Modèle Organisationnel de Traitement (MOT).
Opérationnel/Physique	Technique	Modèle Physique de Données (MPD).	Modèle Opérationnel de Traitement (MOpT).

Table I.1: l'approche par niveau (2)

II-Modélisation des données :

Dans la conception d'un **système d'information**, la modélisation des données est l'analyse et la conception de l'**information** contenue dans le système.

Il s'agit essentiellement d'identifier les entités logiques et les dépendances logiques entre ces entités. La modélisation des données est une représentation **abstraite**, dans le sens où les valeurs des **données** individuelles observées sont ignorées au profit de la structure, des relations, des noms et des formats des **données** pertinentes, même si une liste de valeurs valides est souvent enregistrée. Le **modèle de données** ne doit pas seulement définir la **structure de données**, mais aussi ce que les **données** veulent vraiment signifier (**sémantique**). (4)

II-1-Niveau conceptuel :

Au niveau conceptuel on veut décrire, après abstraction, le modèle (le système) de l'entreprise ou de l'organisme, on doit donc construire: (2)

II-1-a-Modèle Conceptuel de Données (MCD) :

Le modèle conceptuel de données représente la division statique du système d'information, il donne une représentation stable de l'ensemble des données manipulées par l'entreprise ainsi que les relations entre ces données. (2)

Avant la construction du MCD on doit définir les concepts de base sur les quel elle se base:

1. **Objet** (individu ou entité) : En d'autres termes, c'est la représentation dans le système d'information d'un objet matériel ou immatériel du réel perçu.
2. **Relation** (association) : Une relation entre entités (objets) est une association perçue dans le réel entre deux ou plusieurs entités. Une relation n'a pas d'existence propre.
3. **Propriété** (ou l'attribut) : Une propriété est une donnée (une information) élémentaire qui caractérise un objet ou une relation.
4. **Occurrence** :
 - L'occurrence d'un objet (ou entité) : est un élément individualisé appartenant à cet objet (ou entité).
 - L'occurrence d'une relation : est une relation individualisée constituées d'une et d'une seule occurrence de chacun des objets participant à la relation.
5. **Identifiant** :
 - Identifiant d'un objet : est une propriété particulière qui caractérise de façon unique chaque occurrence de cet objet.
 - *L'identifiant figure en première position

dans la liste des propriétés de l'objet.

*L'identifiant est souligné.

-Identifiant de la relation : c'est l'identifiant obtenu par concaténation des identifiants des objets participant à la relation.

6. Cardinalité : Les cardinalités d'un objet par rapport à une relation s'expriment par deux nombres appelés :

-Cardinalité minimale : (égale à 0 ou 1) C'est le nombre minimum de fois qu'une occurrence d'un objet participe aux occurrences de la relation.

-Cardinalité maximale : (égale à 1 ou n) C'est le nombre maximum de fois qu'une occurrence de l'objet participe aux occurrences de la relation.

7. Dimension d'une relation : La dimension d'une relation désigne le nombre d'objets participant à la relation.

Selon le nombre d'objets qui participent à la relation il existe :

-La relation entre deux objets : dite relation **binaire**.

- La relation entre trois objets : dite relation **ternaire**.

- La relation entre plusieurs objets (n) : dite **relation n-aire**.

- La relation entre un objet et lui-même : dite **relation réflexive** ou **uniare**.

La construction d'un modèle conceptuel de données permet au concepteur d'approfondir la connaissance du champ d'étude à travers son vocabulaire et ses entités, donc il suit ces étapes pour obtenir un MCD correct et exact : (2)

-élaboration de la liste des données : dictionnaire de données.

-recherche des individus et leurs identifiants avec rattachement des propriétés aux individus.

-recensement des relations avec l'étude des cardinalités de chaque couple individu-relation.

-schéma conceptuel de données descriptif (BRUT)

-l'application des règles de vérifications, normalisation et décomposition.

-le modèle conceptuel de données. (2)

Dictionnaire de données :

Le dictionnaire de données est un tableau qui regroupe toutes les données du système d'information. (2)

Pour chaque donnée il faut préciser :

- Sa désignation.
- Sa longueur(en caractère).
- Un code (attribué par le concepteur).

- Son type (Numérique, Alphabétique, Alphanumérique).
- Une observation si cela est nécessaire.

Légende : Nous conviendrons de ce qui suit pour désigner le type de la donnée :

- N : Numérique.
- A : Alphabétique.
- AN : Alphanumérique.

La date sera codifiée de la manière suivante :

- JJ : Deux positions numériques pour le jour.
- MM : Deux positions numériques pour le mois.
- AAAA : Quatre (ou deux) position pour l'année. (2)

Désignation de la donnée	Code attribué à la donnée	Type	Taille	Observation
Identifiant d'un client	id_cl	N	3	Identifiant de la table client.
Nom d'un client	nom_cl	A	20	
Prénom d'un client	pre_cl	A	20	
Sexe d'un client	sexe_cl	A	10	
Age d'un client	age_cl	N	2	
Identifiant de l'œil gauche	id_og	N	3	Identifiant de la table œil gauche
Puissance sphérique d'un verre de l'œil gauche	sph_og	AN	3	
Puissance cylindrique d'un verre de l'œil gauche	cyl_og	AN	3	
Axe d'un verre de l'œil gauche	axe_og	AN	3	
Vision de loin de l'œil gauche	v_l_og	AN	3	
Vision de près de l'œil gauche	v_p_og	AN		

Nature du verre de l'œil gauche	type_og	A	20	
Prix du verre de l'œil gauche	prix_og	A	6	NNNN,NN
Identifiant de l'œil droit	id_od	N	3	Identifiant de la table œil droit
Puissance sphérique d'un verre de l'œil droit	sph_od	AN	3	
Puissance cylindrique d'un verre de l'œil droit	cyl_od	AN	3	
Axe d'un verre de l'œil droit	axe_od	AN	3	
Vision de loin de l'œil droit	v_l_od	AN	3	
Vision de près de l'œil droit	v_p_od	AN		
Nature du verre de l'œil droit	type_od	A	20	
Prix du verre de l'œil droit	prix_od	A	6	NNNN,NN
Identifiant de lentille gauche	id_lg	N	3	Identifiant de la table lentille gauche
Puissance sphérique de lentille gauche	sph_lg	AN	3	
Puissance cylindrique de lentille gauche	cyl_lg	AN	3	
Axe de lentille gauche	axe_lg	AN	3	
Rayon de courbure de lentille gauche	RC_lg	AN	3	
Type de lentille gauche	type_lg	A	20	
Choix de lentille gauche	Choix_lg	A	20	
Nature du lentille de l'œil gauche	nature_lg	A	20	
Prix du lentille de l'œil gauche	prix_lg	A	6	NNNN,NN
Identifiant de lentille droite	id_ld	N	3	Identifiant de la table lentille droite

Puissance sphérique de lentille droite	sph_ld	AN	3	
Puissance cylindrique de lentille droite	cyl_ld	AN	3	
Axe de lentille droite	axe_ld	AN	3	
Rayon de courbure de lentille droite	RC_ld	AN	3	
Type de lentille droite	type_ld	A	20	
Choix de lentille droite	Choix_ld	A	20	
Nature du lentille de l'œil droite	nature_ld	A	20	
Identifiant de la monture	id_mont	N	3	Identifiant de la table monture
Type de la monture	Type_mont	A	20	
Categorie de la monture	Cat_mont	A	20	
Quantité des montures existantes	Qte_mont	N	3	
Prix de la monture	Prix_mont	N	6	NNNN,NN
Identifiant du facture	Id_fact	N	3	Identifiant de la table facture
Date de commande	date_co	Date	8	
Prix total	Prix_tot	N	6	NNNN,NN
Versement du client	Vers_cl	N	6	NNNN,NN
Reste à verser	Rest_a_vers	N	6	NNNN,NN

Table I.2 : dictionnaire de données

✚ Les règles de gestion :

Elles expriment les règles auxquelles obéit le système à modéliser. Elles permettent lors de la construction du MCD de mieux comprendre les liens qui existent entre les objets.

En effet, grâce aux règles de gestion nous pouvons, entre autre, déduire des relations entre les objets et déterminer les cardinalités du MCD. (2)

-Un client commande une ou plusieurs montures.

✚ Les contraintes d'intégrités :

Comme nous venons de dire, les règles de gestion représentent les règles liées aux domaines à modéliser. Elles expriment ce que l'on appelle les contraintes d'intégrité du MCD. (2)

✚ Le MCD :

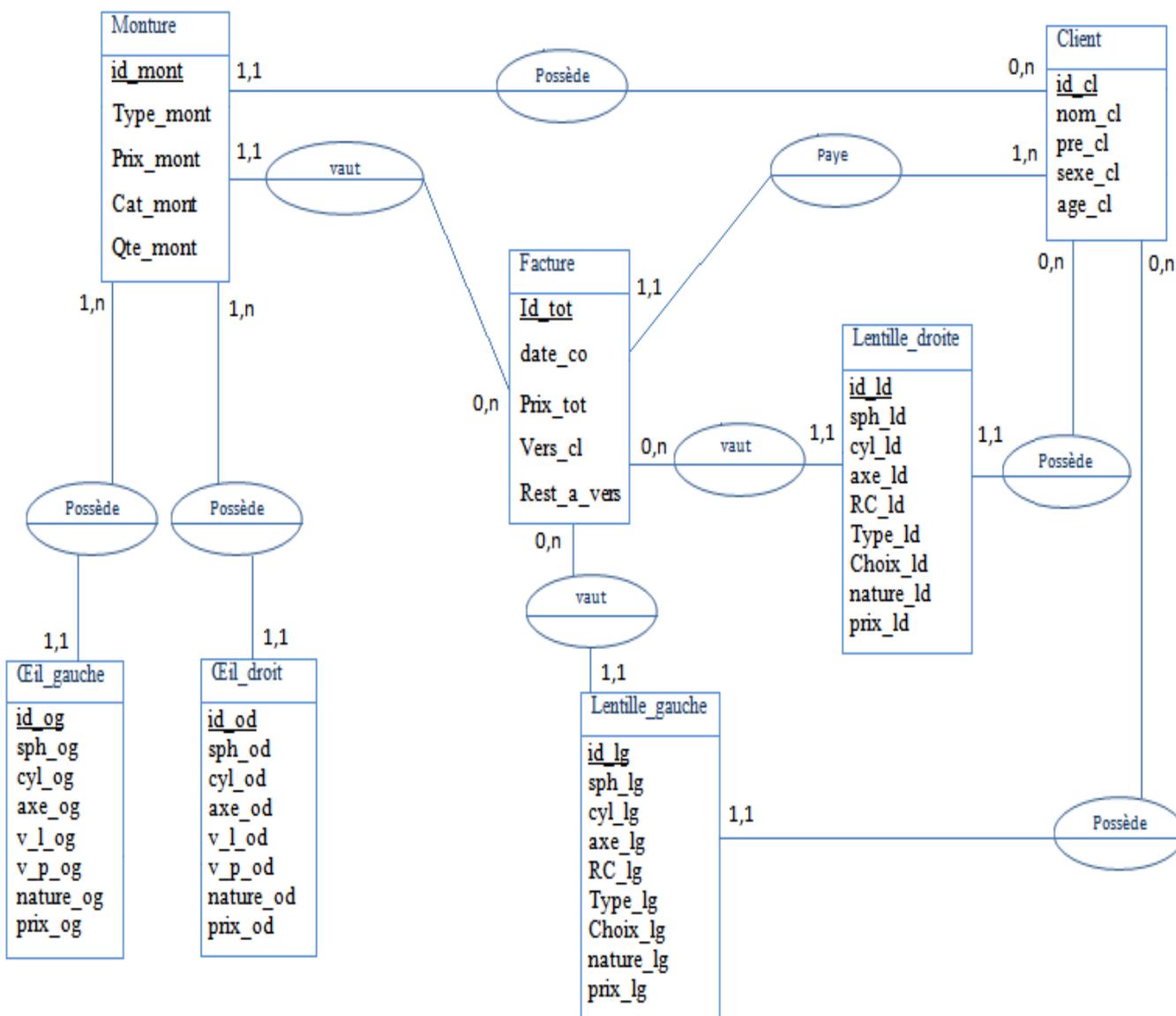


Figure I.1 : le MCD

II-1-b-Modèle Conceptuel de traitements (MCT) :

Le modèle conceptuel des traitements renferme un ensemble de concepts qui permettent de décrire les traitements du S.I sous forme de processus, de manière complète et bien explicite.

Les concepts de base utilisés par le MCT pour représenter les traitements se résument à ce qui suit : (2)

1. **Le processus** : Le processus dans le MCT est une représentation d'un sous-ensemble de l'activité de l'entreprise dont les points d'entrée et de sortie sont stables et indépendants des choix d'organisation.
2. **L'opération** : Une opération est ensemble d'actions interruptible, elle est accomplie par le S.I en réaction à un évènement ou à une conjonction d'évènements pour produire en sortie des résultats ou de nouveaux évènements.
3. **L'évènement** : On peut distinguer trois types d'évènements pour un processus :
 - Evènement externe à l'entreprise et au processus : est soit :
 - > **Déclencheur** d'une opération (sollicitation de l'environnement extérieur).
 - > **Résultat** d'une opération (réponse à l'environnement extérieur).
 - > Ce n'est jamais le **résultat** d'une opération et **déclencheur** d'une autre à la fois.
 - Evènement interne à l'entreprise et externe au processus : c'est tout évènement qui est à la fois : **Résultat** d'une opération d'un processus et **déclencheur** d'une opération d'un autre processus.
 - Evènement interne au processus (et donc à l'entreprise) : c'est un évènement qui est à la fois **résultat** d'une opération et **déclencheur** d'une autre opération dans un même processus.
4. **La synchronisation et règles de synchronisation** : c'est la combinaison d'évènements pour déclencher une opération.

→ Formalisme :

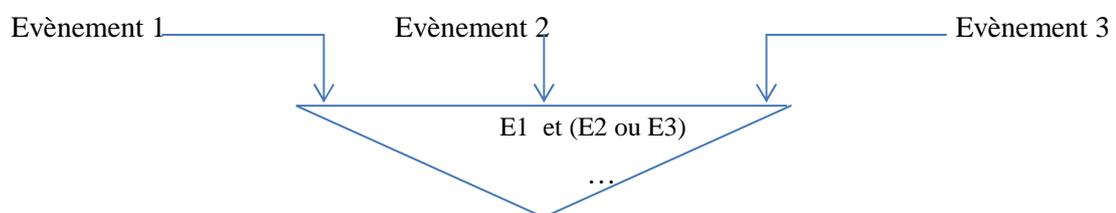


Figure I.2 : Synchronisation d'une opération dont la condition E1 et (E2 ou E3). (2)

5. **Le résultat et règles d'émission de résultats** : permet de décider pour une opération, quels résultats produire en fonction des évènements déclencheurs de celle-ci.

→ Formalisme :

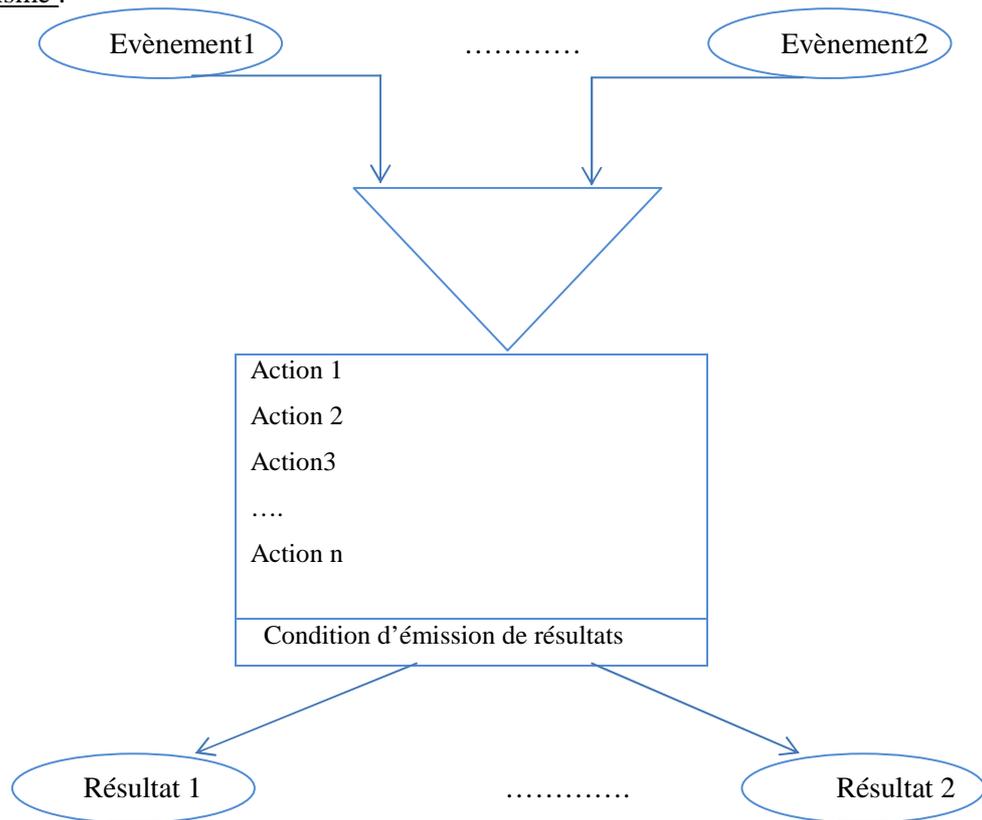


Figure I.3 : Schéma d'une opération. (2)

Les éléments d'un MCT sont représentés en adoptant un formalisme schématique qui sera décrit pour chaque concept étudié. (2)

✚ Règles relatives à la construction du MCT :

Lors de l'élaboration du MCT, il faut tenir compte des points suivants :

- l'ordre d'exécution des opérations.
- les informations échangées par les différents traitements.
- la synchronisation d'évènements et les règles d'émission de résultats.
- les règles de gestions liées aux opérations.

Les règles relatives au MCT peuvent se résumer à :

- Règle1 : la non interruption.
- Règles 2 : homogénéité des opérations.
- Règles 3 : la consommation.
- Règles 4 : la non redondance des opérations.
- Règles 5 : la synchronisation ou entre un évènement externes et un évènement interne. (2)

✚ Le MCT :

Liste des processus :

1. Vérification de l'existence du verre et si le verre s'adapte avec la monture.

2. Client change la monture.
3. Inscription client sur la BDD.
4. Client inscrit.
5. Login et entrée des informations.
6. Client reçoit une facture.

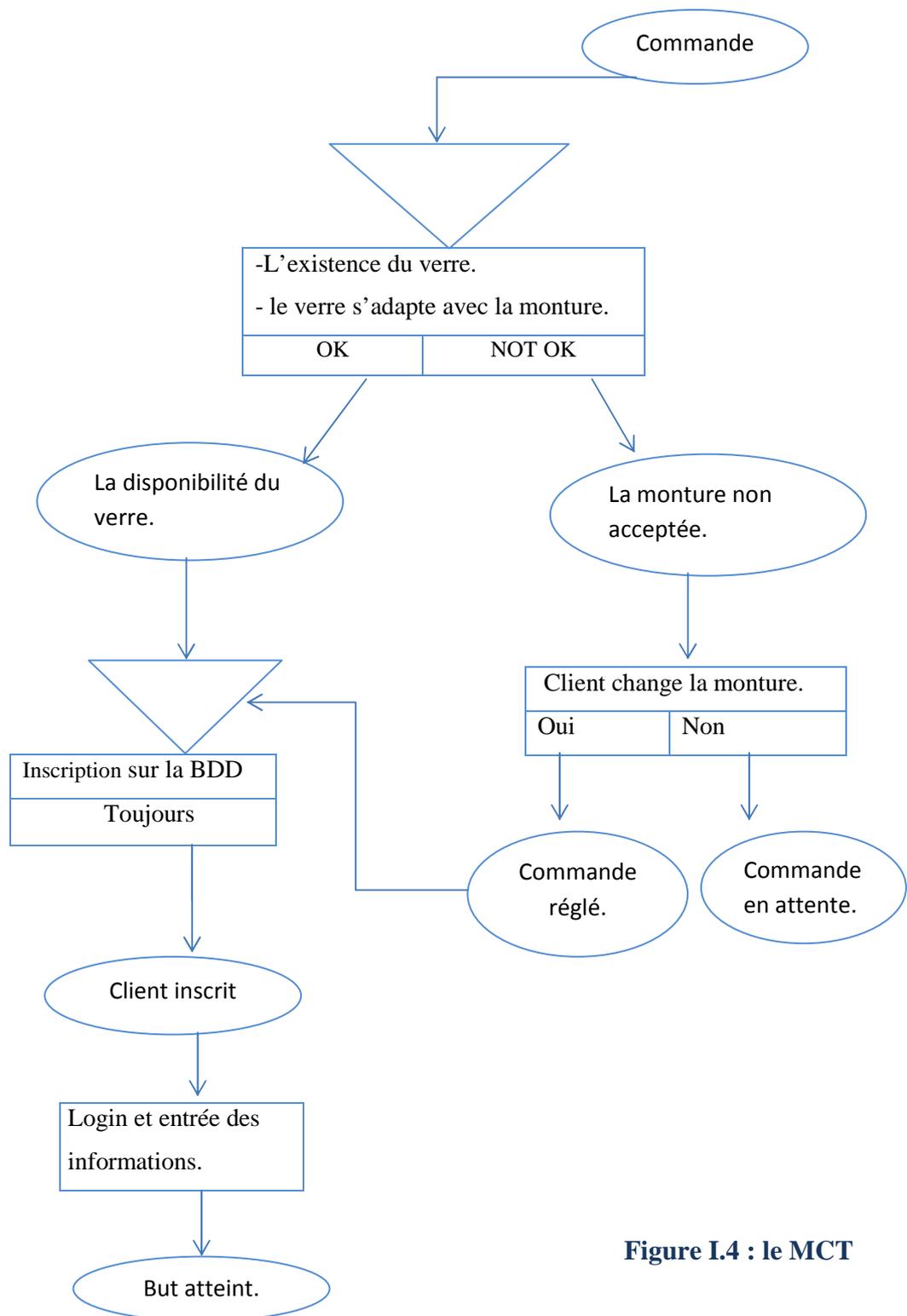


Figure I.4 : le MCT

II-2-Niveau logique/organisationnel :

A ce niveau de préoccupation, les modèles conceptuels sont précisés et font l'objet de choix organisationnels. On construit : (3)

II-2-a-Modèle Logique de Données (MLD) :

Le MLD se situe entre le modèle conceptuel de données MCD et le modèle physique de données MPD, il permet de passer du modèle conceptuel de données à son équivalent informatique, le MLD est donc construit à partir du MCD.

Avant d'énoncer les règles de passage du MCD au MLD relationnel, il faut présenter les concepts manipulés par le MLD : (2)

1. **Le domaine** : c'est l'ensemble de valeurs que peut prendre une donnée.
2. **La relation** (appelée aussi table) : c'est un sous-ensemble du produit cartésien du domaine. Ce sous-ensemble sera désigné par un nom qui sera le nom de la relation (table). Concrètement, une relation et un tableau à plusieurs colonnes concernant chacune un domaine de valeurs.
3. **L'attribut** : chaque domaine participant à la relation a un nom appelé : attribut.
4. **Le schéma d'une relation** : est l'ensemble constitué du nom de la relation suivi du nom de chacun de ses attributs.
5. **Clé d'une relation** : on appelle clé d'une relation, l'ensemble de ses attributs qui permettent d'identifier de façon unique, chacun de ses tuples. (2)

🚧 Passage du MCD au MLD :

On cite plutôt les règles de passages des objets (entités) du MCD aux tables du MLD relationnel :

- Tout objet du MCD se transforme en table dans le MLD.
- Toutes les propriétés de l'objet deviennent les attributs de la table.
- L'identifiant de l'objet devient la clé de la table. (2)

🚧 Le MLD :

Client (id_cl, nom_cl, pre_cl, sexe_cl, age_cl).

Œil_gauche (id_og, sph_og, cyl_og, axe_og, v_l_og, v_p_og, nature_og, prix_og, id_mont*)

Œil_droit (id_od, sph_od, cyl_od, axe_od, v_l_od, v_p_od, nature_od, prix_od, id_mont*)

Lentille_gauche (id_lg, sph_lg, cyl_lg, axe_lg, RC_lg, type_lg, choix_lg, nature_lg, prix_lg, id_cl*, id_tot*)

Lentille_droite (id_ld, sph_ld, cyl_ld, axe_ld, RC_ld, type_ld, choix_ld, nature_ld, prix_ld, id_cl*, id_tot*)

Monture (id_mont, Type_mont, cat_mont, qte_mont, Prix_mont, id_cl*, id_tot*)

Facture (id_tot, date_co, Rest_a_vers, Vers_cl, Prix_tot, id_cl*)

II-2-b-Modèle Organisationnel de traitements (MOT) :

Le modèles conceptuel de traitement comme son nom l'indique, permet de représenter l'ensemble des traitements en tenant compte de l'organisation de l'entreprise qui est matérialisée par les différents poste de travail, ou chaque poste correspond à une unité d'action élémentaire dotée des moyens d'exécution manuel/automatique. (2)

En plus des concepts : Règles d'émissions de résultat, La synchronisation des évènements, Evènement, Résultat vu dans le MCT, ils existent d'autres qui sont introduits dans le MOT.

La périodicité (le temps) : une tâche ne peut être exécutée à n'importe quel moment, le temps pendant lequel un traitement est exécuté, doit être déterminé. La périodicité exprime la fréquence d'exécution d'une tâche. (2)

Phase (procédure fonctionnelle) : est un enchaînement non interrompu de tâches respectant les trois règles des trois unités : poste de travail, nature, périodicité. Autrement dit : ces tâches doivent être de même nature et avoir la même périodicité et être exécutées par le même poste de travail.

Tâche : action pourvue d'une organisation définie par les règles d'organisation.

La procédure : est un enchaînement de phase concourant à une même finalité.

Règles d'organisation : une règles d'organisation exprime l'organisation mise en place en terme de poste de travail, nature de traitement et de temps (ou périodicité). (2)

→ Formalisme du MOT :

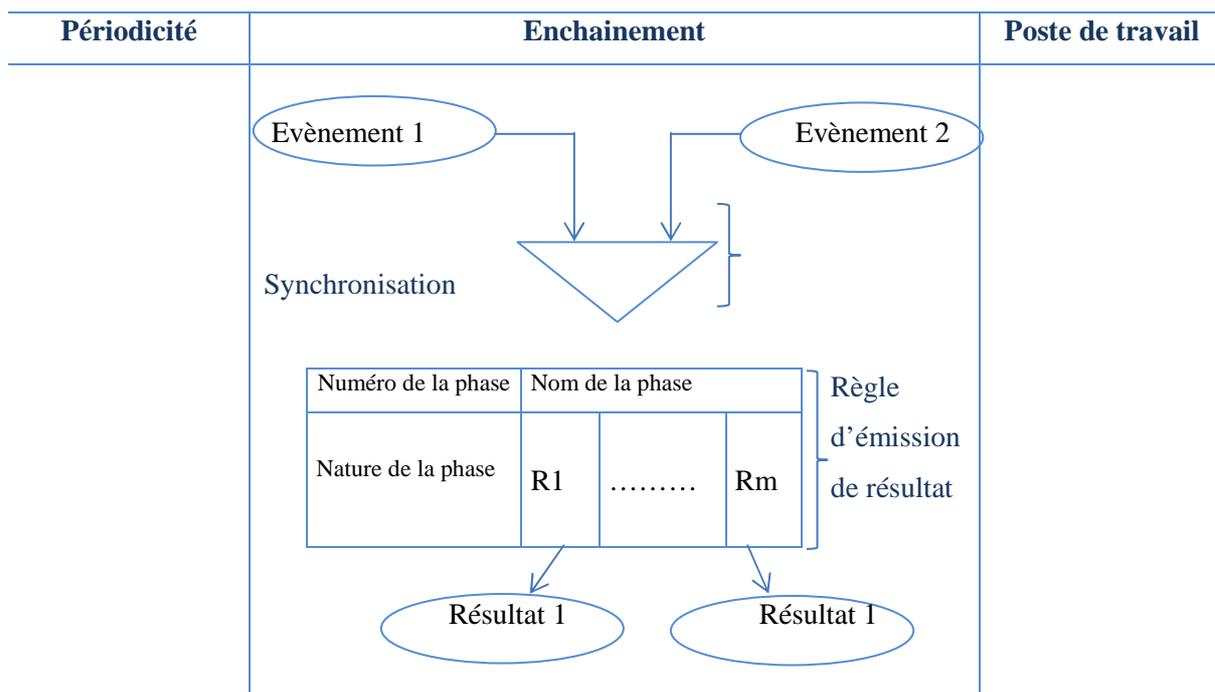
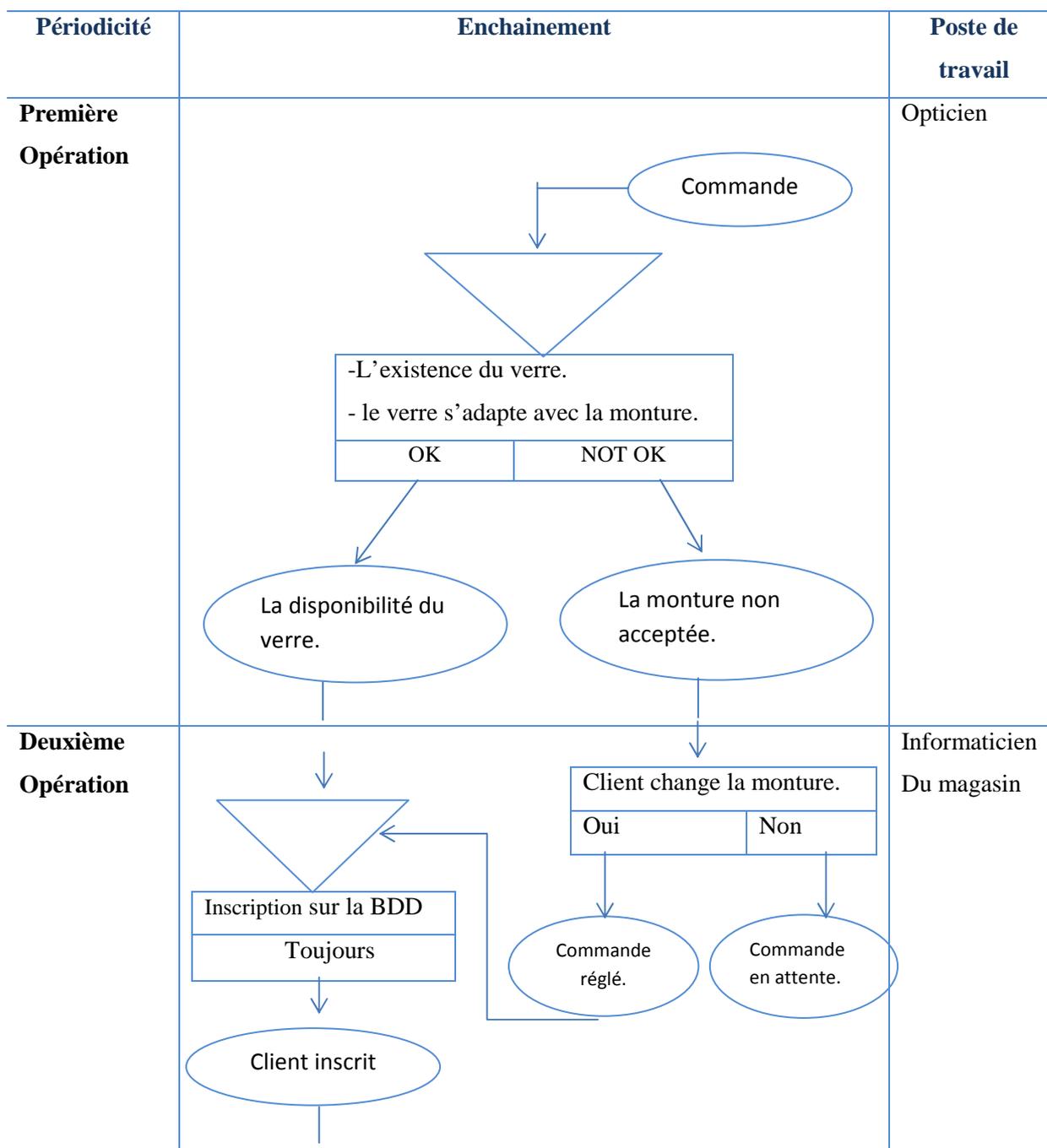


Figure I.5 : Le formalisme complet d'un MOT(2)

✚ Passage du MCT vers le MOT :

- La première étape consiste à déterminer pour chaque opération du MCT l'enchaînement des actions qu'elle renferme.
- La deuxième étape consistera à déterminer pour chaque action l'organisation qui est appliquée.
- La troisième étape aura pour but la construction du MOT. Cela se fera en regroupant tout ensemble de tâches (actions) non interrompues.
- La dernière étape consistera à créer la table descriptive des phases. (2)

✚ Le MOT :



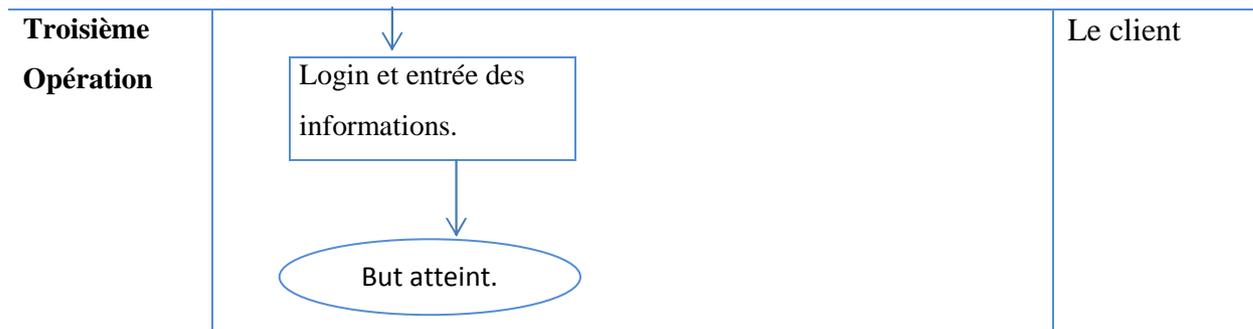


Figure I.6 : le MOT

II-3-Niveau physique/opérationnel :

Les réponses apportées à ce dernier niveau permettent d'établir la manière concrète dont le système sera mis en place. On construit : (3)

II-3-a-Modèle Physique de Données (MPD) :

Le modèle physique de données constitue le dernier modèle de données à construire dans la méthode MERISE. Il s'agit à ce stade de traduire le modèle logique de données (MLD) en une base de données qui sera implémentée dans une machine. (2)

Le MPD prendra en considération les ressources physiques.

- a. **On devra donc spécifier** : le matériel et le type de système de gestion de base de données (SGBD) à utiliser pour créer et gérer la future base de données.
le SGBD utilisé Oracle Database 10g
Ordinateur :
DELL Inspiron N4030
SE : Windows Seven Titan 32bits
- b. **La capacité de traitements de l'unité centrale** (processeur + mémoire centrale) :
La capacité de traitement de l'unité central:
CPU INTE Core I3 2.53GHz
4.00Go de RAM
- c. **La capacité de stockage** : selon le volume des données à stocker (taille des bases de données).
La capacité de stockage : Disque dur : 500 Go

II-3-b-Modèle Opérationnel de Traitements (MOpT) :

Le MOpT est le dernier modèle des traitements à construire. On arrive à ce stade à l'étape de réalisation du logiciel. L'objectif du MOpT est de traduire les phases automatisées du MOT en un ou plusieurs programmes. Le MOpT renferme les programmes de calcul, de saisie et d'édition. Pour créer le logiciel, un environnement de développement (langage de programmation) devra être choisi. Dans notre cas c'est Lazarus . (2)

II-Conclusion:

La méthode Merise aide a faire une séparation (dicotomie) entre les données et traitements en proposant trois modèles pour l'analyse et la conception d'une BDD.

On obtient deux visions du système d'information :

- Une vision statique (données).
- Une visions dynamique (Traitements).

Chapitre II

Réalisation du Projet

I-Introduction :

Après avoir établi une études conceptuelle de mon système, je passe à l'implémentation de l'application définis et détaillée au chapitre précédent tout en présentant les outils utilisés et en expliquant les écrans de notre logiciel. Cette implémentation est basée sur la création de notre BDD sous oracle et réalisée avec IDE Lazarus.(6)

II-Introduction au SGBD Oracle :

Oracle est un SGBD (système de gestion de bases de données) édité par la société du même nom (Oracle Corporation), leader mondial des bases de données.

La société Oracle Corporation a été créée en 1977 par Lawrence Ellison, Bob Miner, et Ed Oates. Elle s'appelle alors Relational Software Incorporated (RSI) et commercialise un Système de Gestion de Bases de données relationnelles (SGBDR ou RDBMS pour Relational Database Management System) nommé Oracle.

En 1979, le premier prototype (RDBMS - RSI1) intégrant la séparation des espaces d'adressage entre les programmes utilisateurs et le noyau Oracle est commercialisé. Cette version est entièrement développée en langage assembleur. La seconde version (RDBMS - RSI2) est un portage de l'application sur d'autres plates-formes.

En 1983 la troisième version apporte des améliorations au niveau des performances et une meilleure prise en charge du SQL. Cette version est entièrement codée en langage C. A la même époque RSI change de raison sociale et devient Oracle.

En 1984 la première version d'Oracle (Oracle 4) est commercialisée sur les machines IBM.

En 1985 Oracle 5 permet une utilisation client-serveur grâce au middleware SQL*Net.

En 1986 Oracle a été porté sur la plateforme 8086.

En 1988 Oracle 6 est disponible sur un grand nombre de plates-formes et apporte de nombreuses nouvelles fonctionnalités ainsi qu'une amélioration notable des performances.

En 1991, Oracle 6.1 propose une option Parallel Server (dans un premier temps sur la DEC VAX, puis rapidement sur de nombreuses autres plates-formes).

En 1992, Oracle 7 sort sur les plates-formes UNIX (elle ne sortira sur les plates-formes Windows qu'à partir de 1995). Cette version permet une meilleure gestion de la mémoire, du CPU et des entrées-sorties. La base de données est accompagnée d'outils d'administration (SQL*DBA) permettant une exploitation plus aisée de la base. En 1997, la version Oracle 7.3 (baptisée Oracle Universal Server) apparaît, suivie de la version 8 offrant des capacités objet à la base de données

Oracle est écrit en langage C et est disponible sur de nombreuses plates-formes matérielles

(plus d'une centaine) dont :

- AIX (IBM)
- Solaris (Sun)
- HP/UX (Hewlett Packard)
- Windows NT (Microsoft) (5)

✚ Les fonctionnalités d'Oracle :

Oracle est un SGBD permettant d'assurer :

- La définition et la manipulation des données
- La cohérence des données
- La confidentialité des données
- L'intégrité des données
- La sauvegarde et la restauration des données
- La gestion des accès concurrents (5)

✚ Outils de programmation :

Oracle dispose d'un grand nombre d'interfaces (API) permettant à des programmes écrits dans divers langages de s'interfacer avec la base de données en envoyant des requêtes SQL. Ces interfaces (appelées précompilateurs) forment une famille dont le nom commence par *PRO** :

- Pro*C
- Pro*Cobol
- Pro*Fortran
- Pro*Pascal
- Pro*PLI (5)

L'interface que j'ai utilisé est Pro*Pascal.

III-Introduction au Lazarus :

Lazarus est un EDI multiplateforme développé en Pascal Objet et pour Free Pascal. Son objectif est de fournir aux programmeurs Pascal Objet un environnement de développement s'approchant le plus possible de Delphi.

Free Pascal est un compilateur sous licence GPL fonctionnant sous Linux, Win32, OS/2, MAC OS X, BSD, etc. Il est conçu pour interpréter et compiler le Pascal Objet de Delphi. À la différence de Java dont le slogan est « write once run anywhere » (écrire une fois, exécuter partout), celui de Lazarus et Free Pascal est « write once compile anywhere ».(3)

✚ Processus de développement :

Lazarus est un logiciel de type RAD, similaire à l'EDI Delphi d'Embarcadero. Sous Microsoft Windows, Delphi reste le plus stable et a la meilleure documentation, mais il n'a pas les possibilités multiplateforme de Lazarus. Sous Linux, Lazarus est l'un des meilleurs EDI disponibles. Les applications Lazarus peuvent être compilées sous Delphi et vice-versa

moyennant quelques adaptations aux programmes (à comparer à la réécriture complète du code lorsque la bascule se fait d'un langage à un autre). Ce processus est plus facile lorsque le programme ne fait pas appel à des formulaires générés par l'éditeur car il y a certaines incompatibilités. Les composants de Delphi peuvent être installés sous l'EDI, mais ils doivent être convertis, via un processus complexe. (3)

IV-Les interfaces de l'application:

IV-1- Principale :

Cette fenêtre autorise à l'opticien seulement pour accéder à l'application après avoir saisi le nom d'utilisateur et le mot de passe :

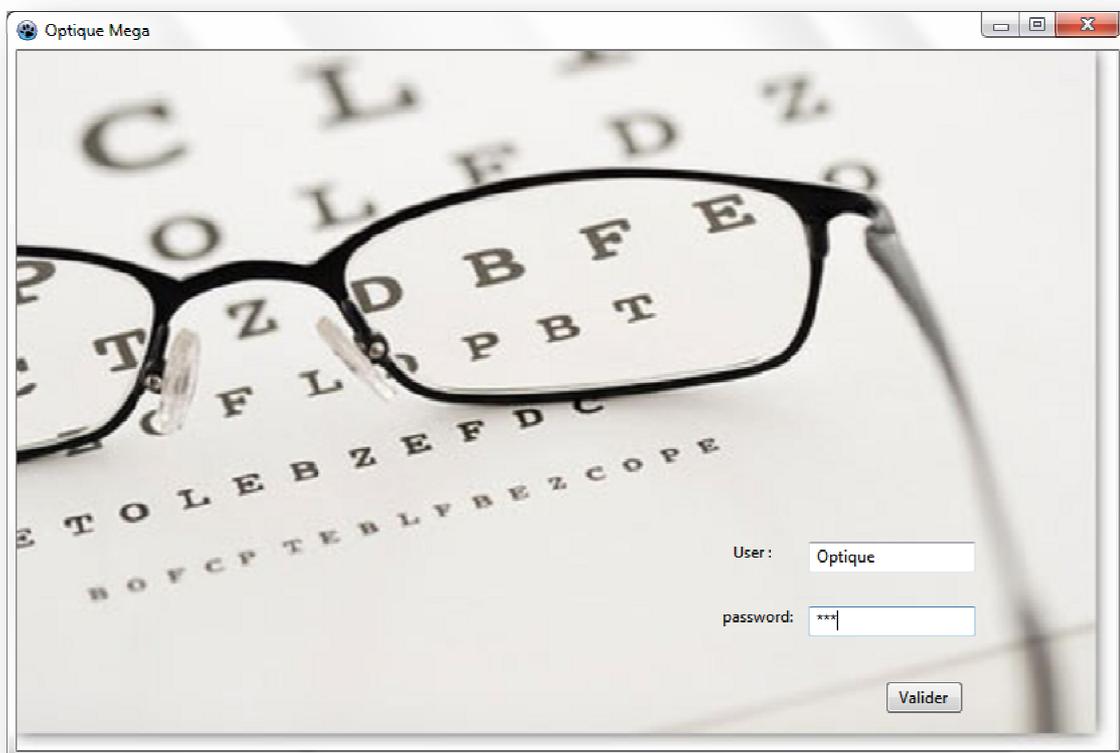


Figure II.1 : Principale

Si l'utilisateur est éroné un message sera affiché et pour le mot de passe aussi :



Figure II.2 : Erreur user ou password

IV-2-Entrée :

Après avoir accéder à l'application une fenêtre s'affiche ou l'opticien peut trouver les champs essentiels permettant de :

- Saisir un nouveau client
- Modifier un client existant
- La recherche d'un client existant
- Gérer le stock des montures

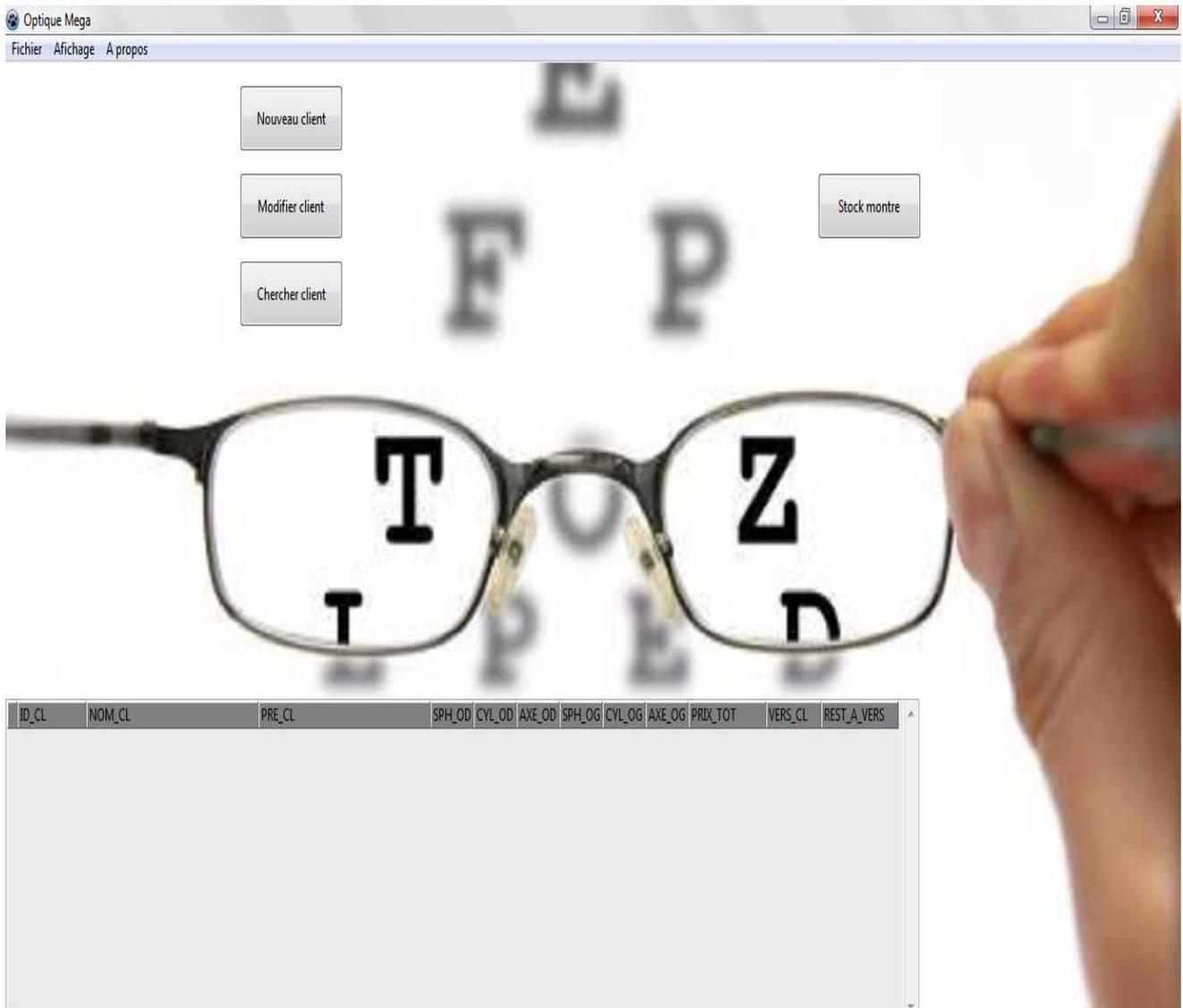
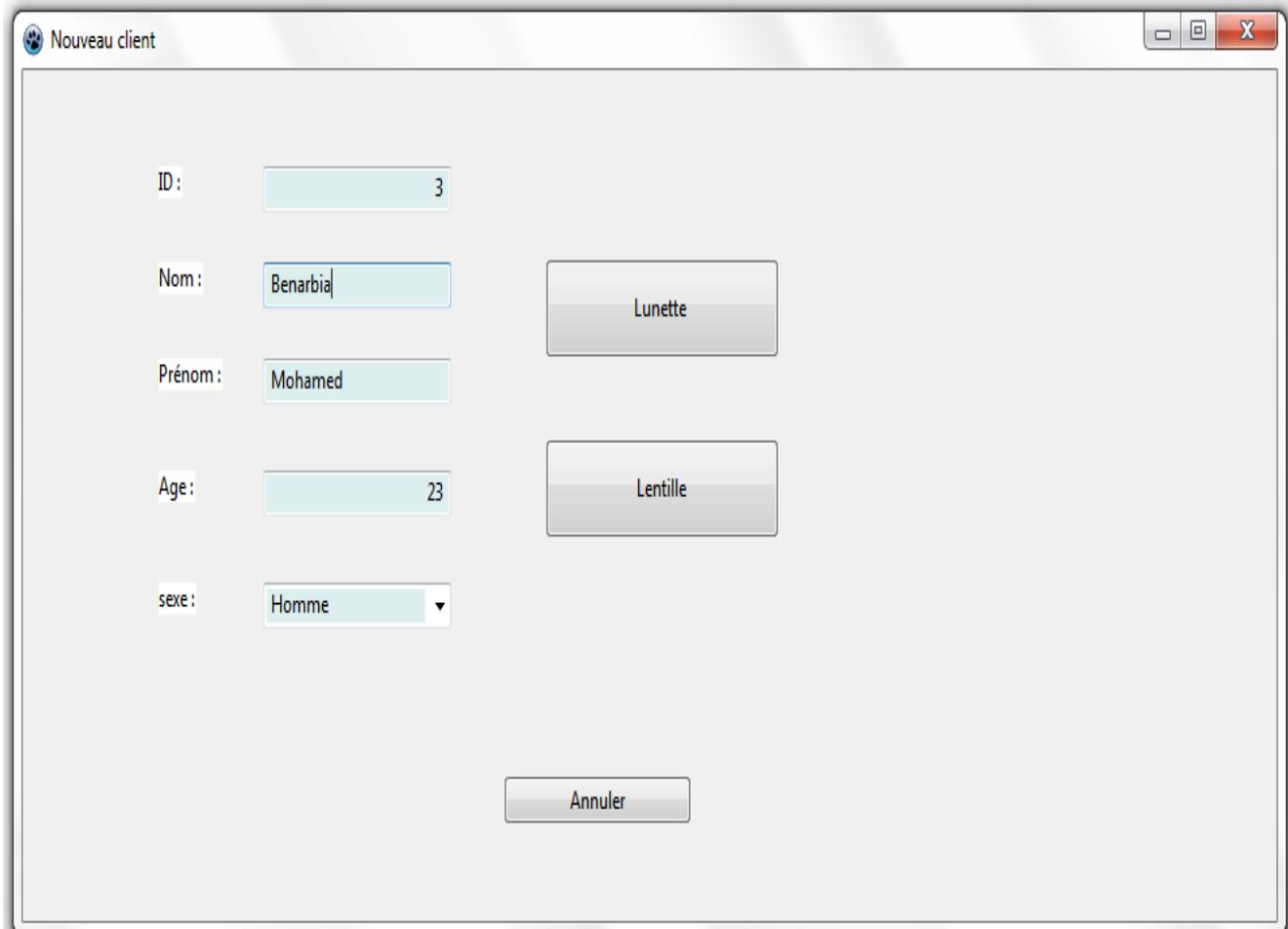


Figure II.3 : Fenêtre d'entrée

IV-3-Ajout :

Dans cette fenêtre on effectue l'ajout d'un client et la saisie des coordonnées personnel de ce dernier. Sauf l'ID qui s'incrémente automatiquement en ajoutant 1 au dernier identifiant existant .



The screenshot shows a window titled "Nouveau client" with the following fields and buttons:

- ID: 3
- Nom: Benarbia
- Prénom: Mohamed
- Age: 23
- sexe: Homme
- Buttons: Lunette, Lentille, Annuler

Figure II.4 : Fenetre client

Et il y a deux cas de correction optique une en portant des Lunettes ou des lentilles :

IV-3-a-Lunette :

Correction

Nom : Benarbia

Prénom : mohamed

	sph	Cyl	axe	vision de loin	vision de prés	Nature	Prix
Oeil Droit							
Oeil Gauche							

Saisir le versement :

enregistrer Anuuler

Figure II.5 : Fenetre correction

Après l'enregistrement de coordonnées des yeux d'un client un message sera affichée qui demande si le client veut acheter une monture :

Correction

Nom : Benarbia

Prénom : mohamed

	sph	Cyl	axe	vision de loin	vision de prés	Nature	Prix
Oeil Droit							
Oeil Gauche							

Error

Voulez vous acheter une monture ?

OK Cancel

enregistrer Anuuler

Figure II.6 : Fenetre message d'achat monture

Si oui une autre fenetre dera affichée pour choisir la monture :

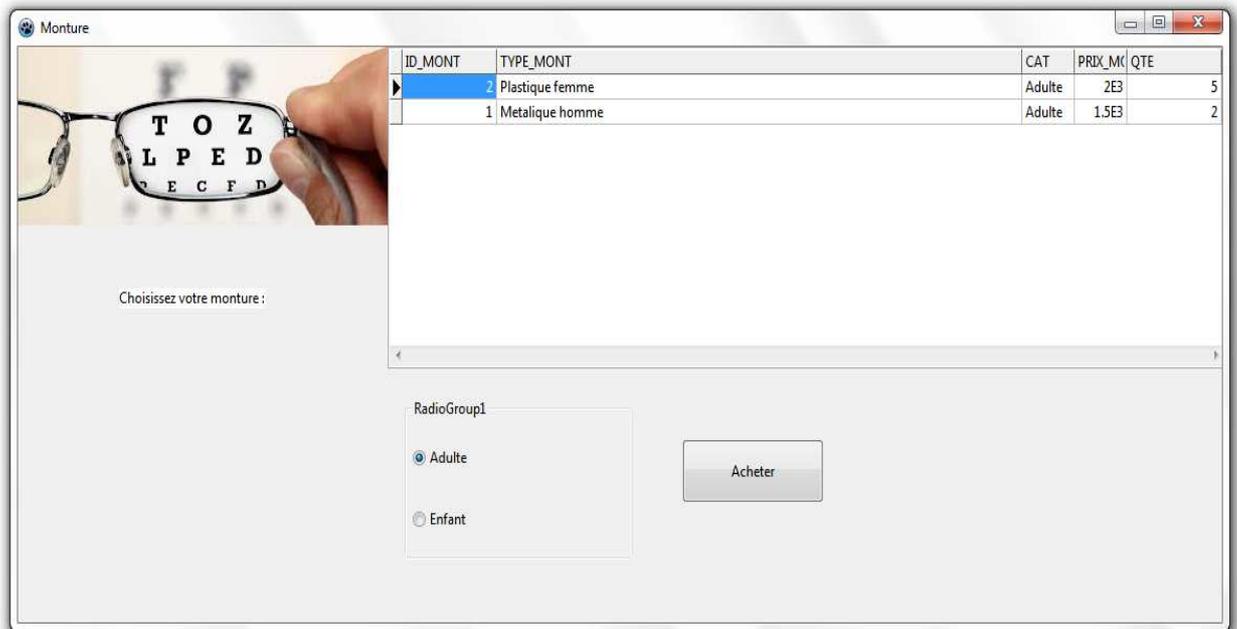


Figure II.7 : Fenetre achat monture

Et quand il selectionne la monture choisie la quatité sera automatiquement décrémentée et la facture sera prête

Sinon .. si le client n'achete pas une monture la facture sera faite seulement pour les verres .

IV-3-b-Lentille :

L'enregistrement des référence de la lentille s'effectue et la facture sera prête.

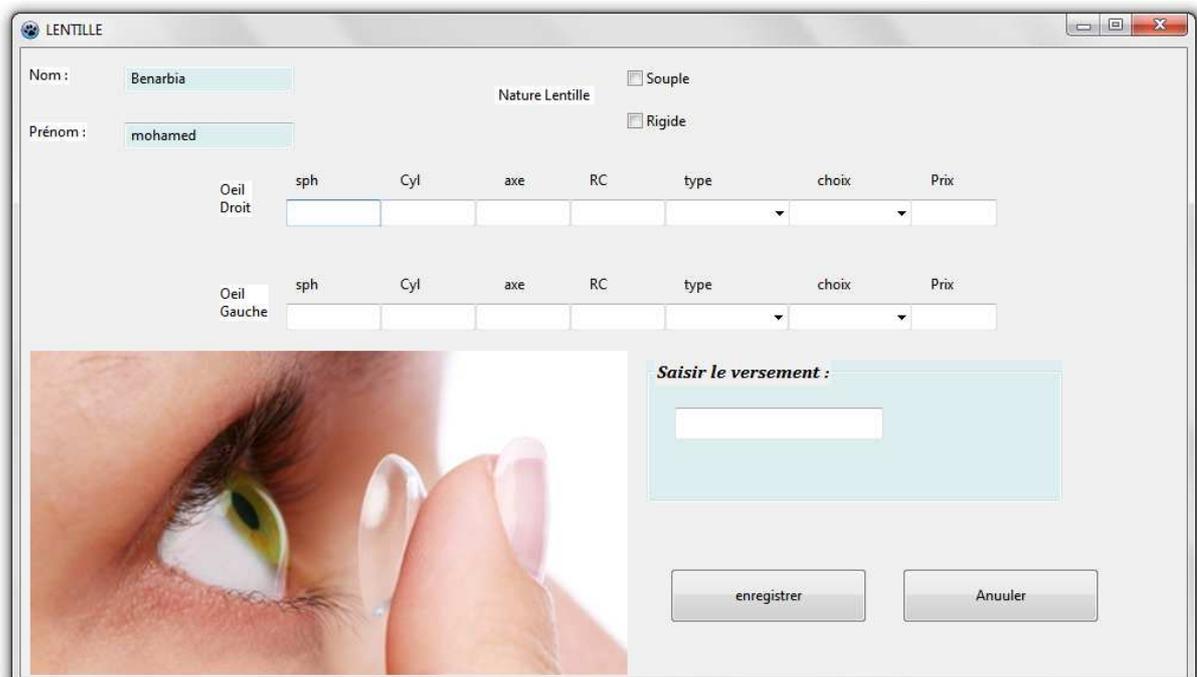


Figure II.8 : Fenetre lentille

IV-4-Modification :

Sera faite au niveau de la fenetre correction ou lentilles selon le client selectionné.

IV-5-Recherche :

En cliquant sur le bouton ‘chercher client’ une fenetre s’aparaitra, demande la saisie du nom de client et effectue une recherche en donnant ses cordonnées

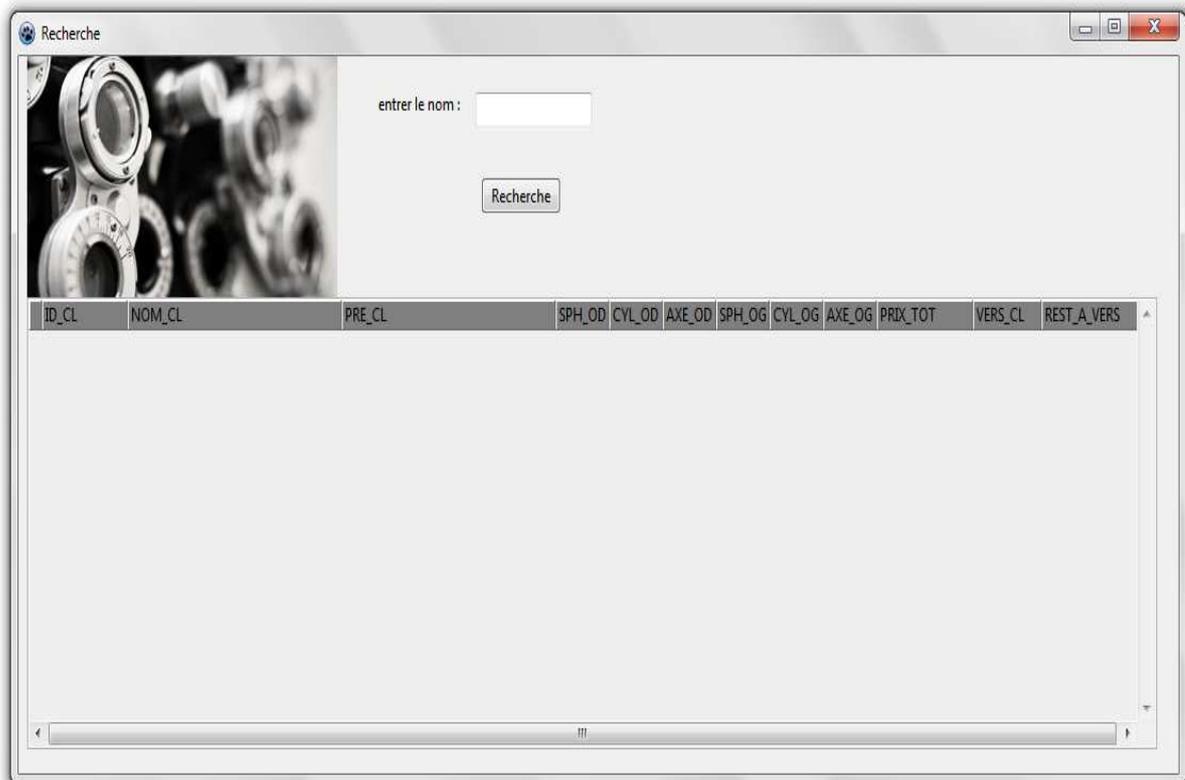


Figure II.9 : Fenetre recherche

IV-6-Stock de monture :

C’est la gestion de stock des montures disponibles dans le magasin.

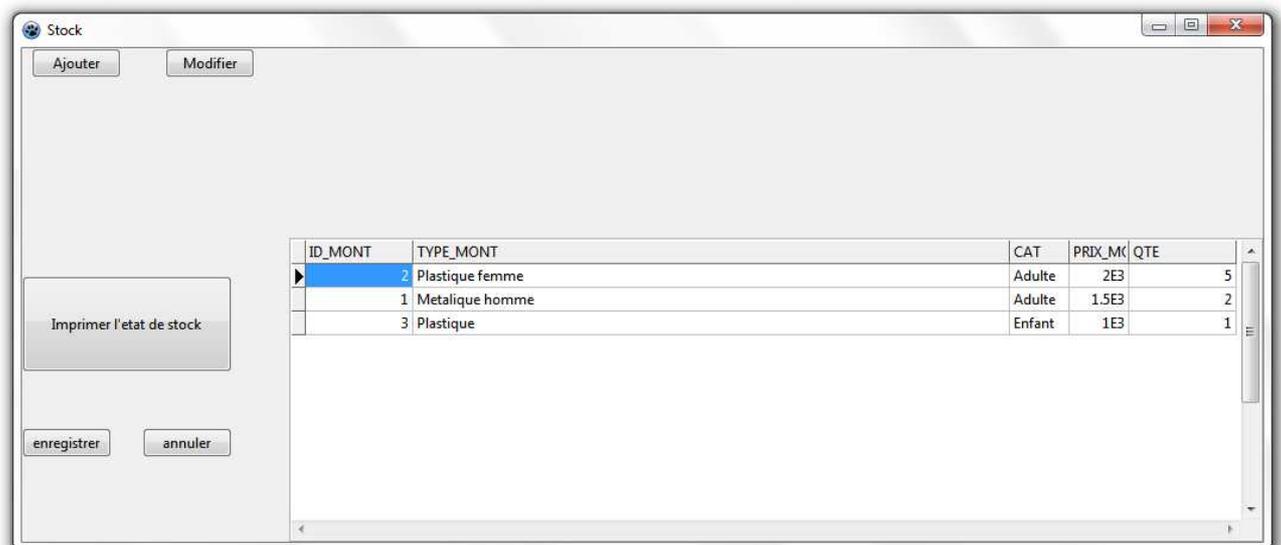
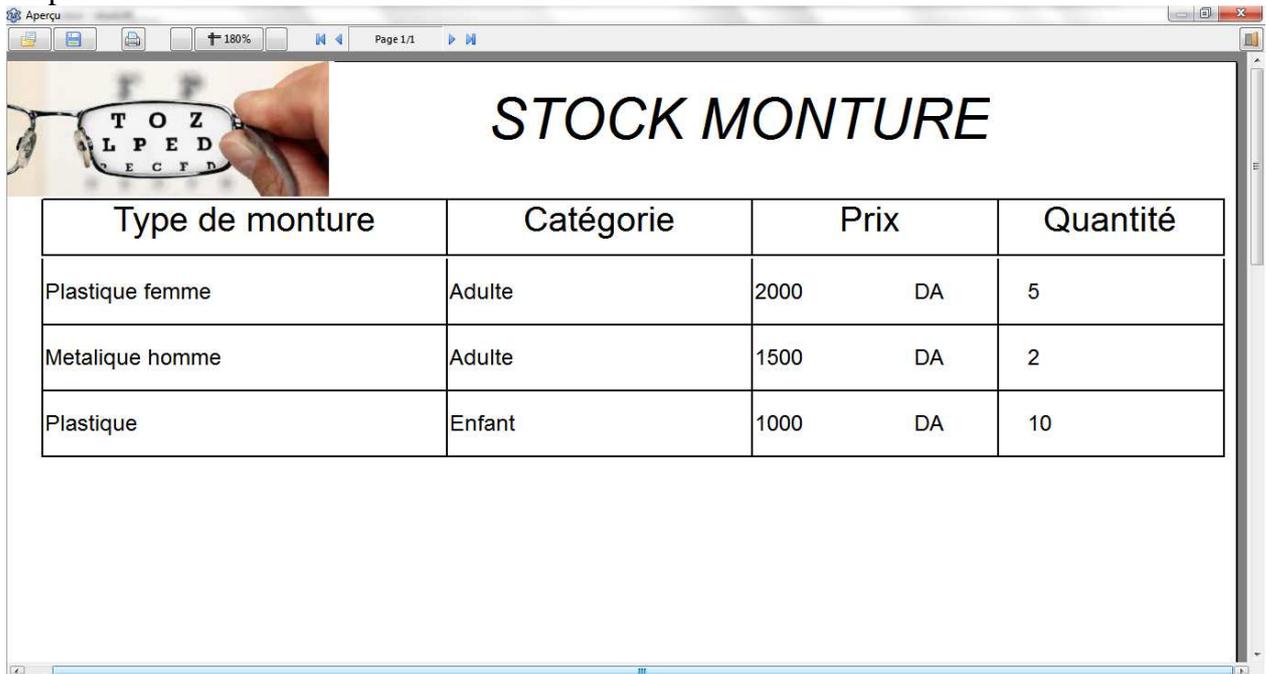


Figure II.10 : Fenetre stock

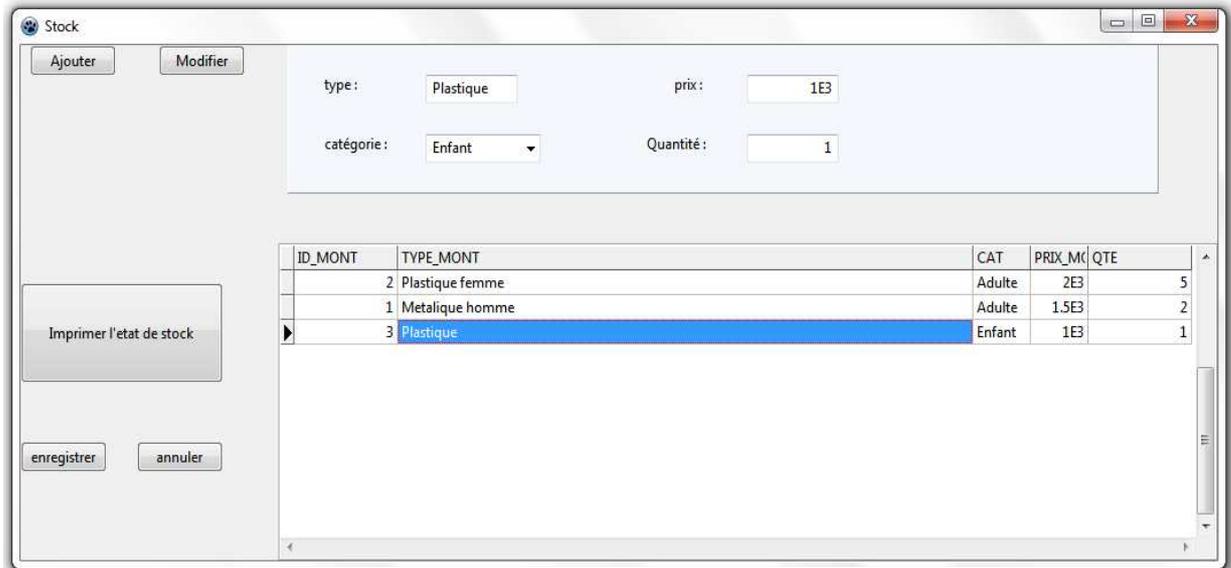
Pour afficher l'état de stock et les montures existantes dans le magasin on clique sur le bouton 'Imprimer l'état de stock' et le résultat sera comme suit :



Type de monture	Catégorie	Prix	Quantité
Plastique femme	Adulte	2000 DA	5
Metalique homme	Adulte	1500 DA	2
Plastique	Enfant	1000 DA	10

Figure II.11 : Etat de stock

Pour modifier une monture , on clique sur modifier puis un panel sera affiché dont ont peut saisir en sélectionnant la monture qu'on souhaite la modifier :



ID_MONT	TYPE_MONT	CAT	PRIX_Mc	QTE
2	Plastique femme	Adulte	2E3	5
1	Metalique homme	Adulte	1.5E3	2
3	Plastique	Enfant	1E3	1

Figure II.12 : Fenetre modification d'une monture.

Pour ajouter une monture, on clique sur ajouter et le même panel sera affiché dont on peut saisir la nouvelle monture ou son indentifiant s'incrémente automatiquement :

The screenshot shows a window titled "Stock" with a form for adding a new item. The form has the following fields:

- type:
- prix:
- catégorie:
- Quantité:

Below the form is a table with the following data:

ID_MONT	TYPE_MONT	CAT	PRIX_MK	QTE
2	Plastique femme	Adulte	2E3	5
1	Metalique homme	Adulte	1.5E3	2
4				
3	Plastique	Enfant	1E3	1

The "Ajouter" button is highlighted in blue. Other buttons include "Modifier", "enregistrer", "annuler", and "Imprimer l'etat de stock".

Figure II.13 : Fenetre d'ajout d'une monture.

V-Conclusion :

Dans ce chapitre, j'ai créer ma BDD sous Oracle SGBD et Lazarus pour l'interface de l'application que j'ai aussi présenté en détaille en expliquant chaque fenetre et chaque instruction dans l'application.

Conclusion Générale :

Au cours de ce mémoire, j'ai présenté les différentes étapes de l'analyse, la conception et la réalisation de mon application pour la gestion de magasin d'optique médicale.

Afin de satisfaire les besoins d'un opticien pour gérer son magasin, j'ai commencé la conception en utilisant la méthode d'analyse Merise de conception et de gestion de projet informatique et cette application a été réalisée avec Lazarus IDE.

Ce projet a fait l'objet d'une expérience intéressante, qui permis d'améliorer les connaissances et les compétences dans le domaine de la conception et la programmation des systèmes d'informations.

Le but souligné a atteint l'état d'offrir aux 'OPTICIENS' un travail correct qui répond à leurs besoins actuels dans la gestion de gérer ce magasin.

En fin, je souhaite que ce rapport soit un bon guide pour d'autres promotions ainsi que l'application réalisé.

Bibliographie

&

Web graphie :

- (1) Mémoire licence PROMOTION 2007 : préparé par : Chikioui Rima et Kheraif Nesrine UBM Annaba .
- (2) Pratique des système d'information : MERISE, par : Mc BELAID et D.BOUYACOUB (PagesBleues).
- (3) www.wikipedia.org/
- (4) <http://mrproof.blogspot.com/>
- (5) <http://www.memoireonline.com/>
- (6) Mémoire licence PROMOTION 2011 :Gestion d'un Magasin de Livre préparé par : Laribi Imen et Rimouche Imen UABT Tlemcen.

Résumé :

Mon projet consiste à créer une gestion de magasin d'optique médicale , l'analyse et la conception sont faites en utilisant la méthode Merise et la création de la base de données a été sous Oracle SGBD . L'application a été réalisé en Lazarus IDE free Delphi .

Mot-clés : Merise , Lazarus , Oracle .

Abstract :

My project is to create a management medical optics store, analysis and design are made using the method Merise and the creation of the database have been done in Oracle SGBD. The application was made in Lazarus IDE Delphi free.

Keywords: Merise, Lazarus, Oracle.

ملخص :

في اطار تقديم مشروع نهاية الدراسة، مشروع حول تنفيذ برنامج يسمح للمبصاري بإدارة محل النظارات الطبية بطريقة سهلة و غير تقليدية، و من أجل تحقيق هذا البرنامج قمت بدراسة تحليلية بإستعمال Merise بعدها قمت بإنشاء قاعدة البيانات بإستعمال نظام إدارة قواعد البيانات Oracle SGBD و أخيراً تم تنفيذ البرنامج بالإعتماد على Lazarus IDE .