Documentation Projet POA

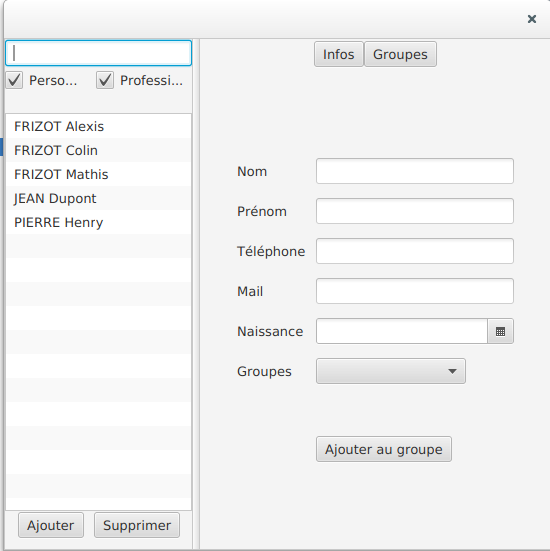
**Contexte :**Notre application est une application permettant la gestion simple et rapide d’un répertoire. Ce répertoire est composé de groupes eux-mêmes composés de contacts qui peuvent êtres des contacts personnels ou des contacts professionnels. Grâce à ses deux types de contacts, l’utilisateur peut choisir de n’afficher que les contacts du type qu’il souhaite selon l’environnement dans lequel il se trouve pour diminuer la liste à afficher. Ainsi, il peut retrouver plus rapidement les contacts professionnels lorsqu’il travail et les contacts personnels en dehors.

Le répertoire étant modifiable il est possible de créer de nouveaux groupes, d’ajouter de nouveaux contacts et de gérer la liste des groupes auxquels appartient un contact.

Par ailleurs, comme tout bon répertoire, chaque contact possède des données modifiables à tout moment. Ainsi, l’utilisateur peut ajouter ou supprimer des numéros de téléphone à un contact afin de ne jamais oublier ses coordonnées. De plus, le contact possédant une date de naissance il est possible de la vérifier afin de ne pas louper son anniversaire.

Enfin, il est possible de rechercher un contact par son nom, prénom, numéro de téléphone ou adresse mail grâce à la barre de recherche prévu à cet effet. Il est donc très simple de retrouver un contact même lorsque la liste est extrêmement grande.

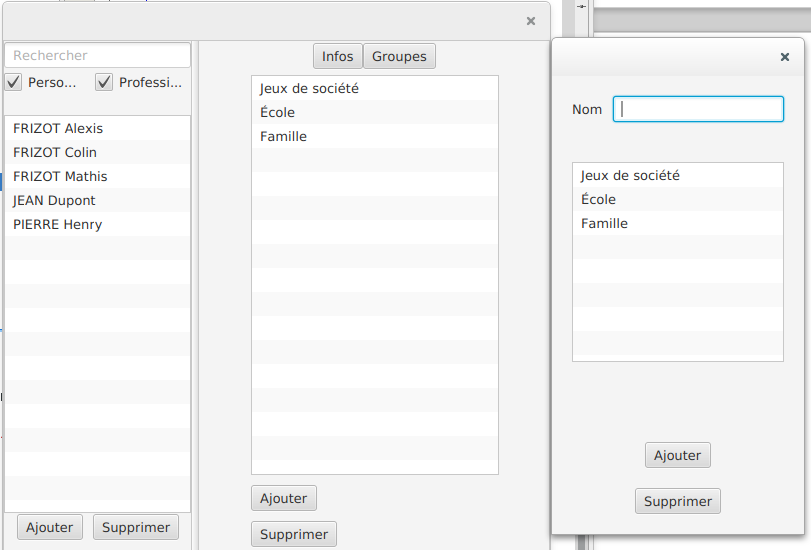
**Sketchs :**

****

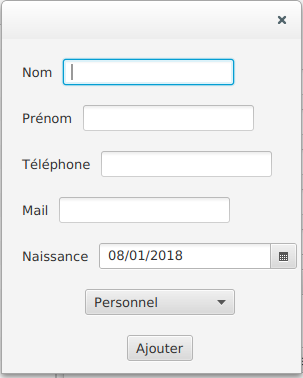
Lors de l'ouverture de l'application nous obtenons la page suivante. Cette page d'accueil permet l'affichage des contacts professionnels et personnels à gauche. Puis dans la partie droite l'onglet "Infos" est préalablement sélectionné et permet l'affichage des informations générales du contact (nom, prénom, téléphones, adresse e-mail, date de naissance). L'onglet "Groupes" permet quant à lui, comme son nom l'indique, de gérer les groupes.

Sur cette page nous avons 5 actions possibles :

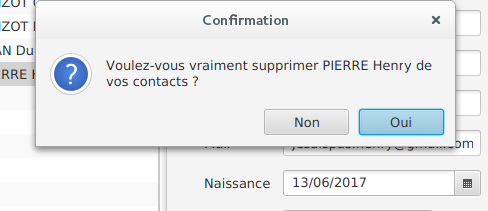
* Sélectionner le type de contacts à afficher grâce aux deux CheckBox (en haut à gauche).
* Ajouter une personne au répertoire (en bas à gauche).
* Supprimer une personne du répertoire (en bas à gauche).
* Supprimer la personne sélectionnée du groupe sélectionné (partie droite, onglet "Groupes").
* Afficher une seconde fenêtre permettant la gestion des groupes (partie droite, onglet "Groupes").



Par la suite, j'ai cliqué sur l'onglet "Groupes", qui nous affiche une liste des différents groupes auxquels appartient le contact. Puis cliqué sur le bouton ajouter (maintenant "Gestion des groupes") qui nous affiche la fenêtre de droite. En écrivant un nom de groupe et en cliquant sur le bouton "Ajouter", on ajoute le nouveau groupe à la liste des groupes de notre répertoire. En sélectionnant un groupe et en cliquant sur "Supprimer", on le supprimer de la liste des groupes de notre répertoire.

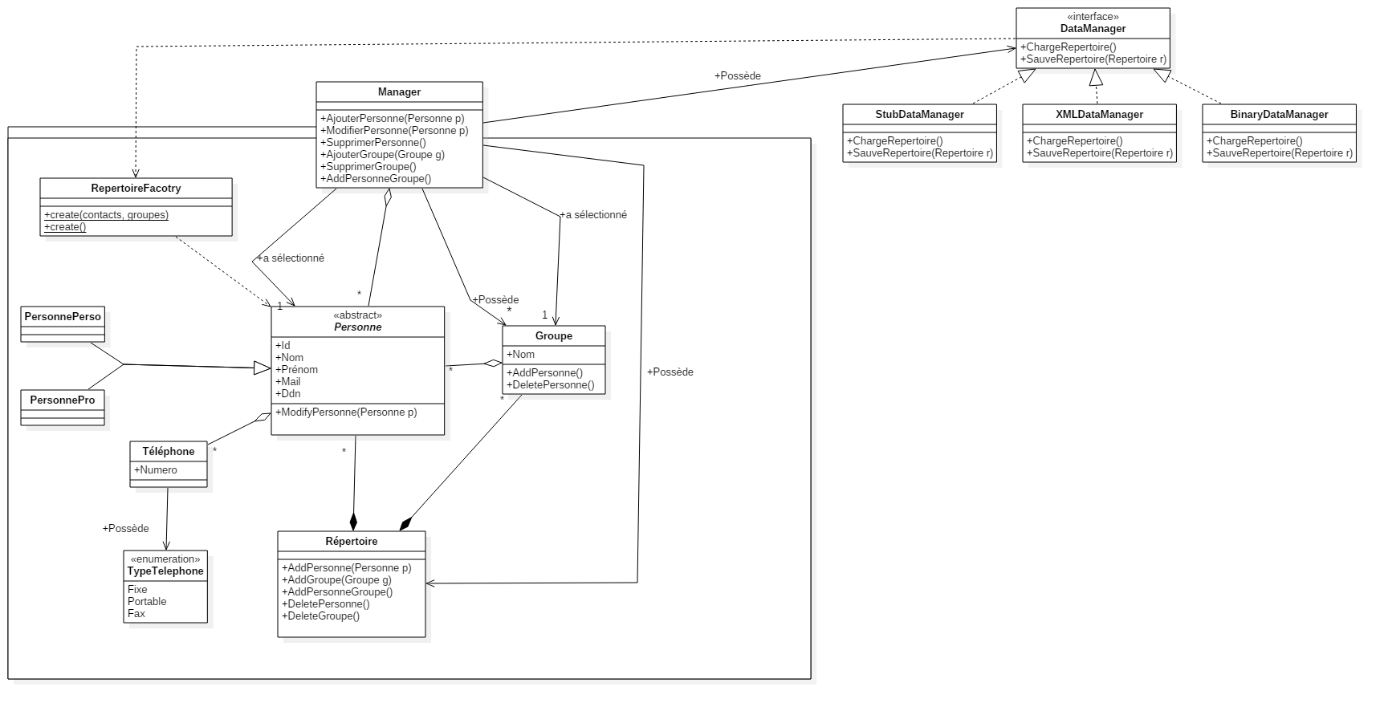


En cliquant sur le bouton ajouter (situé en bas à gauche de la page d'accueil) qui je le rappelle nous permettra d'ajouter une personne au répertoire, Nous tombons sur une fenêtre avec des champs vides et un bouton "Ajouter" qui après avoir rempli les champs voulus nous permettra d'ajouter cet ami à la liste des contacts de notre répertoire.

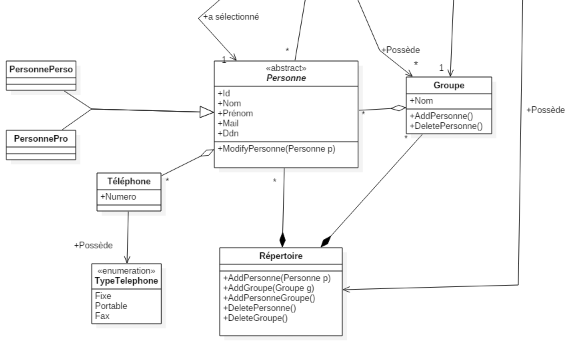


Si nous ne voulons plus de cet ami, il nous suffit de le sélectionné puis de cliquer sur le bouton "Supprimer" en bas à gauche. Un message de confirmation apparaîtra et permet de vérifier le souhait de l'utilisateur quant à la suppression d'un contact afin d'éviter les éventuelles erreurs qui peuvent arrivées.

**UML :**

****

Voici le diagramme de classe correspondant à notre application.

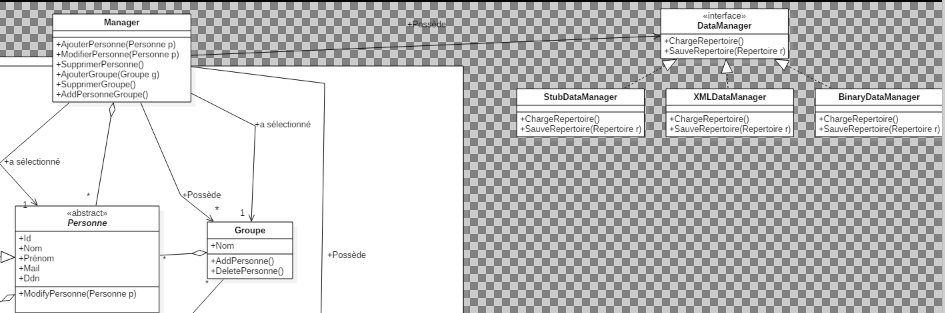


Un répertoire possède une liste de groupes et une liste de contacts. Il est possible d’ajouter et de supprimer des groupes et des contacts ainsi que d’ajouter un contact à un groupe.

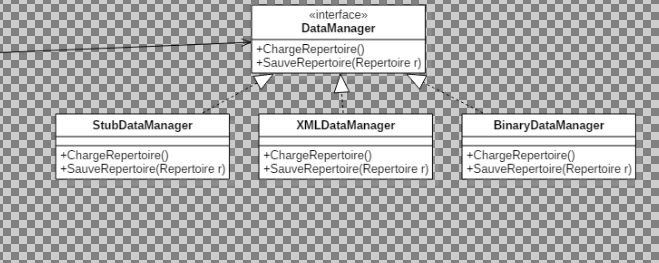
Un groupe est constitué d’un nom afin de le reconnaître ainsi que d’une liste de contacts désignant les contacts appartenant à ce groupe.

Une personne (un contact) peut être de plusieurs types : personnel ou professionnel. Permettant de trier les contacts en fonction de ce que l’utilisateur souhaite afficher.  
Une personne possède un nom, un prénom, une adresse e-mail, une date de naissance et une liste de téléphones lui appartenant.

Un téléphone possède quant à lui un numéro de téléphone en chaîne de caractères et un type de téléphone parmi l’énumération : fixe, portable, fax.



La classe Manager permet de faire le lien entre le Model et la Vue. Elle représente le patron de conception de la Façade et permet de restreindre l’accès au sous-système. De plus, le manager possède un DataManager qui est une interface permettant au manager le chargement et la sauvegarde des données qu’importe la façon utilisée (Stub, XML, Binaire).



Enfin, les classes StubDataManager, XMLDataManager et BinaryDataManager sont des classes implémentant l’interface DataManager et permettant le chargement et la sauvegarde des données d’une manière bien spécifique selon la classe.

**Patron de conception :** Nous avons utilisé le patron de conception Patron de méthode entre la classe abstraite Personne et les classes concrètes PersonnePerso et PersonnePro afin de factoriser les appels de méthodes. Cela permet aussi contrairement à notre idée de départ (qui était de mettre un booléen) de pouvoir dans les avancées futures créer d’autres types de personnes non modifiables par l'utilisateur. Il aurait aussi pu être possible de faire en sorte que personnel et professionnel soient des groupes.

Par ailleurs, nous utilisons le patron de conception de la fabrique simple pour contrôler la création de Répertoire. Pour cela, la classe RepertoireFactory possède deux méthodes statiques "create" prenant des paramètres différents afin de choisir la façon de créer le répertoire. L'une prend une liste de contacts et une liste de groupes afin de créer un répertoire possédant des données dès le départ, l'autre ne prend rien pour créer un répertoire de base.

Dans les avancées futures nous aurions voulu qu’un groupe possède des sous-groupes afin que nos contacts appartenant au groupe "Foot" et au groupe "Marathon" appartiennent tous au groupe "Sport". Pour cela, nous aurions utilisé le patron de conception composite.

**Diagramme de cas d'utilisation :**