



ECOLE POLYTECHNIQUE DE TUNISIE

Projet java

Réalisé par :

Ilyes TRABELSI
Maher DISSEM
Fayad ALI BANNA
Saif eddine JARBOUI
Aymen BEN KHALED

Encadré par :
Imen AMDOUNI

Table des matières

| | | |
|----------|---|-----------|
| 1 | Introduction générale : | 1 |
| 2 | Définition et analyse du cahier des charges : | 2 |
| 2.1 | Introduction | 2 |
| 2.2 | Présentation du cahier des charges | 2 |
| 2.3 | Analyse du cahier des charges | 3 |
| 2.4 | Conclusion | 3 |
| 3 | Conception : | 5 |
| 3.1 | Modélisation de la base de donnée | 5 |
| 3.2 | Identification des cas d'utilisation | 5 |
| 3.2.1 | Cas d'utilisation des administrateurs | 5 |
| 3.2.2 | Cas d'utilisation des membres de l'association | 6 |
| 4 | Réalisation : | 7 |
| 4.1 | Introduction | 7 |
| 4.2 | L'environnement du travail | 7 |
| 4.2.1 | Environnement matériel | 7 |
| 4.2.2 | Environnement logiciel | 7 |
| 4.2.2.1 | Système d'exploitation | 7 |
| 4.2.2.2 | Outils de développement | 7 |
| 4.2.2.3 | Environnement de développement intégré | 8 |
| 4.2.2.4 | Outil d'administration de la base de données | 8 |
| 4.3 | Mode de fonctionnement des interfaces élaborées | 8 |
| 4.3.1 | Users | 9 |
| 4.3.2 | Members | 10 |
| 4.3.3 | Donators | 11 |
| 4.3.4 | Receivers | 13 |
| 4.3.5 | Articles | 13 |
| 4.3.6 | Request | 15 |
| 4.3.7 | Deposite-Sale | 17 |
| 4.3.8 | Statistics | 18 |
| 4.4 | Conclusion | 19 |
| 5 | Tests et Validation | 20 |
| 5.1 | Introduction | 20 |
| 5.2 | Tests | 20 |
| 5.3 | Intégration | 23 |
| 5.4 | Conclusion | 23 |
| 6 | Conclusion générale | 24 |

Chapitre 1

Introduction générale :

La programmation orientée objet (POO) est un paradigme de programmation informatique élaborée au début des années 1960. Elle consiste en la définition et l'interaction de briques logicielles appelées objets qui représente un concept, une idée ou toute entité du monde physique. Il possède une structure interne, un comportement, et il sait interagir avec ses pairs.

Ce paradigme, qui bien sûr comme tous les autres, présente bien évidemment ses utilités. Selon le problème auquel on est confronté, opté pour l'orienté objet peut être la meilleure solution. C'est donc dans ce sens, et en tant que futur polytechnicien, qu'il est de rigueur pour nous d'acquérir des bases solides dans ce domaine. Quoi de mieux donc qu'après 30h d'enseignement orientés objet sur Java : cours, travaux dirigés et travaux pratiques confondus, de mettre notre connaissance à l'épreuve en se penchant dans la réalisation d'un mini-projet.

Chapitre 2

Définition et analyse du cahier des charges :

2.1 Introduction

Une association de bienfaisances en Tunisie a pour but d'aider les familles nécessiteuses. Elle fournit des besoins matériels à ces familles si elles en font la demande.

Cette association est donc fréquemment sollicitée pour recevoir des dons d'objets, elle doit étudier ces offres afin de décider si elle accepte de les acquérir ou non et ensuite accepter ou refuser les demandes de dons.

Pour ce faire, cette association souhaite disposer d'une application informatique pour faciliter la gestion des dons et alléger les processus administratifs.

Nous commencerons dans cette partie par présenter la description et l'analyse du cahier de charge proposé par cette association ainsi que les différents besoins de l'application

2.2 Présentation du cahier des charges

- La gestion des propositions de dons (les offres)

Une proposition de dons faite à l'association est caractérisée par un identifiant unique, une date de réception de l'offre, une description du type de matériel qui comprend la nature et la quantité de l'offre, la référence de l'objet, le nom, le prénom, l'adresse et une description du potentiel donneur. Si l'offre de don est acceptée alors l'offre doit contenir en plus la date d'acceptation de l'offre ainsi que les informations personnelles du membre de l'association l'ayant acceptée.

- La gestion du stock de l'association

Le stock de l'association est constitué par les offres acceptées par l'association. Lorsqu'un objet quitte le stock, il a deux destinations possibles : le don-direct où il est remis directement à un individu ayant fait la demande ou le dépôt-vente où l'objet est destiné à être vendu mais il peut arriver qu'un objet du dépôt-vente retourne dans le stock afin d'être remis à une personne ayant fait une demande.

- La gestion du don direct

Un don direct fait à un bénéficiaire doit contenir les informations à propos du bénéficiaire à savoir le nom, le prénom, l'adresse, le numéro de téléphone, l'adresse mail du bénéficiaire. Il doit également contenir les informations générales sur le don à savoir, une description du type de matériel qui comprend la nature et la quantité de l'offre, la référence de l'objet ainsi que la date d'acceptation et les informations générales à propos du membre l'ayant accepté.

- Les statistiques

On doit pouvoir suivre en temps réel l'évolution du stock, le nombre de propositions de dons reçues et acceptées ainsi que les ratio accepté/refusé par catégorie d'objet, le nombre de donateurs et de bénéficiaires, Les volumes de dons en dépôt-vente et en don direct, les principales catégories en stock et les statistiques sur les dons ayant transité par l'association.

- La recherche rapide

On doit pouvoir rechercher les objets dans le stock par catégorie, par mot figurant dans la description et par propriété par des objets.

2.3 Analyse du cahier des charges

Du cahier de charge ci-dessus, il en sort que l'application que l'application doit fournir les services suivants :

- La gestion des membres

Un membre de l'association est caractérisé par un identifiant unique, un nom, un prénom, une adresse, une adresse mail et un numéro de téléphone.

Il faut pouvoir :

- Consulter la liste des membres
- Ajouter un membre
- Supprimer un membre
- Editer les informations concernant un membre

- La gestion des donateurs

Un donneur potentiel de l'association est caractérisé par un identifiant unique, un nom, un prénom, une adresse, une adresse mail et un numéro de téléphone.

Il faut pouvoir :

- Consulter la liste des donateurs
- Ajouter un donneur
- Editer les informations concernant un donneur

- La gestion des bénéficiaires

Un bénéficiaire est caractérisé par un identifiant unique, un nom, un prénom, une adresse, une adresse mail et un numéro de téléphone.

Il faut pouvoir :

- Consulter la liste des bénéficiaires
- Ajouter un bénéficiaire
- Editer les informations concernant un bénéficiaire

- La gestion du stock

Chaque article du stock de l'association est caractérisé par un identifiant unique, une référence, une référence, le statut de l'article, la date du don, la date de réception du don, la date d'acceptation du don, la quantité reçue, les informations sur le membre ayant accepté la demande.

On doit pouvoir :

- Ajouter des articles au stock
- Modifier les informations concernant les articles
- Mettre à jour les différents statuts (accepté, refusé, en cours de traitement, dépôt, donné) d'un article
- Avoir les informations sur la catégorie de chaque article
- Envoyer des objets au dépôt-vente
- Faire des dons directs aux bénéficiaires

- La gestion du dépôt-vente

Chaque article du dépôt-vente est caractérisé par un identifiant unique, la date du dépôt, le statut de l'article, le prix estimé, le prix de vente, la date de vente.

Il faut pouvoir :

- Consulter la liste des articles dans le dépôt
- Modifier les informations concernant chaque article
- Retourner les articles demandés par des potentiels bénéficiaires pour faire un don direct

- La gestion des requêtes

Chaque requête est caractérisée par un identifiant unique, un statut.

Il faut pouvoir :

- Ajouter de nouvelles requêtes
- Consulter la liste des requêtes
- Mettre à jour le statut de la requête

- La gestion des utilisateurs

Chaque utilisateur de l'application est caractérisé par un identifiant unique, un nom d'utilisateur, un mot de passe et une autorisation.

Il faut pouvoir :

- Ajouter un utilisateur
- Consulter la liste des utilisateurs

2.4 Conclusion

Dans ce chapitre on a présenté le cahier de charge proposé par l'association et son analyse nous a permis de diviser le travail en plusieurs modules que l'on va concevoir et réaliser indépendamment l'une de l'autre pour

ainsi faciliter la programmation et mieux organiser l'application.

La conception et le fonctionnement de chacun de ses modules seront détaillés dans les chapitres qui suivent.

Chapitre 3

Conception :

3.1 Modélisation de la base de donnée

La conception de la base de donnée a été implémentée sous MySQL Workbench 8.0 CE. Cette base comprend 10 tables à savoir les tables : articles pour la gestion des articles, articles request pour la gestion des requêtes, category pour la gestion des catégories, type pour la gestion des types, members pour la gesticions des membres de l'association, donators pour la gestion des donateurs, receivers pour la gestion des bénéficiaires, deposit-sale pour la gestion des articles destinés à la vente, requests pour la gestion des requêtes, users pour la gestion des utilisateurs de l'application. Le diagramme entité association se présente alors comme suit :

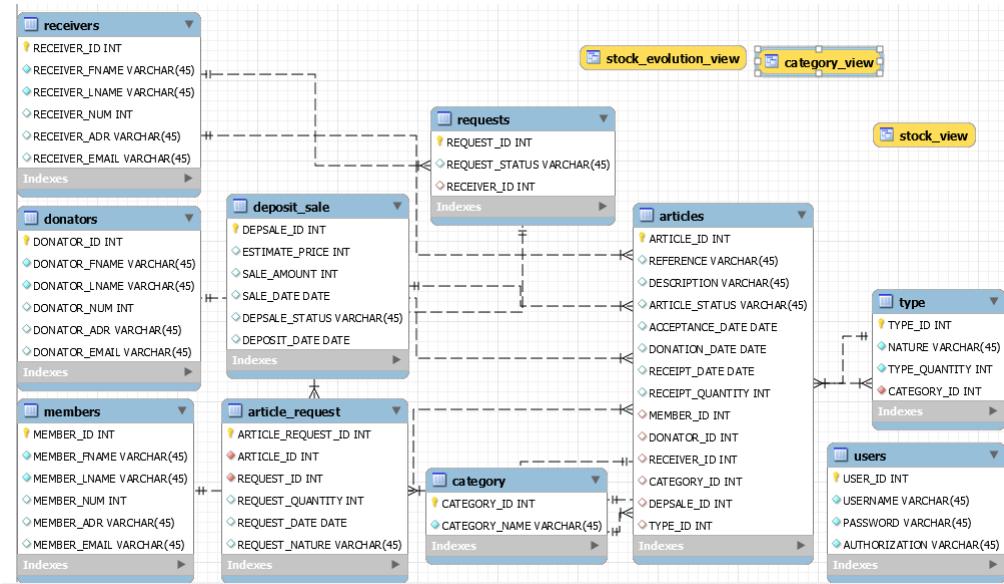


FIGURE 3.1 – Diagramme entité-association de la base

3.2 Identification des cas d'utilisation

Les principaux acteurs utilisant l'application sont : Les administrateurs et les membres de l'association. Les cas d'utilisations sont les rôles joués par les différents des acteurs.

3.2.1 Cas d'utilisation des administrateurs

Un administrateur de l'application doit pouvoir profiter des actions suivantes :

1. La gestion des utilisateurs de l'application
 - Consulter la liste des utilisateurs
 - Ajouter un nouvel utilisateur de l'application
 - Modifier les informations d'un utilisateur
 - Supprimer un utilisateur
 - Rechercher un utilisateur

2. La gestion des membres de l'association
 - Consulter la liste des membres
 - Ajouter un nouveau membre de l'association
 - Modifier les informations d'un membre
 - Supprimer un membre de l'association
 - Rechercher un membre

En plus de ces deux tâches spécifiques, un administrateur a accès à toutes les tâches des membres de l'association que ne verront dans la suite.

3.2.2 Cas d'utilisation des membres de l'association

Les membres(Personnels) de l'association doivent pouvoir profiter des actions suivantes :

1. La gestion des donateurs
 - Consulter la liste des donateurs
 - Ajouter un nouveau donneur à l'association
 - Modifier les informations d'un donneur
 - Supprimer un donneur
 - Rechercher un donneur
2. La gestion des bénéficiaires
 - Consulter la liste des bénéficiaires
 - Ajouter un nouveau bénéficiaire des services de l'association
 - Modifier les informations d'un bénéficiaire
 - Supprimer un bénéficiaire
 - Rechercher un bénéficiaire
3. La gestion des articles
 - Consulter la liste des articles
 - Ajouter un article
 - Mettre à jour les informations relatives à un article
 - Accepter/Refuser un article
 - Rechercher un article
4. La gestion du dépôt-vente
 - Consulter la liste des articles dans le dépôt-vente
 - Ajouter un article au dépôt-vente
 - Mettre à jour les informations relatives au dépôt-vente
 - Retourner les articles depuis le dépôt-vente vers le stock
5. La gestion des requêtes
 - Consulter la liste des requêtes
 - Ajouter une requête
 - Accepter/Refuser une requête
6. Les statistiques
 - Consulter le nombre de propositions de dons reçues
 - Consulter le nombre de donateurs et de bénéficiaires
 - Consulter le nombre de propositions de dons acceptées et le ratio acceptées/reçues par catégorie d'objet
 - Consulter le volume de dons ventilés suivant les deux destinations possibles
 - Consulter les principales catégories d'articles en stock
 - Observer l'évolution du stock
 - Consulter les statistiques sur les dons ayant transité par l'association

Chapitre 4

Réalisation :

4.1 Introduction

On passe maintenant à la partie de la réalisation où on met en réalité toute la conception précédente. On va présenter l'environnement de réalisation sur le plan logiciel et quelques interfaces de notre application.

4.2 L'environnement du travail

4.2.1 Environnement matériel

Nous mentionnons les caractéristiques de nos ordinateurs sur lesquelles nous avons développé l'application :

- 1^{er} Ordinateur :

- Marque : ASUS
- Processeur : Intel I7 – 7500U @ 2.7GHZ
- RAM : 8.00 Go

- 2^{ème} Ordinateur :

- Marque : ASUS
- Processeur : Intel I7 – 7200U @ 2.5GHZ
- RAM : 8.00 Go

- 3^{ème} Ordinateur :

- Marque : Lenovo
- Processeur : Intel I3 – 6100U @ 2.7GHZ
- RAM : 4.00 Go

- 4^{ème} Ordinateur :

- Marque : ASUS
- Processeur : Intel I7 – 7500U @ 2.7GHZ
- RAM : 8.00 Go

- 5^{ème} Ordinateur :

- Marque : ASUS
- Processeur : Intel I7 – 7500U @ 2.7GHZ
- RAM : 8.00 Go

4.2.2 Environnement logiciel

4.2.2.1 Système d'exploitation

Les 4 machines utilisent Microsoft Windows 10 comme système d'exploitation

4.2.2.2 Outils de développement

JDK 14 Le Java Développement Kit (JDK) désigne un ensemble de bibliothèques logicielles de base du langage de programmation Java, ainsi que les outils avec lesquels le code Java peut être compilé, transformé en byte code destiné à la machine virtuelle java.

4.2.2.3 Environnement de développement intégré

NetBeans :

NetBeans est un IDE (environnement de développement intégré) écrit en java, placé en open source. Il offre toutes les facilités d'un IDE moderne (éditeur avec coloration syntaxique, projets multi-langage, refactoring, éditeur graphique d'interfaces et de pages Web).

4.2.2.4 Outil d'administration de la base de données

L'implémentation de notre application se fera avec MYSQL sous l'environnement MySQL Workbench 8.0 (comme il est mentionné dans la partie conception). Il installe et configure automatiquement un environnement de travail complet sous Windows.

4.3 Mode de fonctionnement des interfaces élaborées

La plupart des interfaces possèdent la même structure : elles commencent par la connexion à la base des données, elles exécutent des requêtes SQL afin d'extraire les informations et elles se terminent par la fermeture de la connexion.

Afin d'extraire des informations nécessaires relatives à chaque formulaire, on a fait recours à plusieurs types de requêtes. On peut citer quelques types de requêtes utilisées :

- Sélection d'enregistrement
- Jointure sur des tables différentes
- Mise à jour des tables
 - Ajout
 - Suppression
 - Modification

La démarche établie lors de la demande d'édition d'un formulaire est expliquée par le diagramme suivant :

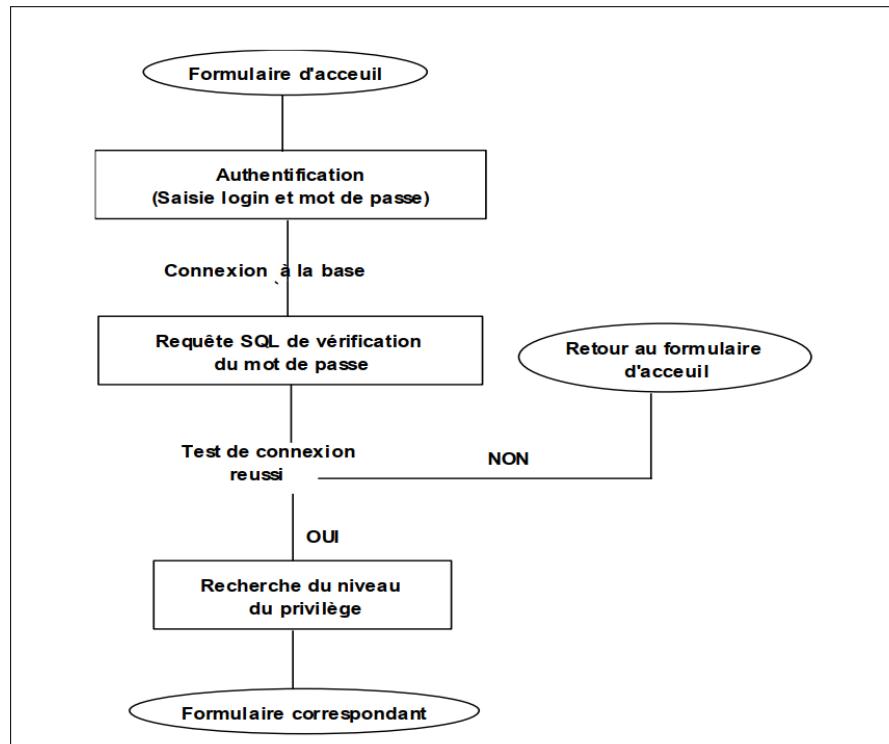


FIGURE 4.1 – Le chemin d'accès aux formulaires établis

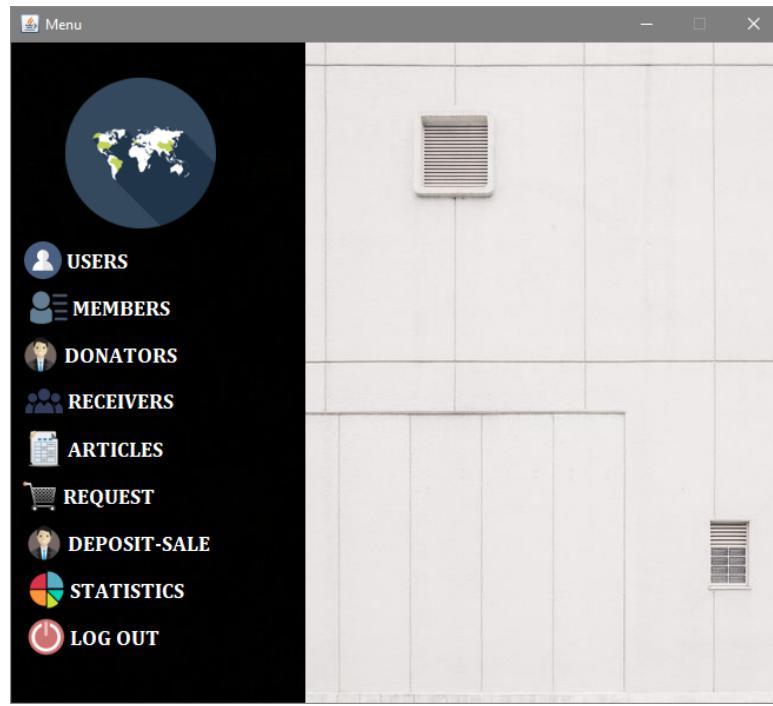


FIGURE 4.2 – Fenêtre des menus

Le diagramme de l'affichage des tableaux dans l'application

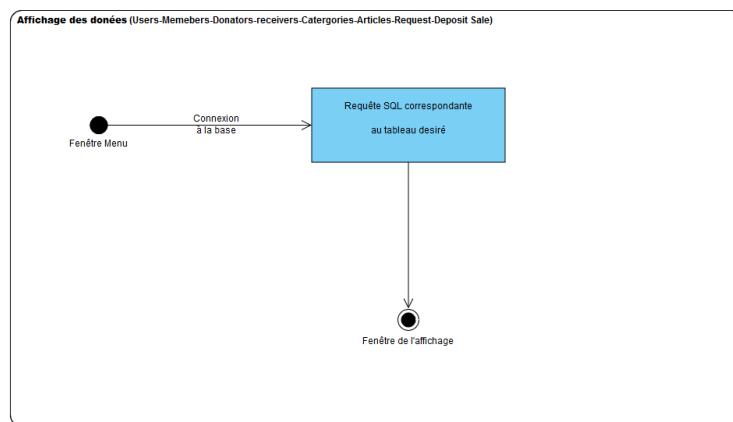


FIGURE 4.3 – Diagramme de l'affichage

4.3.1 Users

La gestion des utilisateurs se fait comme-ci

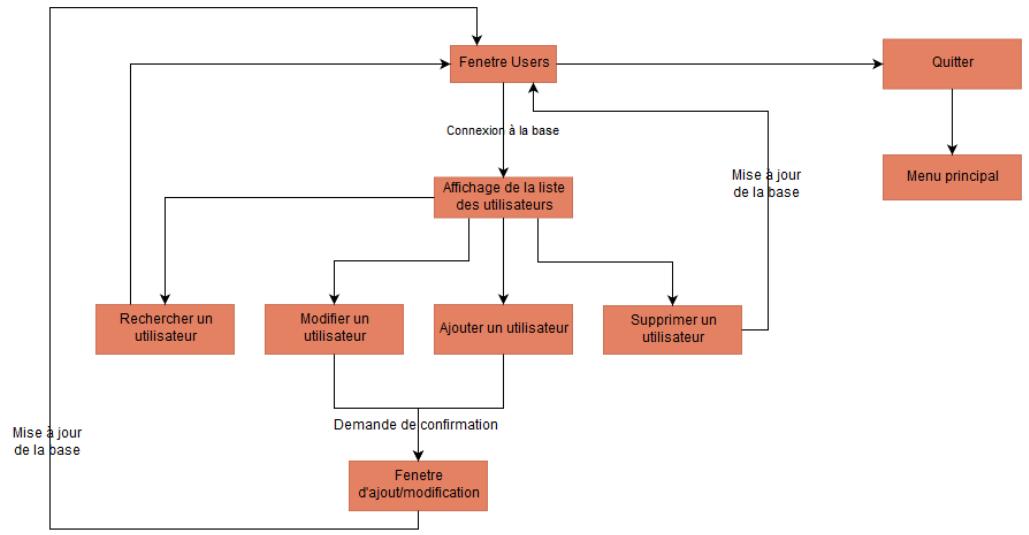


FIGURE 4.4 – Diagramme de la gestion des utilisateurs

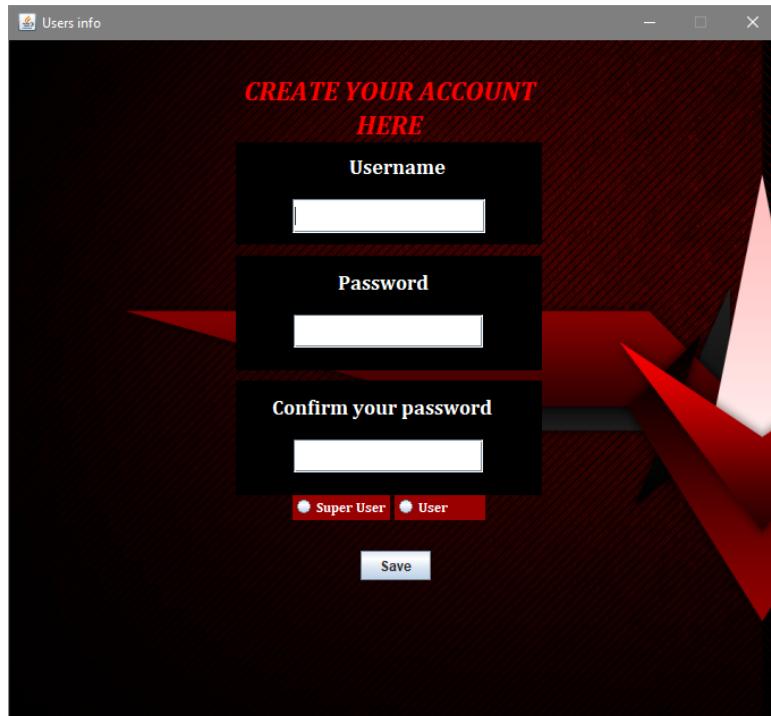


FIGURE 4.5 – Fenêtre de l'ajout d'un utilisateur

4.3.2 Members

La gestion des membres se fait comme-ci

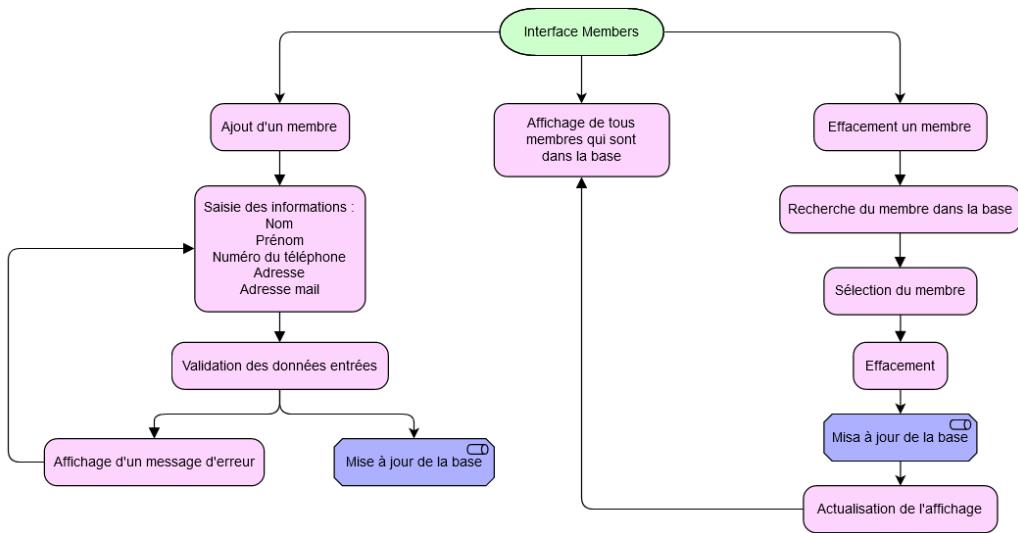


FIGURE 4.6 – Diagramme de la gestion des membres

La fenêtre Members affiche une liste de membres avec les colonnes ID, First name, Last name, Number, Adress et E-mail. Les données sont les suivantes :

| ID | First name | Last name | Number | Adress | E-mail |
|----|------------|-----------|-----------|----------------------|---------------------|
| 1 | Joseph | Antoine | 97848469 | 188 Dogwood Lane | JosephNAntoine@j... |
| 2 | Betty | Fobbs | 196021745 | 2123 Stonecoal Ro... | BettyFFobbs@jour... |
| 3 | Darryl | Beem | 462592762 | 3175 Geraldine Lane | DarrylJBeem@rhyt... |
| 4 | Alice | Sims | 72478230 | 2770 Arbor Court | AliceJSims@dayre... |

En bas de la fenêtre, il existe des champs pour saisir des informations (First name, Last name, Number, Adress, Email) et un bouton "Add Member" pour ajouter un nouveau membre.

FIGURE 4.7 – Fenêtre Members

4.3.3 Donators

La gestion des donneurs se fait comme-ci

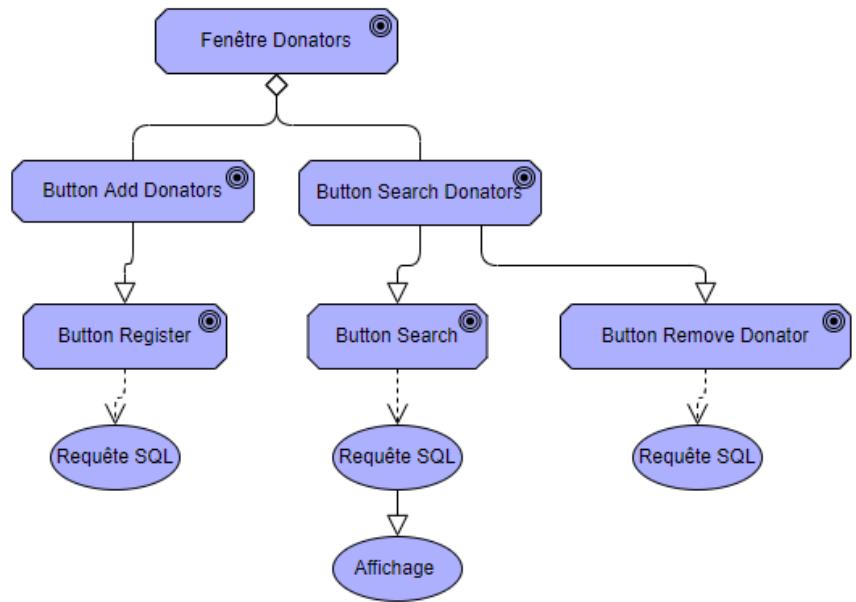


FIGURE 4.8 – Diagramme de la gestion des donneurs

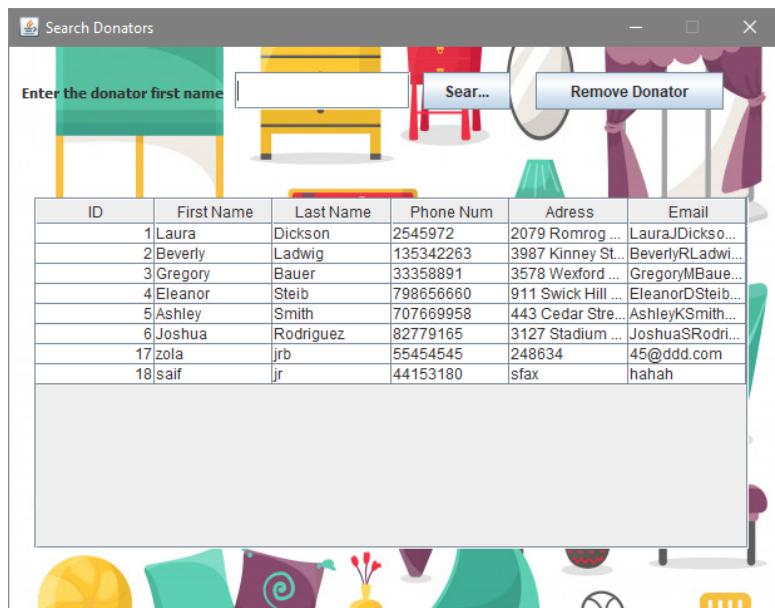


FIGURE 4.9 – Fenêtre DONATORS

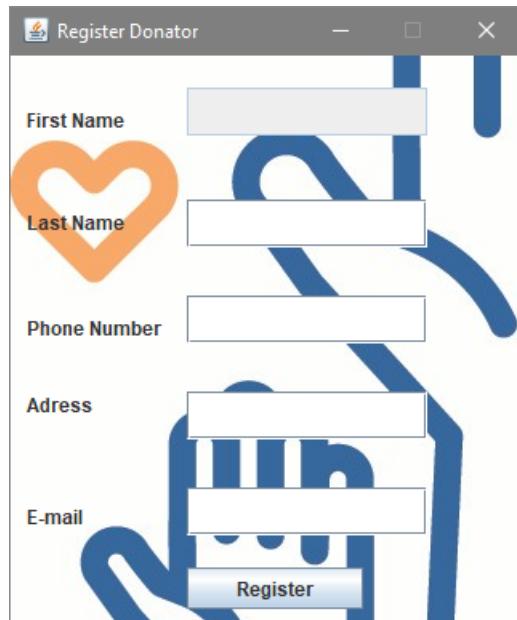


FIGURE 4.10 – Ajout donneur

4.3.4 Receivers

L'utilisateur pourrait consulter les receveurs depuis cette fenêtre

| RECEIVER_ID | RECEIVER_FNAME | RECEIVER_LNAME | RECEIVER_NUM | RECEIVER_ADR | RECEIVER_EMAIL |
|-------------|----------------|----------------|--------------|-------------------------|-----------------------|
| 1 | Kevin | Villalobos | 789621032 | 1673 White Lane | KevinBVillalobos@r... |
| 2 | Frank | Halcomb | 125084588 | 4508 Sardis Station | FrankRHalcomb@r... |
| 3 | Arthur | James | 563586887 | 2411 Jerome Avenue | ArthurMJames@tel... |
| 4 | Mary | Parish | 34055265 | 2256 Still Pastures ... | MaryTParish@jourr... |

FIGURE 4.11 – Fenêtre RECEIVERS

4.3.5 Articles

La gestion des articles se fait comme-ci

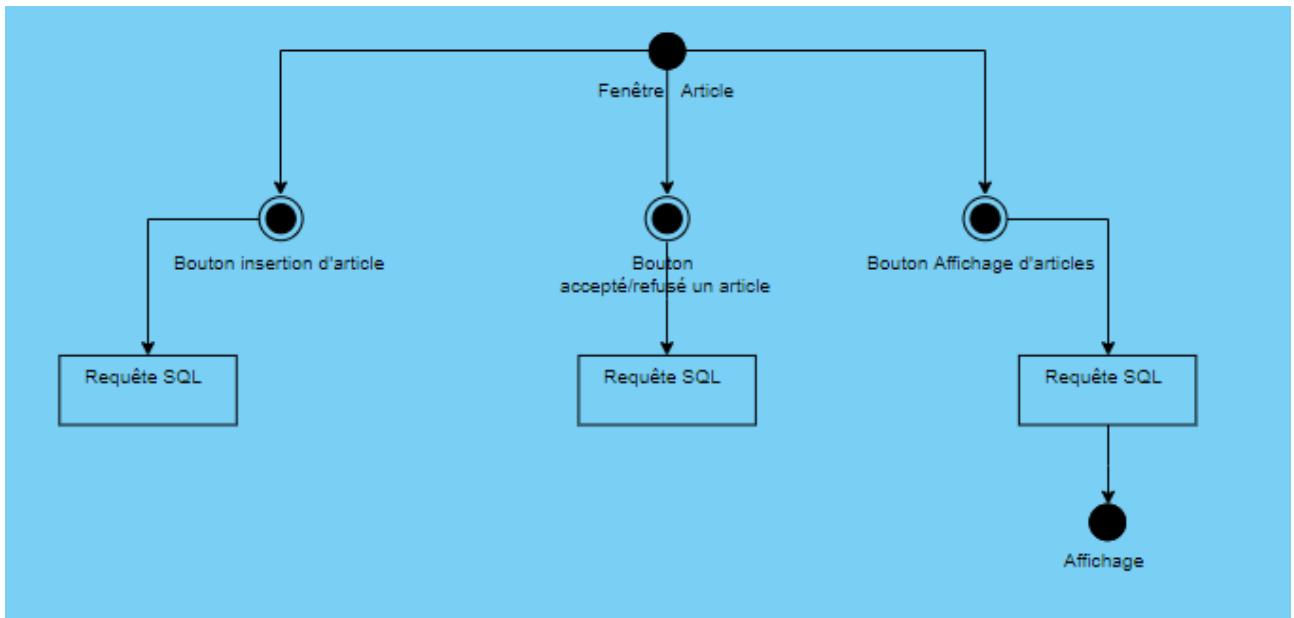


FIGURE 4.12 – Diagramme de la gestion des articles

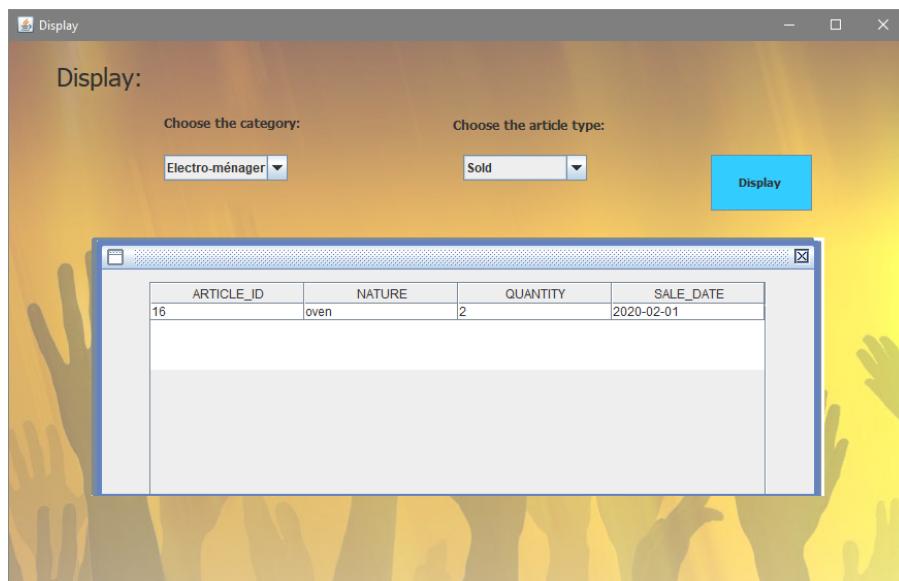


FIGURE 4.13 – Fenêtre ARTICLES

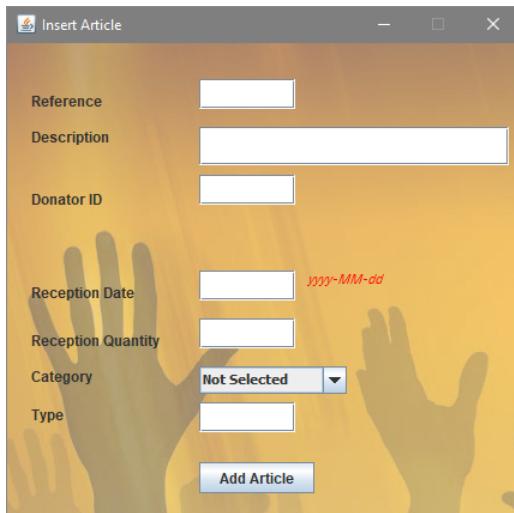


FIGURE 4.14 – Fenêtre d'ajout des articles

4.3.6 Request

La gestion des demandes se fait comme-ci

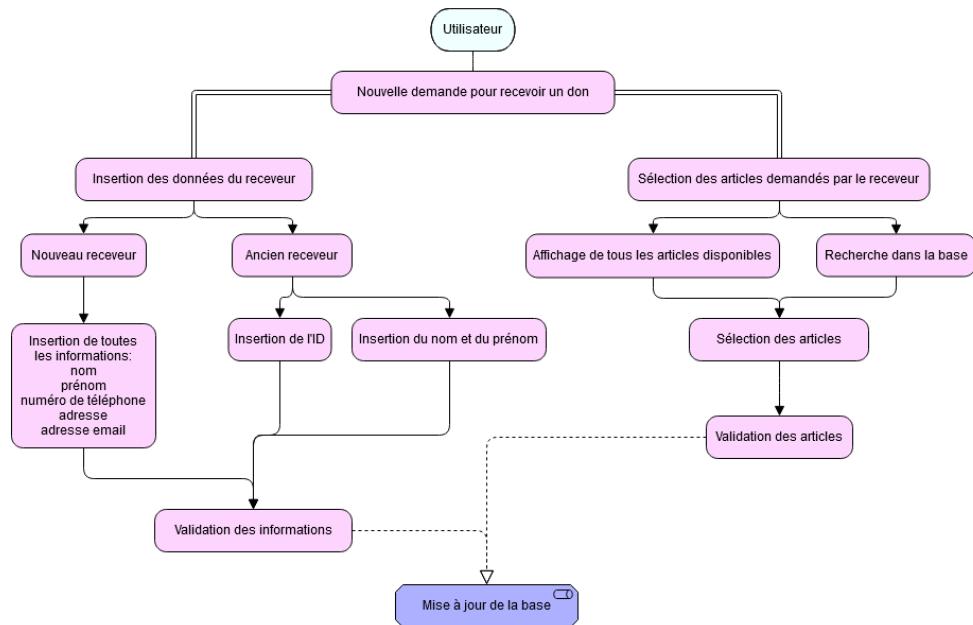


FIGURE 4.15 – Diagramme de la gestion des demandes

The screenshot shows a Windows application window titled "Requests". At the top, there is a toolbar with a "Show" dropdown set to "All", and buttons for "Refresh" and "Add". Below the toolbar is a table with the following data:

| Request ID | Rvr. Name | Rvr. Last Na... | Request Date | Request Sta... |
|------------|-----------|-----------------|--------------|----------------|
| 1 | Kevin | Villalobos | 2020-06-23 | InProgress |
| 2 | Frank | Halcomb | 2020-06-23 | InProgress |
| 3 | Arthur | James | 2020-06-23 | InProgress |
| 4 | Mary | Parish | 2020-06-23 | Refused |

FIGURE 4.16 – Fenêtre REQUEST

Les détails de la demande peuvent être consultés en cliquant sur la demande dans le tableau puis il décide l'accepter ou la refuser ou la remettre en attente

The screenshot shows a window titled "Request Informations". At the top, there are three buttons: "Accept" (highlighted in yellow), "Refuse", and "In Progress". Below the buttons is a table with the following data:

| Description | Quantity | Date | Don. Name | Don. Last Name |
|-------------|----------|------------|-----------|----------------|
| wood chair | 2 | 2020-06-23 | Laura | Dickson |

L'ajout d'une nouvelle demande avec le Bouton "Add" et en remplissant les informations

The screenshot shows a window titled "New donation request". The main title is "New donation Request". It has two sections: "Receiver information" and "Available articles".

Receiver information: Fields include "ID" (empty), "First name" (empty), "Last name" (empty), "Address" (empty), "e-Mail" (empty), and "Number" (empty).

Available articles: A table showing available items:

| ID | Type | Quantity |
|----|------|----------|
| 19 | desk | 1 |

Requested articles: A table showing requested items:

| ID | Type | Quantity |
|----|------|----------|
| | | 1 |

Buttons include "Search", "Add request", and quantity adjustment buttons (">>>", "<<<").

FIGURE 4.17 – Fenêtre New donation request

4.3.7 Deposite-Sale

La gestion de Dépôt-Vente se fait comme-ci

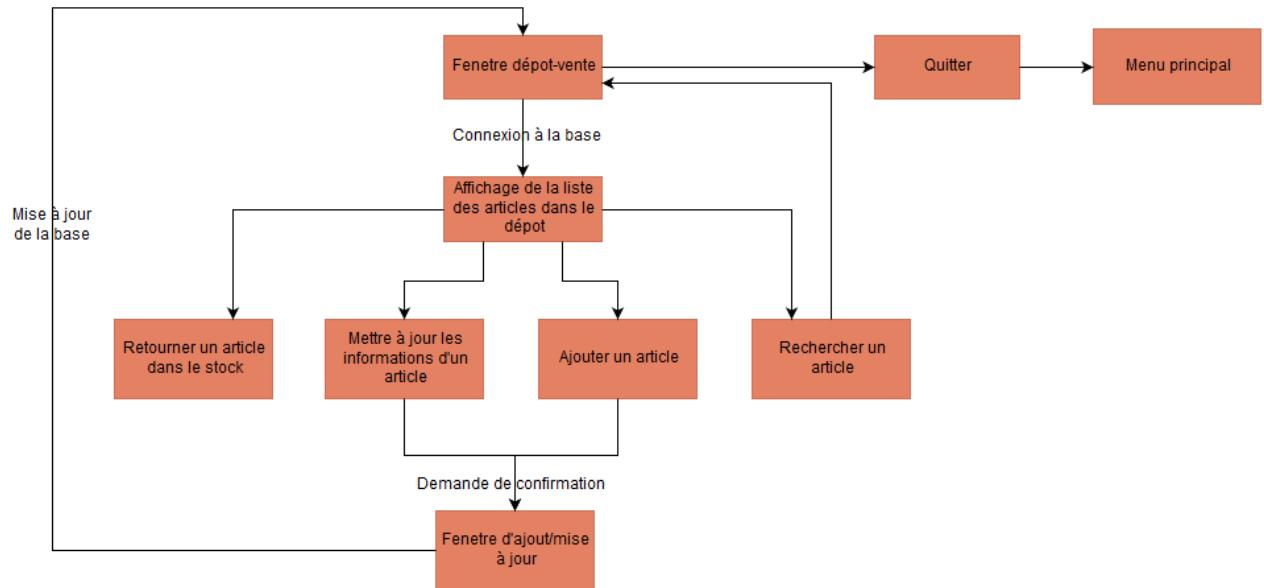


FIGURE 4.18 – Diagramme de la gestion de Dépôt-Vente

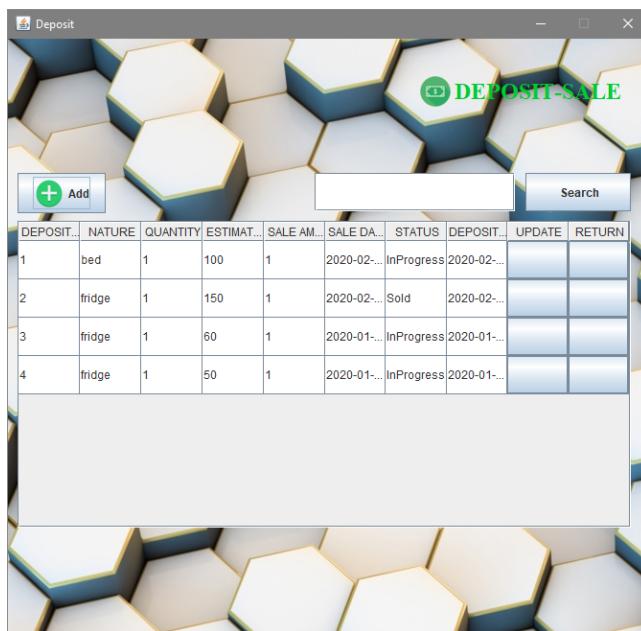


FIGURE 4.19 – Fenêtre Deposite-Sale

L'utilisateur peut retourner les articles en dépôt avec le bouton "RETURN" ou bien modifier ses informations avec le bouton "UPDATE" en saisissant les informations requises ou ajouter un nouveau dépôt

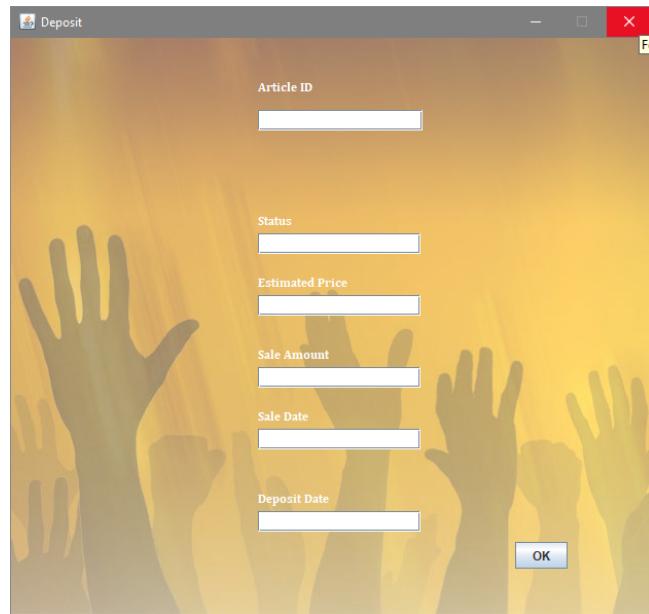


FIGURE 4.20 – Fenêtre Deposite-Sale

4.3.8 Statistics

L'utilisateur peut consulter l'activité de l'association a partir de la fenêtre STATISTICS

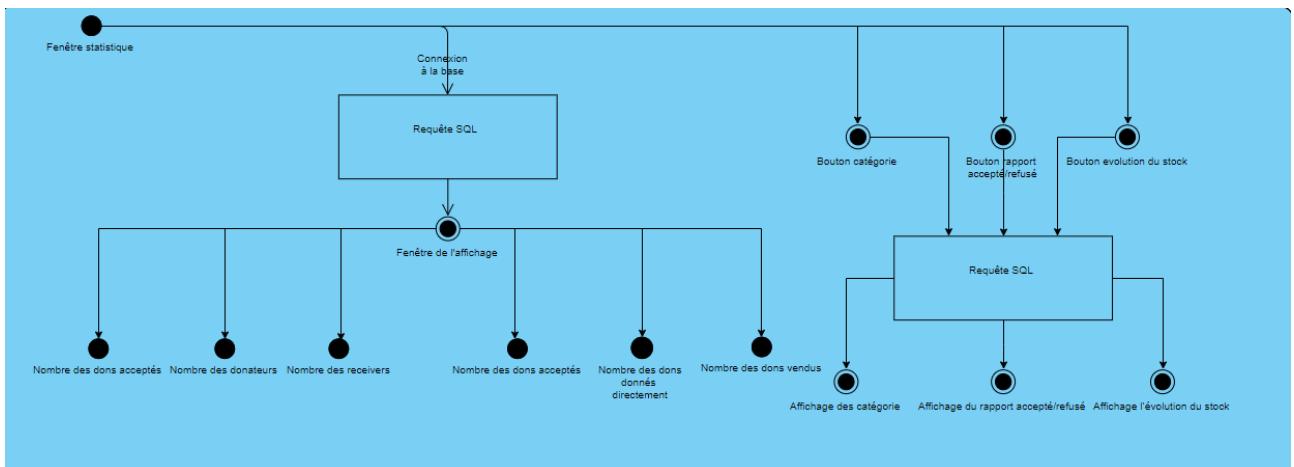


FIGURE 4.21 – Diagramme des statistiques

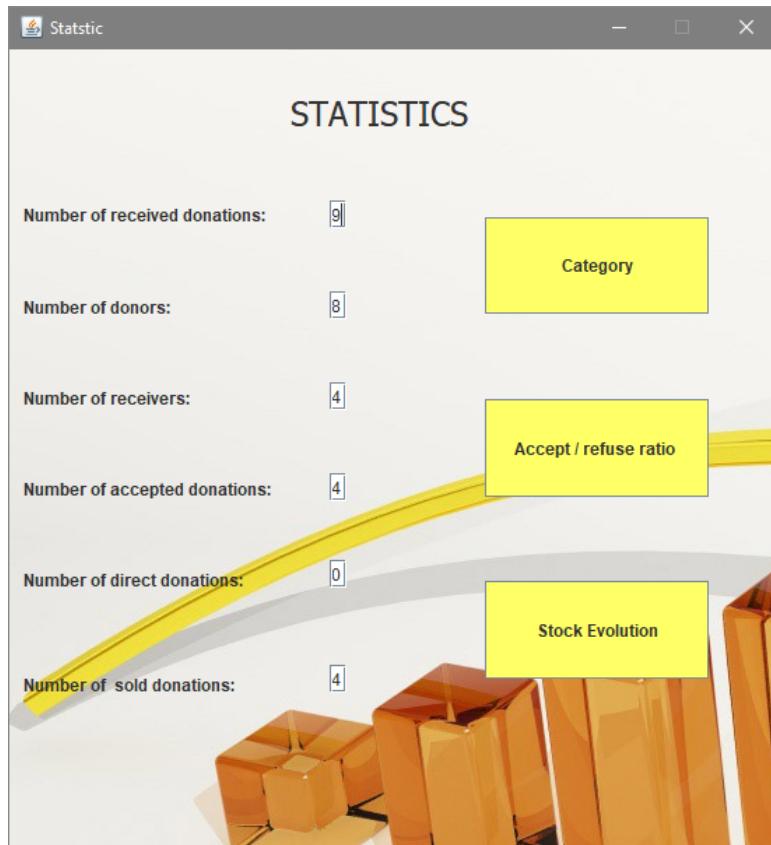


FIGURE 4.22 – Fenêtre Statistics

4.4 Conclusion

A la suite de l'implantation de l'application dont quelques procédures ont été détaillées plus haut, on peut dire que JAVA (Programmation Orienté Objet généralement) a rendu tout ce qu'on veux en réalité avec sa large bibliothèque.

Chapitre 5

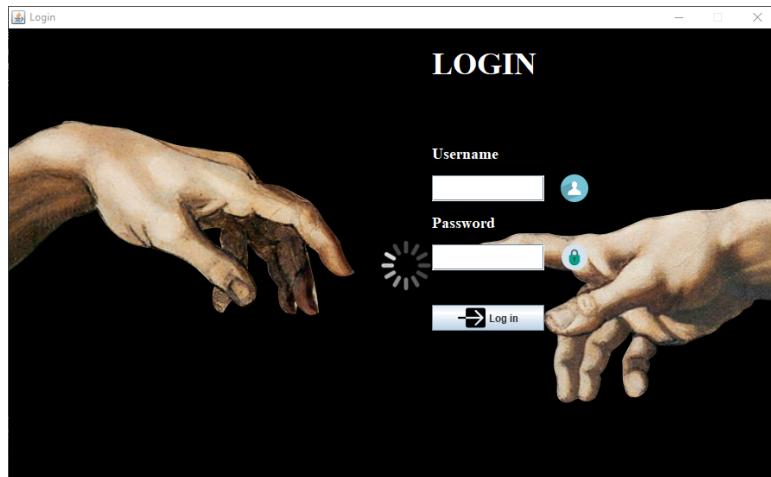
Tests et Validation

5.1 Introduction

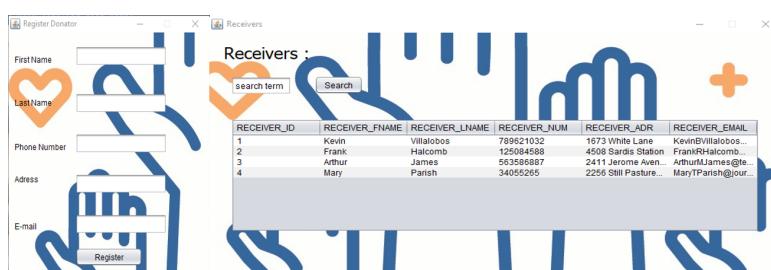
Dans cette partie on va tester et valider toutes les fonctionnalités de notre application et expliquer le processus qui doit être effectuer par l'utilisateur afin de mieux comprendre l'usage de l'application.

5.2 Tests

Tout d'abord il faut faire le login à partir de l'utilisateur si il est déjà registré sinon il faut registrer avant. En effet, on deux types des utilisateurs : un Utilisateur normale et un super utilisateur qui a plus de liberté et des options lors de son usage de l'application par exemple il peut supprimer d'autre utilisateur ou bien les accéptés voici alors un exemple :



Maintenant on va parler de l'ajout des donators et des members. Pour cela on a crée trois intrefaces graphiques qui font l'ajout à partir des données personnelles comme le first name, last name, phone number,NIC number ,adress ,e-amil... voici un exemple des interfaces réalisées :



Aprés avoir ajouter les dontors et les members nous avons crée des interfaces pour afficher les données associé à chaque personne et faire le recherche rapide.voici un exemple :

Members

| | First name | Last name | Number | Adress | E-mail |
|---|------------|-----------|-----------|---------------------|--------------------|
| 1 | Joseph | Antoine | 97848469 | 188 Dogwood Lane | JosephNAntoine... |
| 2 | Betty | Fobbs | 196021745 | 2123 Stonecoal R... | BettyFFobbs@jou... |
| 3 | Darryl | Beem | 462592762 | 3175 Geraldine L... | DarrylJBeem@rhy... |
| 4 | Alice | Sims | 72478230 | 2770 Arbor Court | AliceJSims@day... |

First name Last name Number Adress Email Add Member

Passant maintenant à l'ajout des articles, on a crée une interface graphique à partir de laquelle les members peuvent ajouter les données d'un article récupérer à partir de donateur directement puis il peut ,dans la même interface, préciser le statut des articles et de plus le member peut consulter la liste des articles et leurs statut correspondant et faire une recherche rapide dans cette liste. voici un exemple :

Insert Article

Display:

Reference:

Description:

Donator ID:

Reception Date: yyyy-MM-dd

Reception Quantity:

Category: Not Selected

Type:

Add Article

Choose the category: Electro-ménage

Choose the article type: Sold

Display

ARTICLE_ID NATURE QUANTITY SALE_DATE

| | | | |
|----|------|---|------------|
| 16 | oven | 2 | 2020-02-01 |
|----|------|---|------------|

Pour Attribuer les articles aux receivers, nous avons crée une autre interface graphique pour donner le choix aux members de l'association d'accepter ou de refuser les requests des receivers voici un exemple :

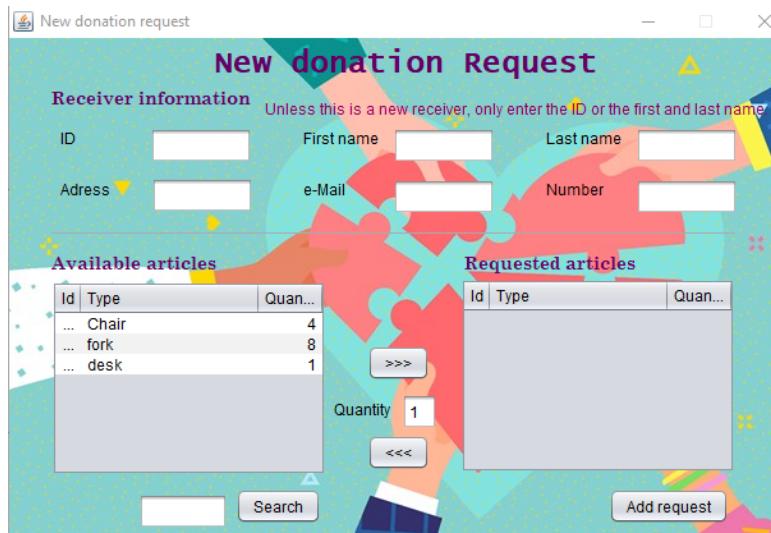
Accept/Refuse Article

| ARTICLE_ID | DESCRIPTION | DONATOR_ID | RECEIPT_QU... | NATURE |
|------------|-------------|------------|---------------|--------|
| 12 | wood chair | 1 | 4 | Chair |
| 18 | small forks | 4 | 8 | fork |
| 19 | small desk | 3 | 1 | desk |

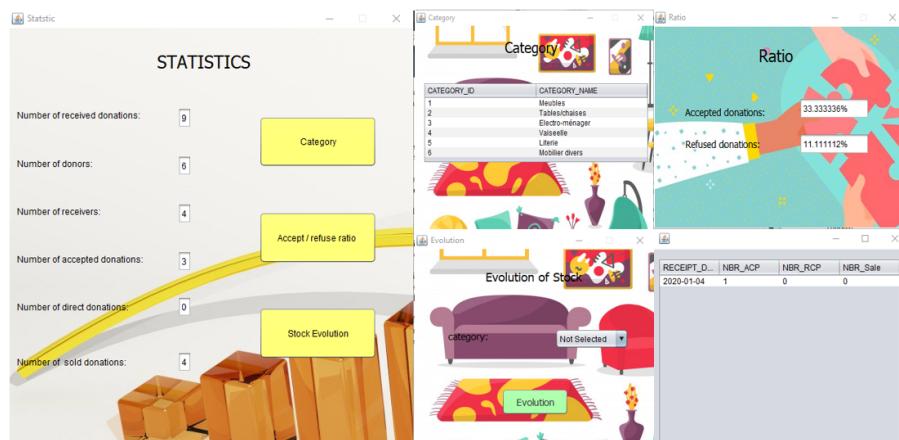
Accept Refuse

Membres ID:

Maintenant pour ajouter des receivers et des nouvelles requests on a crée une interface graphique qui à la fois permet d'ajouter des nouvelles requests et receivers. voici l'exemple suivant :



Pour rendre plus efficace et rapide les calculs statistiques on a crée une interface graphique qui permet de connaître le nombre des donneurs, le nombre des receivers, le nombre des dons récus, le nombre des dons acceptés, le nombre des dons attribuées aux receivers et le nombre des dons vendus. Cette interface permet aussi de voir les catégories des dons de l'association ainsi que l'affichage des nombres des dons acceptés, vendus et attribuer chaque jour avec les dates correspondantes et de plus connaitre le spourcentage des dons acceptés et refusés. L'exemple suivant explique la fonctionnalité de cette interface graphique :

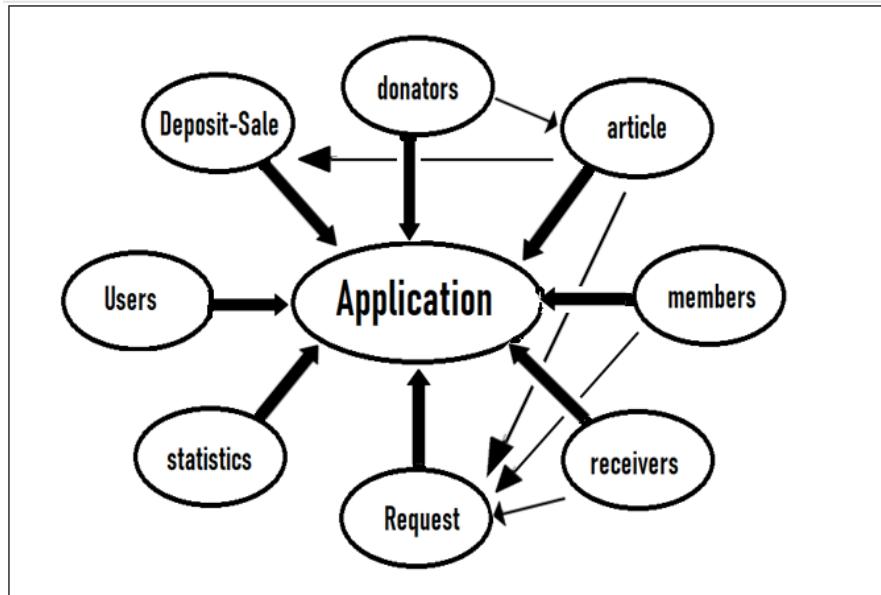


Enfin nous avons crée une interface graphique qui gère les dons mis à vente, à partir de cette interface on peut consulter tous les dons qui ont été mis à vente ainsi que modifier les paramètres associés tel que le prix par exemple ou ajouter un nouveau don à vendre. Voici enfin un exemple :

| DEPOS... | NATURE | QUANT... | ESTIMA... | SALE A... | SALE D... | STATUS | DEPOS... | UPDATE | RETURN |
|----------|--------|----------|-----------|-----------|-----------|------------|-----------|--------|--------|
| 1 | bed | 1 | 100 | 1 | 2020-0... | Returned | 2020-0... | | |
| 2 | fridge | 1 | 150 | 1 | 2020-0... | Sold | 2020-0... | | |
| 3 | fridge | 1 | 60 | 1 | 2020-0... | InProgr... | 2020-0... | | |
| 4 | fridge | 1 | 50 | 1 | 2020-0... | InProgr... | 2020-0... | | |

5.3 Intégration

Les parties relatives à notre travail,gestion de stock(les dons),sont enfin regroupées ensemble pour construire une application permettant de faciliter la gestion des dons pour les associations de bienfaisance.



5.4 Conclusion

A la fin de ce chapitre, il est important de signaler l'importance des tests unitaires et les tests effectués lors de la validation pour la correction des erreurs qui peuvent être commises lors de la conception et pour éviter les fautes de saisie et de choix faite par l'utilisateur de l'application.

Nous notons aussi que la partie intégration est une tâche délicate et nécessite beaucoup de soins pour assurer le bon fonctionnement des différents modules de l'application interagissant entre eux.

Chapitre 6

Conclusion générale

En quelques mots, ce mini projet de POO nous a permis de nous mettre en situation, d'évaluer le problème, de l'analyser pour finalement trouver une solution. Et bien que ça été une première approche avec l'orientée objet et avec Java, nous nous sommes vite rendus compte des avantages que ça pouvait nous offrir : une représentation structurée du problème avec des classes indépendantes.