



RAPPORT

Jean-baptiste Loirot, Thomas Combarieu, Abderrahim
Khantouch, mathis

AP4 Amset_admin

Table des matières

Table des matières

Présentation générale	1
Contexte initial :	2
🧩 BESOIN INITIAL	2
✅ Solution proposée	2
Plan du site	3
Maquette du site	4
MCD	6
Descriptif technique de l'application	7
Moyens techniques :	7
Organisation du code	8
Guide rapide d'utilisation	9
Descriptif des tâches	14
Diagramme de Gantt	15
Difficultés rencontrées	16
Une conclusion	17

Présentation générale

Contexte initial :

Amset est une SSII (Société de Services et d'Ingénierie en Informatique). Elle répond aux besoins de ses clients (organisations privées ou publiques) dans le domaine des nouvelles technologies et de l'informatique.

BESOIN INITIAL

Amset souhaite créer une application back-office en client lourd Java Swing, en travaillant sur la partie « gestions des utilisateurs » et ainsi permettre à un administrateur ayant accès à cette application de pouvoir :

- Voir la liste des clients
- Ajouter un client
- Modifier un client
- Supprimer un client

Solution proposée

Afin de répondre au besoin exprimé par Amset, il est proposé de développer une application back-office en **Java Swing** permettant à un administrateur de gérer les clients.

L'application sera structurée selon une architecture **MVC (Modèle-Vue-Contrôleur)**,

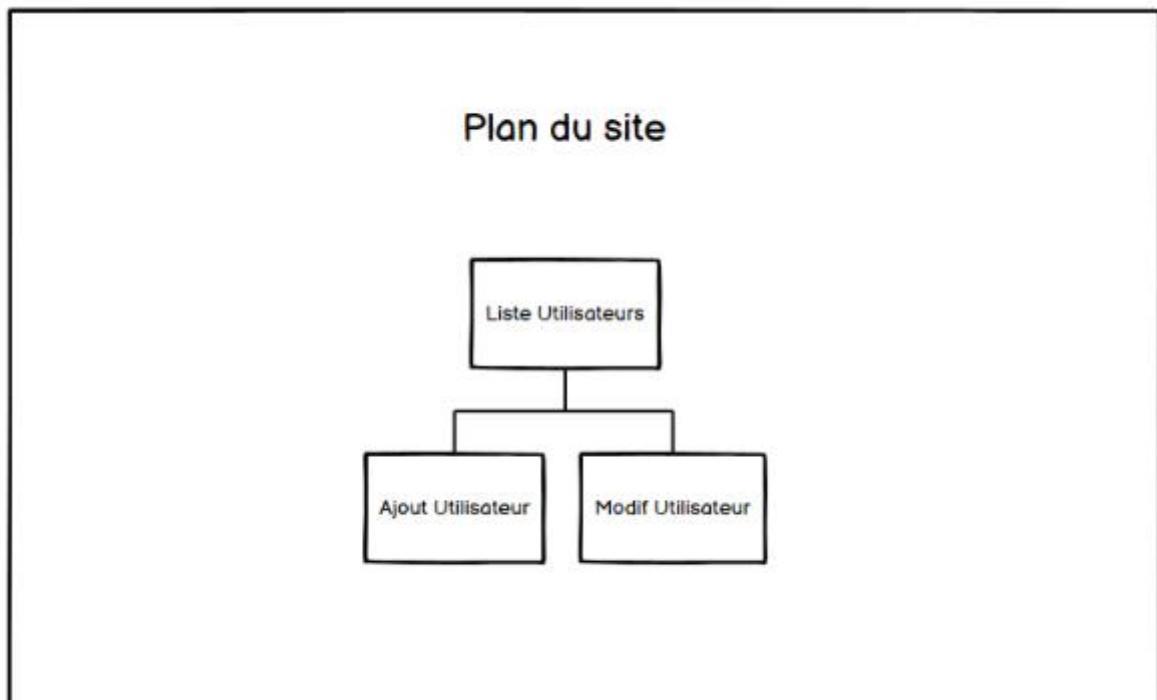
L'interface permettra à l'administrateur de :

- **Consulter** la liste des clients via une table affichant les informations principales.
- **Ajouter** un nouveau client à l'aide d'un formulaire dédié.
- **Modifier** les informations d'un client existant après sélection dans la liste.
- **Supprimer** un client avec confirmation préalable.

Les données seront stockées dans une **base de données relationnelle**, et les opérations **CRUD** (Create, Read, Update, Delete) seront effectuées.

Plan du site

Voici le plan du site Amset avec tous les pages qui seront disponible pour la version finale du site



Maquette du site

Liste Utilisateur

Nom	Prenom	Mail
Combarieu	Thomas	t.combarieu@gmail.com
Loirot	Jean-baptiste	Jb@gmail.com
Khantouch	Abderrahim	Abderrahim@hotmail.fr
Xiong	Teddy	Teddy74@gmail.com

Ajouter

Modifier

Supprimer

Ajout Utilisateur

Nom :

Prenom :

Mail :

Identifiant :

Mot de passe :

Ajouter

Annuler

Modif Utilisateur

Nom :

Prenom :

Mail :

Identifiant :

Mot de passe :

Modifier

Annuler

Liste Utilisateur

Nom	Prenom	Mail
Combarieu	Thomas	t.combarieu@gmail.com
Loirot	Jean-baptiste	Jb@gmail.com
Khantouch	Abderrahim	Abderrahim@hotmail.fr
Xiong	Teddy	Teddy74@gmail.com

Confirmation de Supression

! Voulez-vous supprimer cet utilisateur ?

Yes Non Cancel

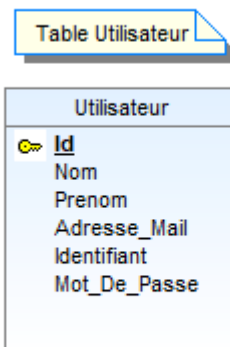
Ajouter

Modifier

Supprimer

MCD

La table UTILISATEUR permet de stocker les informations essentielles des utilisateurs, telles que leur nom, prénom, adresse mail, identifiant et mot de passe, avec un identifiant unique auto-incrémenté pour chaque enregistrement.




Descriptif technique de l'application


Tout d'abord avant de commencer l'application nous avons commencé par établir le plan et la maquette du site avec l'outil Balsamiq, ensuite nous avons utilisé GitHub pour que nous puissions travailler ensemble. Ensuite pour cette application on utilise l'IDE NetBeans utilisant le langage Java. Après, nous avons repris le MCD réalisé lors de l'AP3 avec WinDesign pour qu'il y ait seulement la table Utilisateur sur laquelle nous allons effectuer les opérations du CRUD.


Moyens techniques :


GitHub : Pour le versionnage et la gestion collaborative du code source. 

WampServer : Serveur local pour le développement et le test. 

WinDesign : Pour modéliser la base de données. 

Google Drive, Word : Pour la gestion de la documentation. 

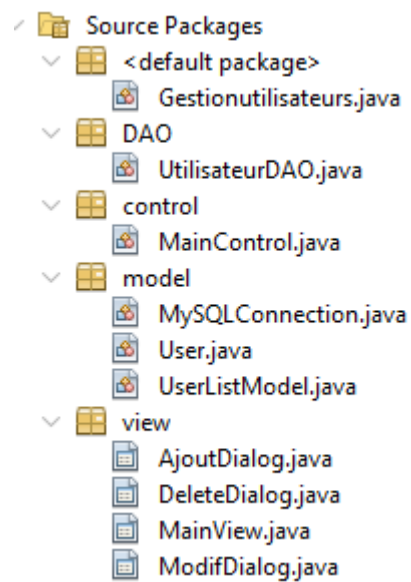
IDE Netbeans : pour le code (en Java). 

Serveur Web LAMP : sur Proxmox pour la gestion des données en préproduction 

Organisation du code

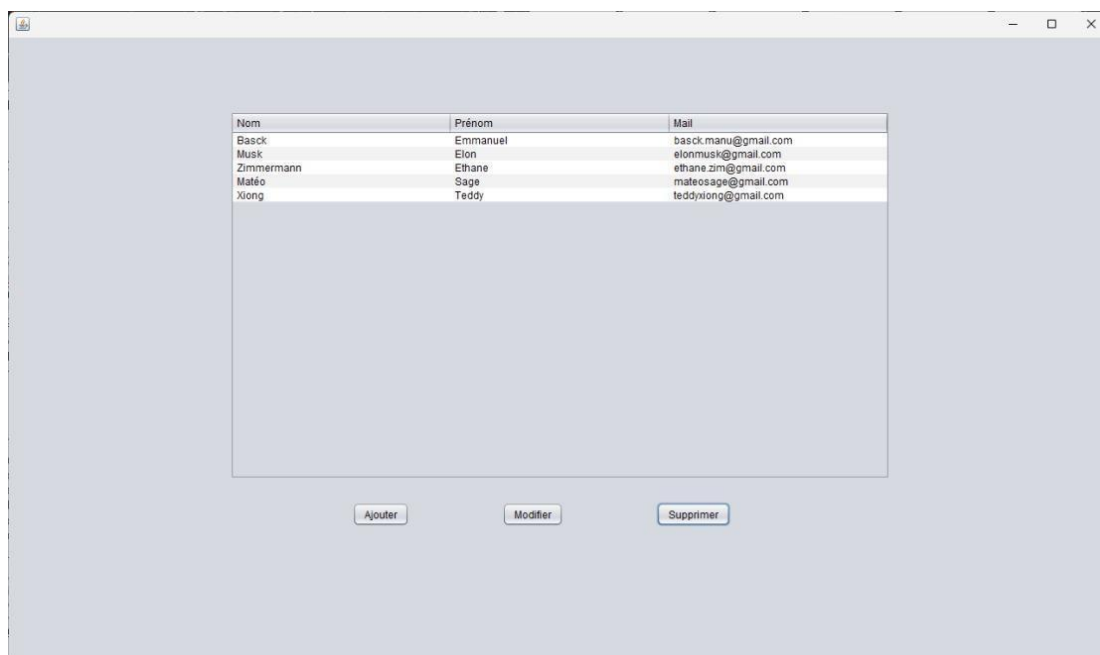
Nous avons également organisé en package de façon à avoir :

- ☐ Le package View qui contient les classes de l'interface graphique (Swing)
- ☐ Le package Model qui contient les classe « métier » (définies à partir du MCD)
- ☐ Le package DAO (Data Access Object) qui contient les classes d'accès aux données



Guide rapide d'utilisation

Pour commencer au lancement de l'applications nous pouvons consulter directement la liste des utilisateurs



Ensuite, nous avons la possibilité d'ajouter, de modifier ou de supprimer un utilisateur.

Lorsque nous cliquons sur ajouter, la page AjoutDialog s'ouvre, et nous pouvons remplir les champs de texte en fonction de la donnée demandée (Nom, Prénom, etc...).

A screenshot of a form titled 'AjoutDialog' for adding a new user. The form has five text input fields with labels: 'Nom', 'Prenom', 'Adresse Mail', 'Identifiant', and 'Mot de passe'. The 'Mot de passe' field is masked with asterisks. At the bottom of the form, there are two buttons: 'Ajouter' and 'Annuler'.

Nom :

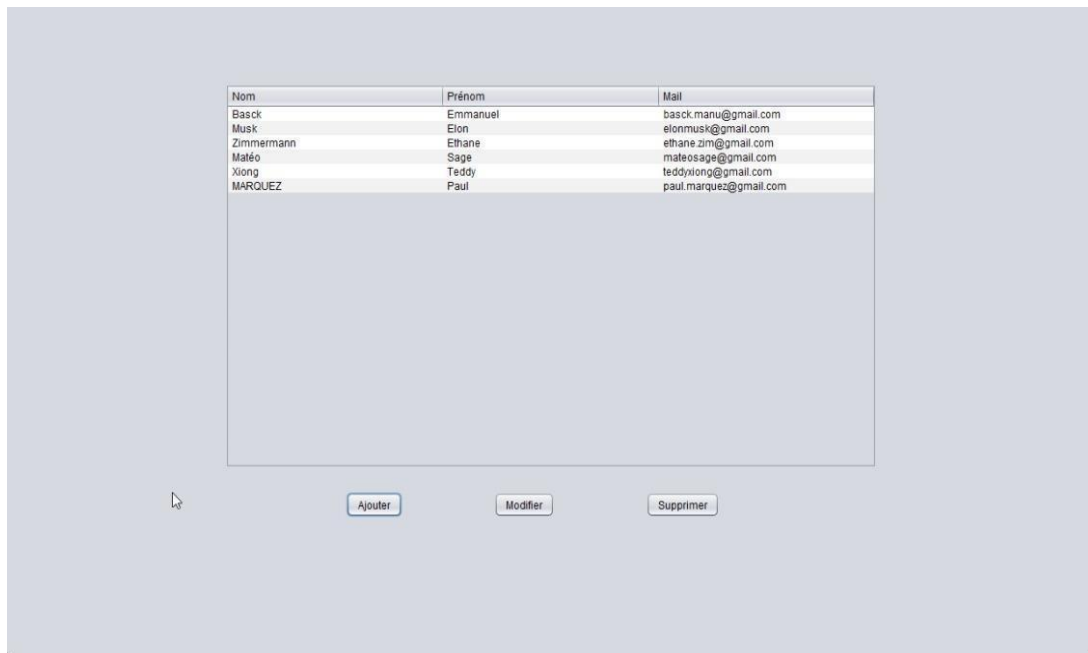
Prenom :

Adresse Mail :

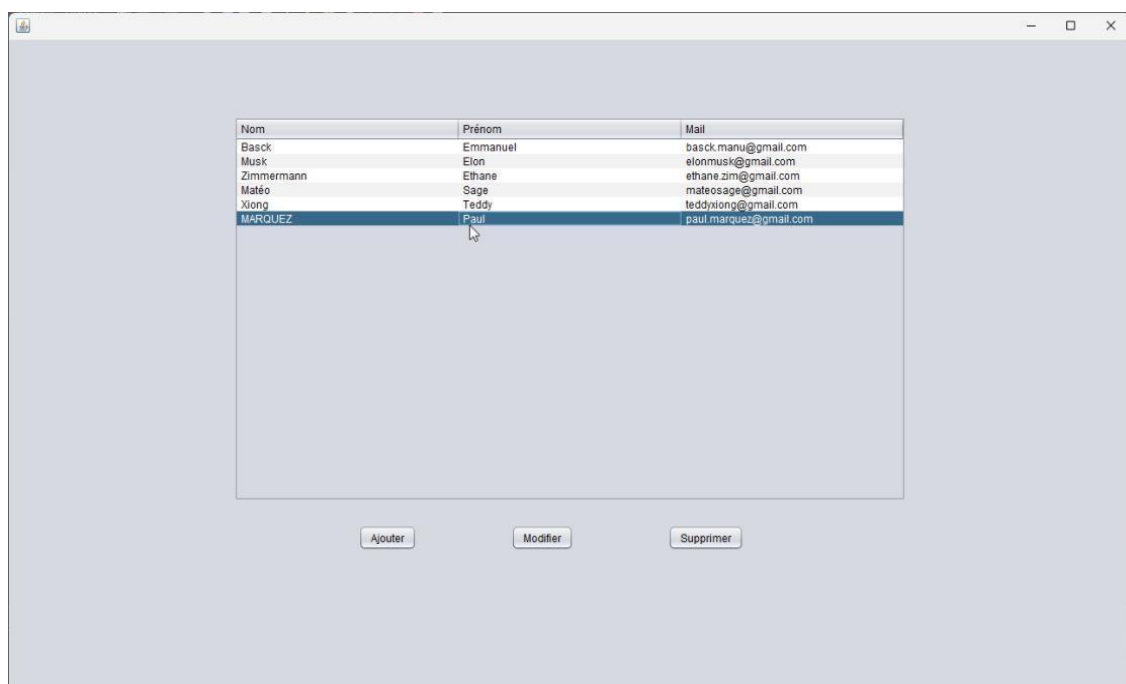
Identifiant :

Mot de passe :

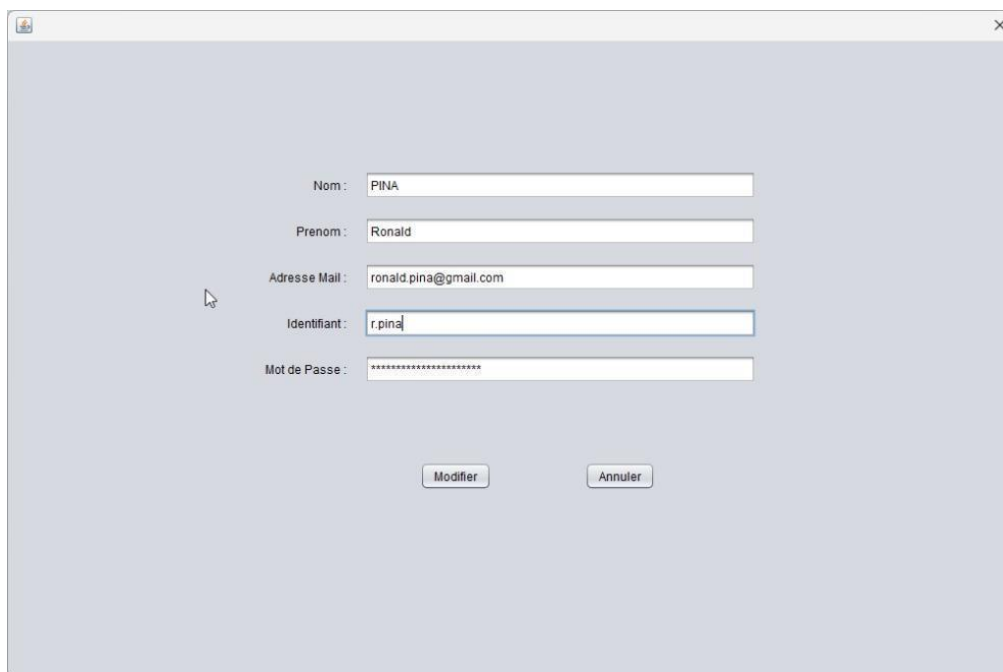
Une fois l'ajout effectué, un nouvel utilisateur s'ajoute à notre liste.



Afin de modifier un utilisateur, nous devons sélectionner l'utilisateur dans la liste.

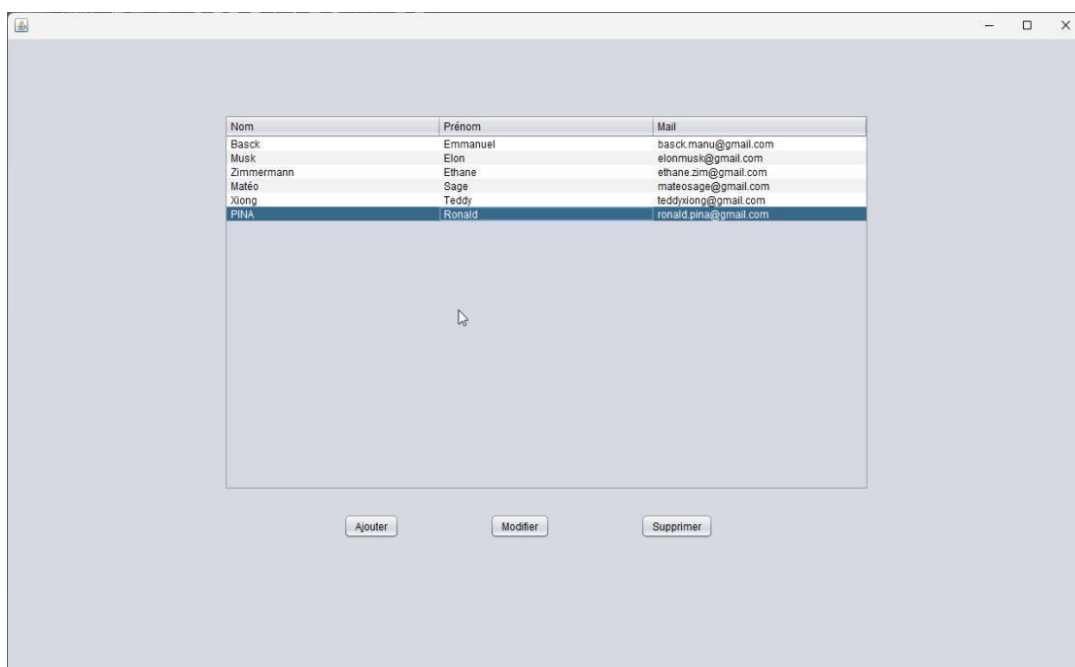


La même page que pour l'ajout s'affiche, mais cette fois-ci pour la modification. Les données sont donc déjà renseignées dans les champs de texte.



A screenshot of a web application window showing a form for modifying a user. The form contains five text input fields with the following labels and values: 'Nom : PINA', 'Prenom : Ronald', 'Adresse Mail : ronald.pina@gmail.com', 'Identifiant : r.pina', and 'Mot de Passe : *****'. Below the fields are two buttons: 'Modifier' and 'Annuler'.

Nous avons plus qu'à modifier notre utilisateur et cliquer sur « Modifier » on constatera qu'il a reçu une modification.

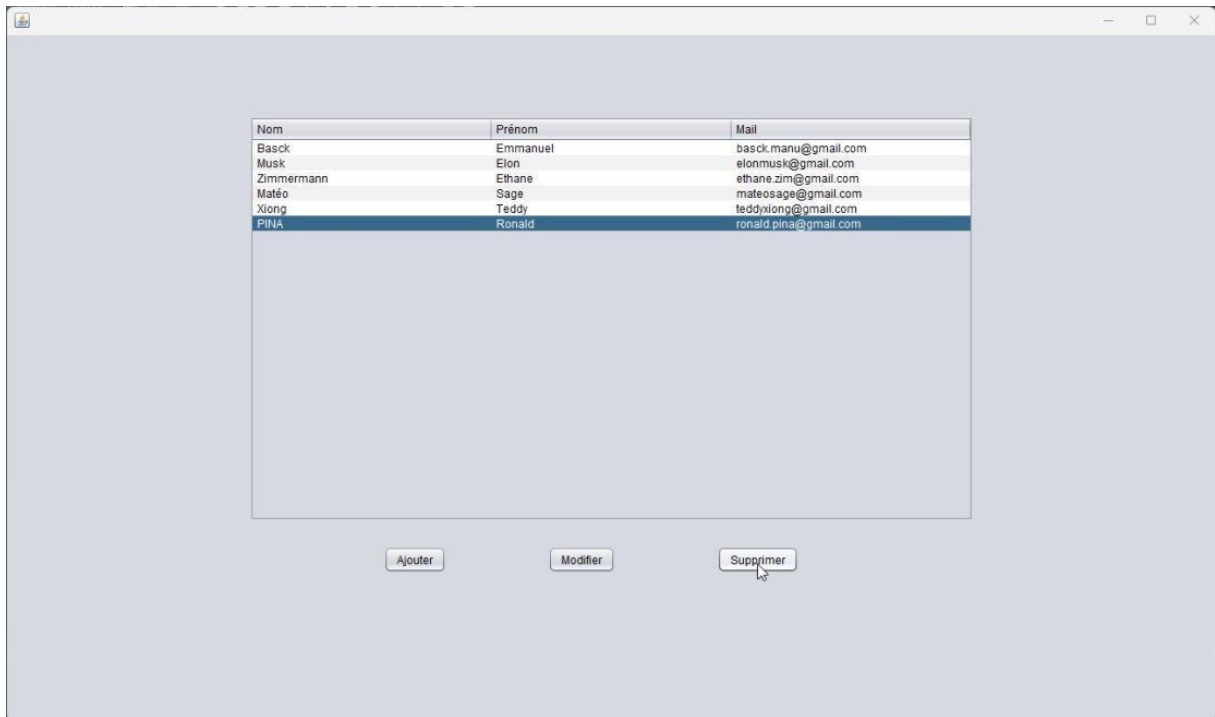


A screenshot of a web application window showing a table of users. The table has three columns: 'Nom', 'Prénom', and 'Mail'. The data is as follows:

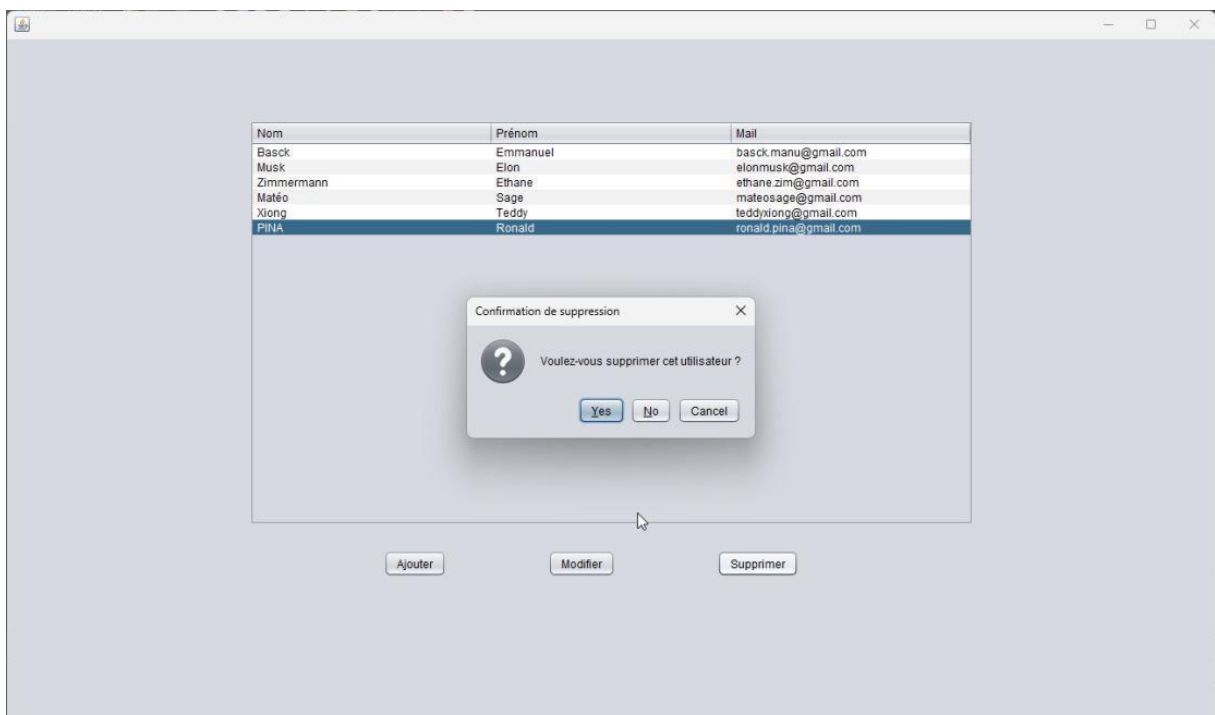
Nom	Prénom	Mail
Basck	Emmanuel	basck.manu@gmail.com
Musk	Elon	elonmusk@gmail.com
Zimmermann	Ethane	ethane.zim@gmail.com
Matéo	Sage	mateosage@gmail.com
Xiong	Teddy	teddyxiong@gmail.com
PINA	Ronald	ronald.pina@gmail.com

Below the table are three buttons: 'Ajouter', 'Modifier', and 'Supprimer'.

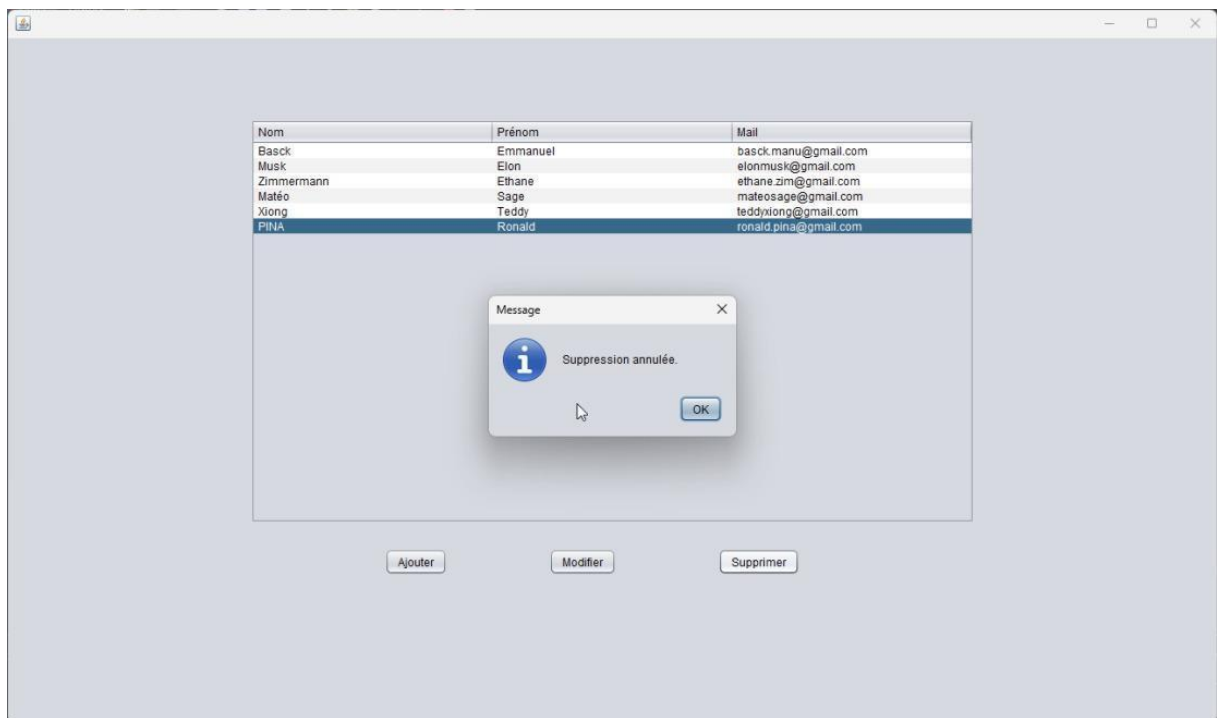
Pour finir, afin de supprimer un utilisateur nous devons, comme pour la modification, sélectionner un utilisateur dans la liste.



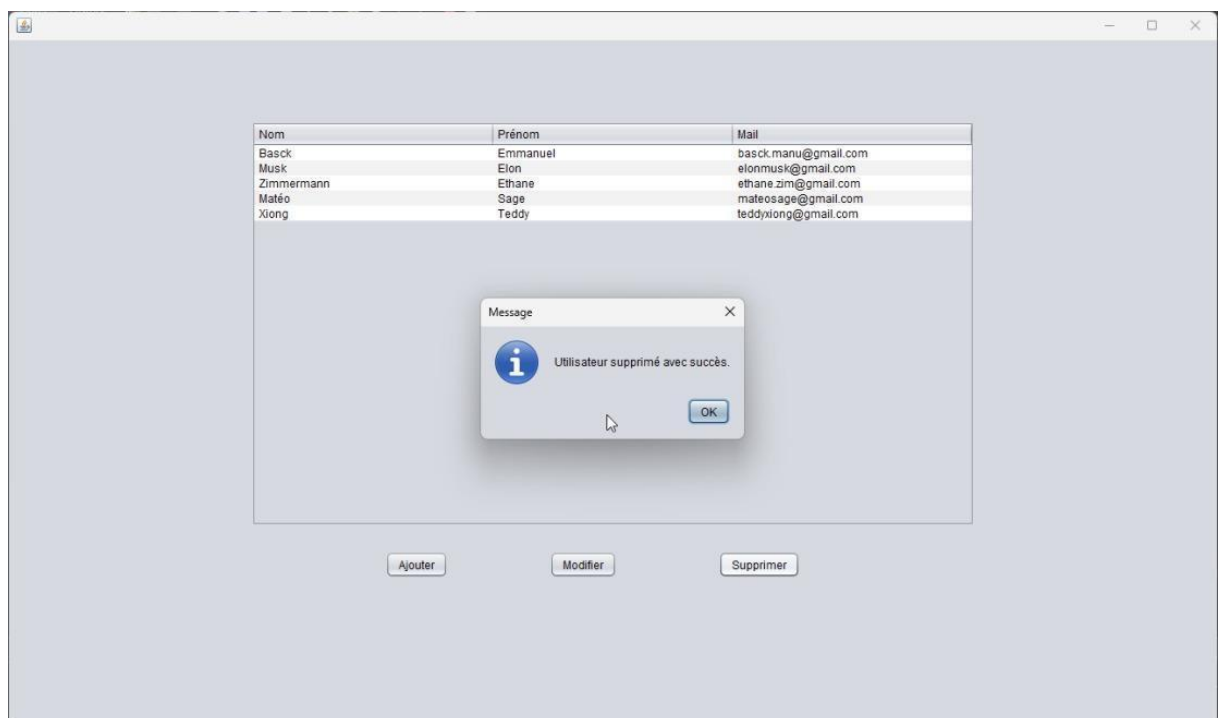
Ensuite en appuyant sur le bouton « Supprimer un utilisateur », un message s’affiche, demandant la confirmation de suppression.



Si on clique sur « Cancel » ou « No » la suppression sera annulée avec un message signalant l'annulation de la suppression.



Dans le cas ou on valide notre suppression en cliquant sur « Yes » alors une notification nous confirmera notre suppression.



Il est également possible de faire une suppression groupée en sélectionnant plusieurs utilisateurs.

Descriptif des tâches

Dans le cadre du projet pour la société Amset, nous avons été amenés à réaliser différentes tâches allant de l'analyse des besoins à la mise en œuvre technique. Voici un aperçu détaillé des actions menées tout au long de ce projet.

Date	Mathis	JB	Thomas	Abderrahim
21/03/2025	<ul style="list-style-type: none">• Formulaire Modifier• Méthode Update	<ul style="list-style-type: none">• Utilisateur DAO	<ul style="list-style-type: none">• Formulaire Ajouter• Méthode Create	<ul style="list-style-type: none">• Table Utilisateur• Liste des utilisateurs
28/03/2025	<ul style="list-style-type: none">• Méthode Update		<ul style="list-style-type: none">• Méthode Create• Support Oral	<ul style="list-style-type: none">• Méthode Delete
04/04/2025			<ul style="list-style-type: none">• Méthode Create	<ul style="list-style-type: none">• Méthode Delete
11/04/2025				<ul style="list-style-type: none">• Méthode Delete
18/04/2025	<ul style="list-style-type: none">• Documentation• Support Oral	<ul style="list-style-type: none">• Documentation• Support Oral	<ul style="list-style-type: none">• Diagram de Gantt• Documentation• Support Oral	<ul style="list-style-type: none">• Documentation• Support Oral
25/04/2025	<ul style="list-style-type: none">• Documentation• Support Oral	<ul style="list-style-type: none">• Documentation• Support Oral	<ul style="list-style-type: none">• Documentation• Support Oral	<ul style="list-style-type: none">• Documentation• Support Oral

Diagramme de Gantt

Le diagramme explique la durée de chaque tâche effectuée du début à la fin pendant les horaires de travail, ainsi que son avancement dans la finalité. Elles sont représentées par une couleur pour voir leurs évolutions pendant le projet.

Diagramme de Gantt				2025					
				Mars		Avril			
Semaines ->				21	28	4	11	18	25
Tache	Avancement	Début	Fin						
Utilisateur DAO	100%	21/03/2025	21/03/2025						
Table Utilisateur	100%	21/03/2025	21/03/2025						
Formulaire Ajout	100%	21/03/2025	21/03/2025						
Formulaire Modifier	100%	21/03/2025	21/03/2025						
Liste des Utilisateur	100%	21/03/2025	28/03/2025						
Méthode Create	100%	21/03/2025	04/05/2025						
Méthode Update	100%	21/03/2025	28/03/2025						
Méthode Delete	100%	28/03/2025	11/05/2025						
Diagramme de Gantt	100%	18/04/2025	18/04/2025						
Documentation	100%	18/04/2025	25/04/2025						
Support Oral	100%	28/03/2025	25/04/2025						

Difficultés rencontrées

Problèmes	Solutions
Gérer le repo GitHub depuis Netbeans	Demander aux professeurs la démarche pour réparer les erreurs git que Netbeans affiche et se coordonner lors des phases de commit, de pull et de push
Faire fonctionner le formulaire d'ajout d'un utilisateur	Les variables utilisées dans la fenêtre modale de l'ajout d'un utilisateur ne donnaient aucun résultat, à cause des variables que Netbeans a généré qui n'ont pas été utilisés dans les fonctions nécessaires pour retourner des données au Controller
Générer dans la vue principale la liste des utilisateurs	Il a fallu changer comment la liste des utilisateurs est générée en appelant la méthode getAll () de la classe DAO Utilisateur DAO au lieu d'avoir une liste brute
Supprimer un utilisateur	<p>Au lieu de supprimer un utilisateur avec le bouton supprimer en créant une fonction qui récupère l'id de l'utilisateur sélectionné dans le Controller (ce qui ne fonctionnait pas)</p> <p>On utilise à la place une fonction de la classe de la liste d'utilisateurs qui récupère l'id directement en utilisant un objet contenant la liste des utilisateurs.</p>

Une conclusion

Mathis : Ce projet m'a apporté une grande cohésion d'équipe, des connaissances en développement objet et ce fut l'un de mes projets préférés sur lequel j'ai pu travailler notamment grâce à l'équipe avec laquelle j'ai pu avancer très rapidement.

Thomas : Le projet Amset m'a apporté beaucoup de connaissance côté développement objet, notre équipe avait une bonne coordination et une bonne gestion du temps afin de réussir la totalité des objectifs attendus, malgré les problèmes rencontrés.

JB : Ce projet m'a permis de développer à la fois mes compétences techniques, notamment en Java et en conception d'interfaces avec Swing, et mes capacités à travailler en équipe. J'ai appris à mieux m'organiser, à communiquer efficacement, et à collaborer pour atteindre des objectifs communs. C'était une expérience très formatrice, que j'ai trouvée motivante et enrichissante, autant sur le plan humain que professionnel.

Abderrahim : Le projet m'a permis de renforcer d'autant plus mes connaissances liées au modèle MVC pour les clients lourds et de me familiariser avec l'ide Netbeans que jusqu'ici je n'avais que très peu utilisé.

Il m'a aussi permis de me confronter aux mêmes difficultés que l'on peut rencontrer lorsqu'une équipe doit utiliser un nouvel IDE pour développer son application, j'ai trouvé cette expérience utile dans la perspective de ne pas reproduire les mêmes erreurs en entreprise.