République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Ahmed Draia – Adrar Faculté des Sciences et de la Technologie
Département des Mathématiques et Informatique

Projet de fin d'études pour l'obtention du diplôme de

Master en Informatique

Option : Systèmes Intelligents

Thème

La préservation de la confidentialité pour la publication des données dynamiques

Réalisé par :

Labaire Abdelbasset

Chamanamas Ibrahim

Date de soutenance : 19 Octobre 2020

Membres du jury :	
Pr.OMARI MOHAMMED	Président
Dr.MAMOUNI EL MAMOUN	Examinateur
Dr. Salheddine KABOU	Encadreur

Année Universitaire 2019/2020

Remerciements

بسم الله الرحمان الرحيم
«وما توفيقي إلا بالله عليه توكلت و به استعين »

En termes de rédaction de notre mémoire de fin d'étude, nous remercions à tout instant notre dieu, qui nous a donné l'effort pour la continuité et l'arrivée à ce travail et qui nous à conduit à la réussite durant notre vie.

Nous voudrions exprimer notre profonde gratitude à notre encadreur Dr . Salheddine Kabou de nous avoir proposé un tel sujet intéressant, nous ouvrant ainsi les portes sur un domaine de recherche assez vivant, et pour son aide et ses conseils ainsi que ses remarques précieuses et ses injonctions justifiées qui nous ont été d'un apport inestimable, nous lui serions toujours reconnaissantes de tout cela.

Que les membres du jury, trouve ici nos vifs remerciements d'avoir accepté d'évaluer ce travail et pour le temps qu'il a consacré pour la lecture de ce mémoire.

Nous tenons à exprimer nos sincères remerciements à tout le personnel de Département des Mathématiques et Informatique

Toute personne qui a contribué de près ou de loin à l'achèvement de ce travail : nos amis, nos collègues pour leurs aides.

Enfin, nous espérons que ce travail aura la valeur souhaitée.

Merci a tout

Sommaire

Liste des Figures

Liste des Tableaux

Liste d'abréviation

PPDP: Privacy preserving data publishing

PCDP: La Préservation de la Confidentialité pour les Données Publiées

PPDDP: Privacy Preserving Dynamic Data Publication

XAMPP: X Apache MySQL Perl PHP

IDE: Integrated Development Environment

CRUD: Create, Read, Update, Delete

UML: Unified Modeling Language

GPL : General Public License

IBM : International Business Machines

FTP : File Transfer Protocol

SGBD : Système de Gestion de Base de Données

SGBDR Système de Gestion de Base de Données Relationnel

Résumé

Nombreuses organisations, exigent la collecte et le partage des données contenant des informations personnelles. La confidentialité de ces données doit être préservée avant l'externalisation au public commercial, c'est-à-dire aucune information sensible ne doit être divulguée. La préservation de la confidentialité pour les données publiées PCDP fait référence au processus de publication d'informations utiles tout en préservant la confidentialité des données. Dans les applications pratiques, la publication de données est plus compliquée lorsque les organisations publient plusieurs fois pour différents destinataires ou après des modifications (insertions, suppressions ou mise à jour). La préservation de la confidentialité pour la Publication de données dynamiques est un nouveau processus de préservation de confidentialité qui aborde l'anonymisation des données à des fins différentes. Dans notre mémoire, on va étudier la technique m-invariance qui assure l'anonymisation des données dynamiques.

Abstract

Many organizations require the collection and sharing of the data containing personal information. The privacy of this data must be preserved before outsourcing to the commercial public, i.e. any sensitive information should not be disclosed. Privacy preserving data publishing PPDP refers to the process of publishing useful information while preserving data privacy. In practical applications, data publishing is more complicated where the organizations publish a multiple times for different recipients or after modifications (insertions, deletions or update). Privacy preserving dynamic data publication PPDDP is a new process in privacy preservation which addresses the anonymization of the data for different purposes. In this paper, we will study the m-invariance technique that maintains the anonymization of dynamic data

Introduction générale

De nos jours, l'utilisation d'internet et des nouvelles technologies, pour satisfaire l'évolution des besoins de différents types d'utilisateurs (affaire, particulier), fait partie de la vie quotidienne. Toute information est disponible partout dans le monde à tout moment. Cela n'était pas possible il y a quelques années. Récemment, un nombre important de possibilités d'accès à l'information publique et privée sont apparues. Ainsi, nous avons un accès généralisé à un grand débit via internet grâce au déploiement de dispositifs fixes, mobiles ou encore sans fil qui permettent la connexion à l'internet sans presque se soucier de la limitation géographique.

De nombreuses organisations, en particulier les petites et les moyennes entreprises exigent la collecte et le partage des données contenant des informations personnelles. La confidentialité de ces données doit être préservée avant leurs publication, c'est à dire toute information sensible ne doit pas être divulgués. Selon une étude faite par 400 professionnelles dans le domaine de la technologie d'information, ils ont prouvé que la confidentialité est le facteur numéro *Un* pour la sécurité de Cloud **[14]**. Ce facteur assure que les données d'un client ne soient accessibles que par les entités autorisées. Les différentes solutions comportent plusieurs mécanismes de confidentialité telle que la gestion des identités et des accès, le cryptage, et l'anonymisation.

Anonymisation des données est l'une des techniques de la confidentialité qui se traduisent par la conservation de l'information, ce qui rend les données inutiles pour tout le monde sauf les propriétaires. Cette technique a été largement étudiée dans la littérature en proposant plusieurs modèles qui ont essayé de répondre aux problèmes causés au niveau de la confidentialité des données statiques (une seule publication des données).

Dans les applications pratiques, la publication de données est plus compliquée lorsque les organisations publient plusieurs fois pour destinataires différents ou après modifications (insertions, suppressions ou mises à jour) pour fournir des données à jour.

L'objectif de notre travail est d'étudier une technique qui satisfait les exigences de la publication des données dynamiques (plusieurs publications) tout en palliant les problèmes causées par les modèles de la publication des données statiques

Plan du mémoire

Notre mémoire est organisé selon le plan suivant :

- Dans le premier chapitre nous présentons la préservation de la confidentialité pour les données publiées Après, nous définissons des méthodes de préservation de la confidentialité dans les données publiées.
- Dans le deuxième chapitre nous présentons la méthode M-invariance pour publication de données dynamiques préservant la confidentialité
- Ensuite, nous avons représenté des différents diagrammes : diagramme de cas d'utilisation et description textuelle et diagramme de séquence.
- Le troisième chapitre est consacré à la réalisation de notre application en représentant les choix techniques, la réalisation du modèle de données, l'implémentation et la description des différents traitements (interfaces).
  Enfin, nous achevons le mémoire avec une conclusion générale en résumant notre travail.

Chapitre 1

La préservation de la confidentialité des données publiées

1. Introduction

La collecte de l'information numérique par les gouvernements, les entreprises et les particuliers a créé d'énormes possibilités pour la prise de décision basée sur la connaissance. Poussé par des avantages mutuels, ou par des règlements qui exigent que certaines données à publier, il y a une demande pour l'échange et la publication de données entre les différentes parties. Par exemple, les hôpitaux agréés en Californie sont tenus de présenter des données démographiques spécifiques sur chaque patient déchargé de ses installations [Carlisle et al, 2007]. Les données détaillées spécifiques à un individu contiennent souvent des informations sensibles, et la publication de ces données viole immédiatement sa confidentialité.

La tâche la plus importante est de développer des méthodes et des outils pour la publication de données d'une sorte que ces données doivent rester pratiquement utiles tout en préservant leurs confidentialités. Ce concept est appelé : PCPD ***La préservation de la confidentialité pour les données publiées*** (Privacy preserving data publishing). **[1]**

Dans ce chapitre, on vise à discuter et donner quelques définitions sur tous ce qui concerne l'approche de l'anonymisation ? Quelles sont ses types ? Les opérations sur les quelles, l'anonymisationse déroule parfaitement et la différence entre le terme d'anonymisation et le terme de cryptage.

2. La préservation de la confidentialité pour les données publiées

La publication de données préservant la confidentialité (PPDP) fournit des méthodes et des outils pour publier des informations utiles tout en préservant la confidentialité des données. Le PPDP a récemment fait l'objet d'une attention considérable dans les milieux de la recherche et de nombreuses approches ont été proposées pour différents scénarios de publication de données.

La confidentialité individuelle peut être maintenue à de nombreux égards, avec des propriétaires et des limitations différents.

2.1 L'anonymisation

L'anonymisation est un traitement qui consiste à utiliser un ensemble de techniques de manière à rendre impossible, en pratique, toute identification de la personne par quelque moyen que ce soit et ce de manière irréversible.

2.2 Anonymiser tout en préservant au maximum l'utilité du jeu de données

Le processus d'anonymisation vise à éliminer toute possibilité de ré-identification. **Il implique donc une nécessaire perte de qualité des données**. Leur exploitation future est ainsi limitée à certains types d'utilisation.

Pour construire un processus d'anonymisation pertinent, il est ainsi conseillé de :

- supprimer les éléments d'identification directe ainsi que les valeurs rares qui pourraient permettre une ré-identification aisée des personnes **par exemple** la connaissance précise de l'âge des individus présents dans un jeu de données peut permettre dans certains cas de ré-identifier très facilement les personnes centenaires
- distinguer les informations importantes des informations secondaires ou inutiles (c'est-à-dire supprimables)
- définir la finesse idéale et acceptable pour chaque information conservée
- définir les priorités (par exemple, est-il plus important de conserver une grande finesse sur telle information ou de conserver telle autre information).

L'enchaînement des techniques d'anonymisation à mettre en place qui peuvent être regroupées en deux familles : la randomisation et la généralisation.

- **La randomisation** consiste à modifier les attributs dans un jeu de données de telle sorte qu'elles soient moins précises, tout en conservant la répartition globale. Cette technique permet de protéger le jeu de données du risque d'inférence.

Exemple : permuter les données relatives à la date de naissance des individus de manière à altérer la véracité des informations contenues dans une base de données.

- **La généralisation** permet de généraliser les attributs du jeu de données en modifiant leur échelle ou leur ordre de grandeur afin de s'assurer qu'ils soient communs à un ensemble de personnes. Cette technique permet d'éviter l'individualisation d'un jeu de données. Elle limite également les possibles corrélations du jeu de données avec d'autres.

Exemple : dans un fichier contenant la date de naissance des personnes, il est possible de remplacer cette information par la seule année de naissance, ou une fourchette (par exemple : individus entre 20 et 30 ans).

Chaque technique d'anonymisation présente ses propres avantages et sera à décider en fonction du traitement de données et de l'objectif pour suivi.

3. Différences entre anonymisation et pseudonymisation

La pseudonymisation est un traitement de données personnelles réalisé de manière à ce qu'on ne puisse plus attribuer les données relatives à une personne physique sans avoir recours à des informations supplémentaires. En pratique la pseudonymisation consiste à remplacer les données directement identifiants (nom, prénom, etc.) d'un jeu de données par des données indirectement identifiants (alias, numéro dans un classement, etc.) **[15]**

4. La différence entre le cryptage et l'anonymisation

Bien que l'anonymisation et le cryptage sont des sujets connexes et ce sont des techniques utiles pour la sécurisation des données (confidentialité).

L'anonymisation des données est le processus de transformation des données afin qu'il puisse être traité d'une manière utile, tout en évitant que les données ne soient liées à des identités individuelles des personnes, des objets ou de l'organisation

Le cryptage consiste à transformer les données afin de les rendre illisible pour ceux qui n'ont pas la clé pour décrypter. Le cryptage peut être un outil utile pour faire l'anonymisation, et plus particulièrement lorsque la dissimulation des informations d'identification dans un ensemble de données **[2]**.

5. Méthodes de préservation de la confidentialité dans les données publiées

Quelques méthodes traditionnelles de préservation de la confidentialité dans les données publiées sont décrites brièvement ici. Les méthodes utilisées offrent traditionnellement une certaine protection de la vie privée, mais leurs inconvénients ont conduit à l'apparition de méthodes plus récentes.

5.1 De-identification

La de-identification est une technique traditionnelle d'exploration de données préservant la confidentialité, dans laquelle, afin de protéger la confidentialité des personnes, les données doivent d'abord être nettoyées avec une généralisation (remplacement des quasi identifiants par

des valeurs moins particulières mais cohérentes sur le plan sémantique).et suppression (ne publiant pas du tout certaines valeurs) avant la publication pour l'exploration de données **[3].**

Atténuer les menaces de ré-identification les concepts K-anonymat de l-diversité et de T-Closeness a été introduits pour améliorer l'exploration de données traditionnelle préservant la confidentialité **[4]**.

Il existe trois méthodes de de-identification préservant la confidentialité, à savoir K-anonymat, L-diversité et T-proximité. Certains termes courants sont utilisés dans le domaine de la confidentialité de ces méthodes:

- Les attributs d'identificateur incluent des informations qui distinguent de manière unique et directe des personnes, telles que le nom complet, le permis de conduire, le numéro de sécurité sociale.
- Attributs quasi-identifiant: un ensemble d'informations, par exemple, sexe, âge, date de naissance, code postal. Cela peut être combiné avec d'autres données externes afin de ré-identifier les individus.
- Les attributs sensibles sont des informations privées et personnelles. Exemples: maladie, salaire, etc.
- Les attributs insensibles sont les informations générales et inoffensives.
- Les classes d'équivalence sont des ensembles de tous les enregistrements composés des mêmes valeurs sur le disque. quasi-identifiants.

5.2 K-anonymat

Le concept de k-anonymat émerge d'une généralisation de la dépersonnalisation par quasi-identificateurs. Puisque la majorité des ensembles de données n'ont pas la combinaison grande-taille/peu de quasi-identificateurs requise, une méthode plus robuste fut développée. **[5]**

Dans le contexte des problèmes de k-anonymisation, une base de données est une table composée de **n** lignes et de **m** colonnes, où chaque ligne où chaque ligne de la table représente un enregistrement relatif à un individu particulier d'une population et les entrées dans les différentes lignes n'ont pas besoin d'être unique.

t#	sin	Gender	Zipcode	DOB	Disease
1	12313222	M	12499	07 june1985	HIV
2	12354562	M	12423	03-mai-87	FLU
3	12333396	M	13001	12 june 1998	FLU
4	12548787	M	13078	31-mars-93	Gastritis
5	15487252	F	13223	09-oct-90	Miocarditis
6	33266958	F	13009	13-mai-96	Miocarditis

Tableau 1 Table de données patient (données d'origine)

t#	Gender	Zipcode	DOB	Disease
1	M	124**	june1985	HIV
2	M	12***	Mar[1987-1999]	FLU
3	M	130**	june 1998	FLU
4	M	130**	[1965-2000]	Gastritis
5	F	132**	oct-90	Miocarditis
6	F	1300*	[1996-2000]	Miocarditis

Tableau 2 Table de donnée patient (données anonymisées)

5.3 L-diversité

Il s'agit d'une forme d'anonymisation basée sur le groupe qui est utilisée pour protéger la confidentialité des ensembles de données en réduisant la granularité de la représentation des données. Cette diminution est un compromis qui entraîne une perte de viabilité des algorithmes de gestion de données ou d'exploration pour gagner de la vie privée.

Le modèle **[6]** de L-diversité est une extension du modèle de k-anonymat qui diminue la granularité de la représentation des données à l'aide de méthodes incluant la généralisation et la suppression enregistrement donné mappe sur au moins k enregistrements différents dans les données.

Le modèle de l-diversité gère quelques-unes des faiblesses du modèle de k-anonymat dans lequel les identités protégées au niveau de k-individus ne sont pas équivalentes à la protection des valeurs sensibles correspondantes généralisées ou supprimées, en particulier lorsque les valeurs sensibles d'une homogénéité du groupe.

Le modèle de l-diversité inclut la promotion du travail intra-groupe

Diversité des valeurs sensibles dans le mécanisme d'anonymisation. Le problème avec cette méthode est que cela dépend de la gamme d'attributs sensibles. Si vouloir rendre les données L-divers si l'attribut sensible n'a pas autant que des valeurs différentes, des données fictives être inséré. Ces données fictives amélioreront la sécurité, mais pourraient entraîner des problèmes en cours d'analyse. La méthode de la diversité L est également sujette aux attaques d'asymétrie et de similarité et ne peut donc empêcher la divulgation d'attributs. **[7]**

t#	sin	Gender	Zipcode	DOB	Disease
1	12313222	M	12499	07 june1985	HIV
2	12354562	M	12423	03-mai-87	FLU
3	12333396	M	13001	12 june 1998	FLU
4	12548787	M	13078	31-mars-93	Gastritis
5	15487252	F	13223	09-oct-90	Miocarditis
6	33266958	F	13009	13-mai-96	Miocarditis

Tableau 3 Table de données patient (données d'origine)

t#	Gender	Zipcode	DOB	Disease
1	M	1****	[1960-1991]	HIV
2	M	1****	[1987-1999]	FLU
3	*	1****	[1998-2000]	FLU
4	M	13***	[1965-2000]	Gastritis
5	F	1****	[1990-2000]	Miocarditis
6	F	130**	[1996-2000]	Miocarditis

Tableau 4 Table de données patient (données diverses)

5.4 T-Closeness

Il s'agit d'une amélioration supplémentaire de l'anonymisation basée sur le groupe L-Diversité, utilisée pour préserver la confidentialité dans les ensembles de données en réduisant la granularité d'une représentation de données. Cette réduction est un compromis qui entraîne une perte d'adéquation des algorithmes de gestion des données ou d'extraction afin de gagner en confidentialité.

Le modèle **[8]** de proximité t (distance égale / hiérarchique) étend le modèle de l-diversité en traitant les valeurs d'un attribut de manière distincte en tenant compte de la distribution des valeurs de données pour cet attribut.

Une classe d'équivalence est dite t-close si la distance entre le transfert d'un attribut sensible de cette classe et la distribution de l'attribut dans la table entière est inférieure à un seuil t. On dit qu'une table a T-closeness si toutes les classes d'équivalence avoir T-closeness.

6. La préservation de la confidentialité des données dynamique

La publication de données dynamiques préservant la confidentialité (PPDDP) est un nouveau processus de préservation de la vie privée qui traite de l'anonymisation des données à des fins différentes.

La complexité de l'anonymisation des ensembles de données dynamiques est causée par les mises à jour des données (la suppression, l'insertion et la modification).

Les mises à jour des ensembles de données dynamiques se divisent en deux types: **mise à jour externe** et **mise à jour interne**.

6.1 La méthode de Byun et al.

Byun et al. **[9]** Est le premier à identifier d'éventuelles attaques contre la vie privée dues à une nouvelle publication et développe une solution pour prévenir efficacement ces attaques.

La définition de l'ensemble de données entièrement dynamique évolue depuis Byun et al. ont d'abord proposé l'idée de republication des ensembles de données dynamiques.

Intuitivement, les données d'un ensemble de données dynamiques ne restent pas les mêmes dans chaque version ultérieure.

Cependant, cette solution ne prend en charge que les insertions et n'est pas applicable en présence de suppressions. La confidentialité préservant la republication d'un ensemble de données entièrement dynamique reste un problème ouvert

6.2 La méthode de M-Distinct

Supposait en outre que si les micros données peuvent être entièrement dynamiques (c'est-à-dire, insertions / mises à jour / suppressions), il existe une certaine corrélation entre les anciennes valeurs et les nouvelles.

L'algorithme **[10]** de m-distinct présente le concept de l'ensemble de mise à jour candidat (ensemble de mise à jour candidat pour une valeur sensible, **s** - est l'ensemble des valeurs sensibles possibles auxquelles **s** peut être mis à jour, c'est-à-dire que **s** peut être mis à jour à n'importe quelle valeur dans sa mise à jour candidate avec une probabilité égale), en profitant des mises à jour des valeurs d'attributs sensibles qui ont les corrélations entre l'ancienne et la nouvelle pour résoudre le problème de la publication continue des données.

Name	Zip	H	disease
Ken	14k	20	dyspepsia
Julia	16k	23	Pneumonia
Tom	24k	32	Pneumonia
Harry	26k	35	Gastritis
Lily	29k	17	Glaucoma
Ben	31k	19	Flu

Tableau 5 Les micro données originales 1

Name	Zip	H	Disease
Ken	14k	20	Dyspepsia
Julia	18k	31	Lung Cqnce
Tom	15k	27	Pneumonia
Harry	23k	32	Dysepsia
Lily	12k	17	Glaucoma
Ben	26k	35	Pneumonia

Tableau 6 Les micro données originales 2

6.2.1 Les concepts de la méthode m-distinct

Nous formulons le concept suivant pour décrire les candidats à la mise à jour d'une valeur **[11]** :

- **Candidate Update Set (CUS) :**

Supposons que a soit un élément dans le domaine de l'attribut A (a ∈ dom (A)), son candidat update set CUS (a) est l'union de certains éléments dans dom (A), de sorte qu'un a une probabilité de mise à jour non nulle.

Notez que si b ∈ CUS (a), alors CUS (b) ⊆ CUS (a) doit tenir. De même, nous avons la notion suivante pour un groupe de valeurs sensibles

- **Update Set Signature (USS) :**

Supposons que le groupe QI g contienne n enregistrements et que leurs valeurs sensibles soient s1, s2, ..., sn, respectivement. La signature du jeu de mise à jour d'USS (g) est alors un multi-ensemble:

{CUS (s1), CUS (s2) ... CUS(sn)}.

Étant donné que l'USS est un ensemble multiple de CUS, le même CUS peut apparaître plusieurs fois dans un USS, car plusieurs enregistrements peuvent avoir la même valeur sensible et différentes valeurs sensibles peuvent même avoir le même ensemble de mise à jour candidat.

- **Légal Update Instance:**

Un ensemble de valeurs sensibles S = {s1, s2, ..., sn} est une instance de mise à jour légale d'un USS si les conditions suivantes sont réunies:

- Le nombre de valeurs sensibles dans S est égal à le nombre de CUS aux USS: | S | = | USS |.
- Pour toute valeur si dans S, il existe au moins un ensemble de mise à jour candidat CUSj tel que si ∈ CUSj.
- Pour tout ensemble de mise à jour candidat CUSj dans USS, il y a au moins une valeur sensible si dans S telle que si ∈ CUSj.

Exemple

Dans l'ensemble sensible C1 (tableau 5, 6) de Julia est {Dyspepsie, Pneumonie}.

À l'aide de connaissances de base implicites, nous savons que

CUS (dyspepsie) = {dyspepsie, gastrite, autres maladies du système digestif} et

CUS (pneumonie) = {pneumonie, grippe, cancer du poumon, autres maladies du système respiratoire}.

Ainsi, la signature du jeu de mise à jour de Julia dans la première version est {**CUS** (dyspepsie), **CUS** (Pneumonie)}.

Si nous choisissons au hasard un élément de **CUS (dyspepsie)** et de **CUS (pneumonie)** respectivement, alors les deux éléments doivent être une instance légale d'**USS (Julia1).**

7. Conclusion

Dans une société de plus en plus axée sur les données, les informations personnelles sont souvent collectées et distribué avec facilité.

Dans ce chapitre, nous avons présenté la définition et la protection de la confidentialité des personnes dans la publication des données. En particulier, nous nous sommes concentrés sur les Méthodes de préservation de la confidentialité dans les données statique (k-anonyme, L-Diversité et T-Closeness), et les méthodes des données dynamique (Byun et al. (2006), m-distinct).

Chapitre 2

Analyse ET Conception

1. Introduction

Dans ce chapitre, nous allons présenter la méthode de préservation de la confidentialité de donnée dynamique (m-invariance) et l'étape d'identification des besoins et spécification des fonctionnalités, de processus de développement que nous avons utilisé dans notre projet.

Cette étape consiste à identifier et modéliser les besoins des utilisateurs à partir des diagrammes des cas d'utilisations. Les interactions entre les acteurs et le système (au sein des cas d'utilisation) seront explicités sous forme textuelle (description textuelle) et sous forme graphique au moyen des diagrammes de séquence qui représentent exactement le déroulement de la tâche courante. StarUML est l'outil utilisé pour crée les différents diagrammes.

2. M-invariance

Byun et al. **[12]** Ont abordé le problème de la publication continue de données dans le scénario d'insertion uniquement. Xiao et al. Ont identifié que la publication continue de données est plus complexe que cela. Ils ont montré que même si les rejets continus suivent le principe proposé par Byun et al. Ils sont vulnérables à d'autres inférences plus sophistiquées. Plus précisément, ils ont étendu le scénario de publication continue où les micros données sont modifiées avec à la fois des insertions de nouveaux enregistrements et des suppressions de certains précédents.

La première étude sur la confidentialité préservant la publication d'ensembles de données entièrement dynamiques, qui peuvent être modifiés par n'importe quelle séquence d'insertions et de suppressions. Le cœur de notre solution est l'intégration de deux nouveaux concepts:

M-L'invariance et ***la généralisation contrefaite***. Le premier est un nouveau principe de généralisation, dont la satisfaction assure une forte protection des informations sensibles en republication. Cette dernière est une technique qui facilite l'application de m-l'invariance, en présence d'absence critique.

2.1 Généralisation de M-Invariance

T(n-1) la dernier table des micros données

T*(n-1) la dernier table publié

T(n) la table T(n-1) après la mise a jour

T ∗ (n) la table de généralisation de nouveau version publié

R(n) Statistiques de contrefaçon publiées

Cette section élabore le calcul des **{T ∗ (n), R (n)}** publiés lors de la n-ième publication. Nous nous concentrons sur **T ∗ (n)** car une fois qu'il est prêt, produire **R (n)** est trivial.

2.1.1 L'Algorithme

L'algorithme **[13]** Nous visons à atteindre deux objectifs intuitifs. ***Premièrement***, le nombre de tuples contrefaits doit être minimisé, car ils ne correspondent à aucun enregistrement des micros données. ***Deuxièmement***, nous utilisons la moindre généralisation pour déformer les valeurs de QI.

Selon le lemme : le calcul de T ∗ (n) ne nécessite que les tableaux des micros données T (n - 1), T (n) et la dernière relation publiée
T ∗ (n - 1).
Divisons les tuples de T (n) en deux ensembles disjoints
S∩ = T (n) ∩T (n − 1) et S− = T (n) −T (n − 1).

Notre algorithme garantit deux propriétés:

- Pour tout tuple **t ∈ S∩**, ses groupes d'hébergement généralisés **t.QI ∗ (n - 1)** et **t.QI ∗ (n)** ont la même signature.

- Pour tout tuple **t ∈ S−**, son tuple généralisé **t ∗** dans **T ∗ (n)** est dans un groupe QI qui a au moins m tuples, et tous les tuples ont des valeurs sensibles distinctes.

2.1.2 La phase Division

Pour chaque **t ∈ S∩,** nous définissons sa signature comme la signature de son groupe d'hébergement généralisé en **T ∗ (n - 1).**

Cette phase partitionne simplement S∩ en plusieurs compartiments, de sorte que chaque compartiment ne contient que les tuples avec la même signature.

Exemple

T (1) : Tableau 6 (b), T (2) : Tableau 5 (a)
S∩ = T (2) ∩T (1)
S∩ = {Bob, David, Jane, Linda, Gary et Steve.}

Name	Age	Zip	disease
Bob	21	12000	dyspepsia
Alice	22	14000	bronchitis
Andy	24	18000	flu
David	23	25000	Gastritis
Gary	41	20000	flu
Helen	36	27000	Gastritis
Jane	37	33000	dyspepsia
Ken	40	35000	flu
Linda	43	26000	Gastritis
Paul	52	33000	dyspepsia
Steve	56	34000	Gastritis

Tableau 7 (a) Micro donnée

Name	Age	Zip	disease
Bob	21	12000	dyspepsia
David	23	25000	gastritis
Emily	25	21000	Flu
Jane	37	33000	Dyspepsia
Linda	43	26000	Gastritis
Gary	41	20000	Flu
Mary	46	30000	Gastritis
Ray	54	31000	Dyspepsia
Steve	56	34000	Gastritis
Tom	60	44000	Gastritis
Vince	65	36000	Flu

Tableau 8 (b) Micro donnée

Dans l'exemple en cours, S∩ contient les tuples de Bob, David, Jane, Linda, Gary et Steve.

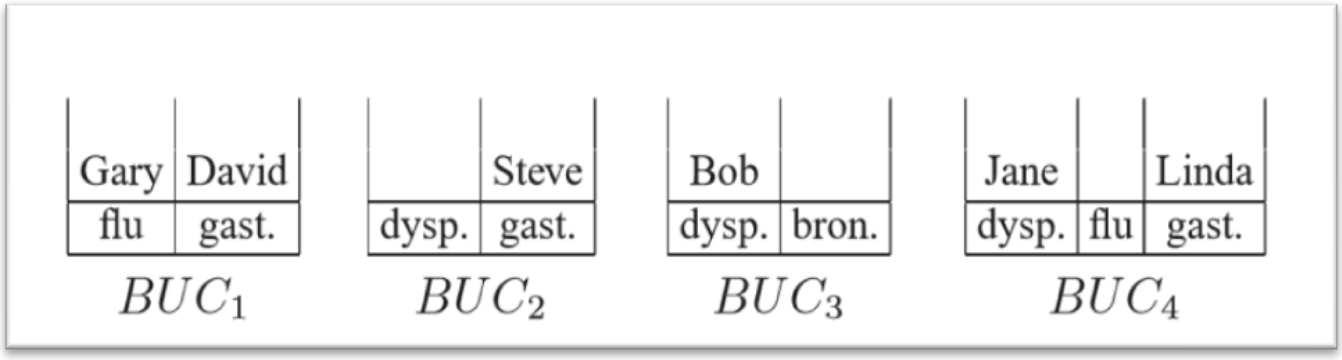

Figure 1 Contenu du seau après la phase de division

Le tuple de Bob, par exemple, a une signature {dyspepsie, bronchite} (c'est-à-dire les valeurs sensibles du groupe 1 du tableau 1b). C'est le seul élément du seau BUC3. Un seau peut avoir plusieurs tuples. Par exemple, BUC1 contient Gary et David, car ils partagent une signature équivalente {flu, gastrite}.

2.1.3 La phase Équilibrage

Contrairement à la phase précédente, nous travaillerons avec les valeurs sensibles des tuples, par opposition à leurs signatures.

Nous disons qu'un seau BUC est équilibré, si chaque valeur sensible de sa signature appartient au même nombre de tuples dans BUC.

Par exemple, sur la figure 1, BUC1 est équilibré, car sa signature a deux valeurs **flu** et **gastrite**, chacune étant possédée par un tuple. L'objectif de cette phase est d'équilibrer tous les seaux.

S− est égal à **{Emily, Mary, Ray, Tom, Vince}**

Chaque seau BUC est inspecté tour à tour. Si BUC n'est pas équilibré, il y a une «pénurie» de certaines valeurs sensibles dans BUC. Dans ce cas, nous essayons de combler la pénurie en déplaçant les tuples de S− dans BUC, tant que le S− résultant est toujours éligible m (la raison sera claire dans un certain temps).

Dans la figure 2, BUC2 est déséquilibré, car il y a une (tuple avec) gastrite mais pas de dyspepsie. S− est égal à {Emily, Mary, Ray, Tom, Vince}. Nous pouvons déplacer Ray (dont la

valeur de la maladie est la dyspepsie) vers BUC2, car il reste 2 flu et 2 gastrites dans S−, qui est toujours éligible.

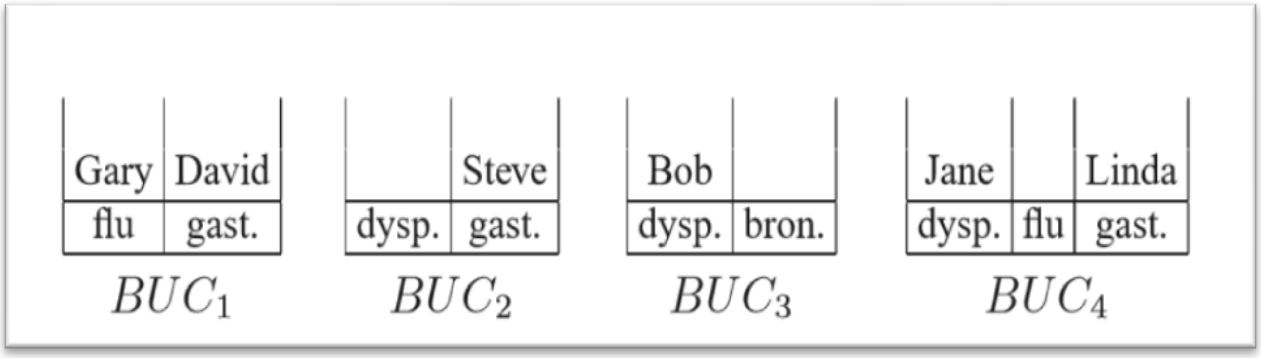

Figure 2 Contenu du seau avent la phase d'équilibrage

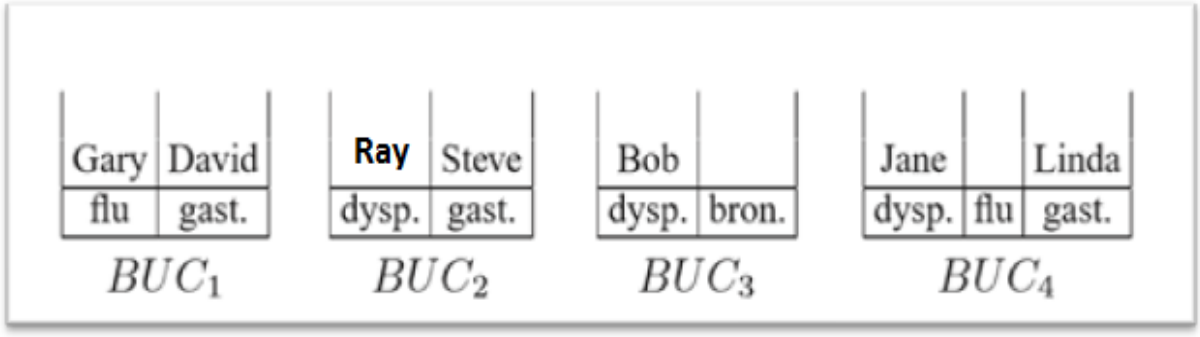

Figure 3 Contenu du seau avant la phase d'équilibrage

Le BUC2 mis à jour, illustré à la figure 3, devient équilibré.

Si S− ne peut pas être utilisé pour fixer un BUC de seau non équilibré, il y a deux possibilités:

- aucun tuple dans S− porte la ou les valeurs sensibles requises
- S− n'est plus éligible m après un retrait de tuple.

Dans les deux cas, nous insérons des contrefaçons pour équilibrer BUC. Pour suivant notre exemple de la figure 3, BUC3 et BUC4 sont déséquilibrés, mais aucun d'eux ne peut être corrigé avec S−.

Plus précisément, BUC3 a besoin d'une bronchite, qui est absente dans S−.

BUC4 a besoin d'une flu bien qu'il y ait des tuples avec flu dans S−, le retrait de l'un d'eux laisse 2 gastrites et 1 flu dans S−, violant la contrainte d'éligibilité 2.

Par conséquent, comme sur la figure 3, deux contrefaçons c1 et c2 (avec des valeurs sensibles bronchite et flu) sont ajoutées respectivement à BUC3 et BUC4, les deux étant maintenant équilibrées.

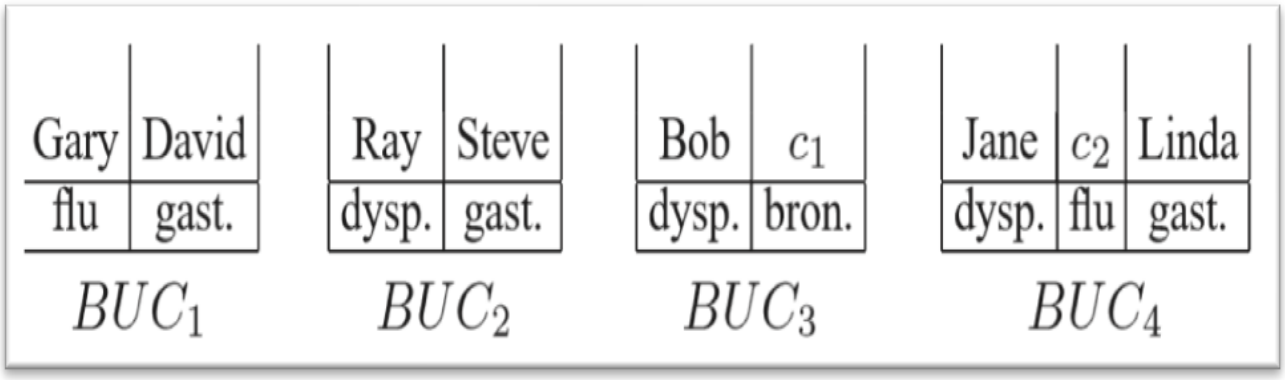

Figure 4 Contenu du seau après la phase d'équilibrage

2.1.4 La phase d'affectation

Dans cette phase, nous attribuons les tuples restants dans S− à des compartiments, soumis à deux règles.

Tout d'abord, chaque tuple **t ∈ S−** ne peut être placé que dans un compartiment dont la signature inclut t [As].

Deuxièmement, à la fin de la phase, tous les seaux sont toujours équilibrés. Si nécessaire, de nouveaux compartiments (la signature de chaque compartiment contient au moins m valeurs) peuvent être générés et ils obéissent également à ces règles. Comme prouvé plus loin, un tel schéma d'attribution existe toujours, tant que S− est éligible m.

Dans l'exemple en cours, **S− = {Emily, Mary, Tom, Vince}** après la phase d'équilibrage. La figure 5 illustre les compartiments après toutes les affectations. Les 4 tuples de S− sont tous placés dans BUC1, qui reste équilibré.

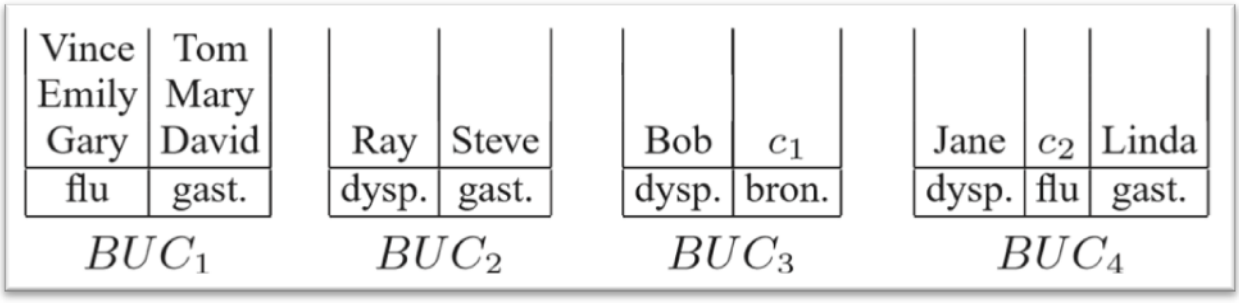

Figure 5 Contenu du seau après la phase d'affectation

2.1.5 La phase Répartir

Cette dernière phase traite chaque seau BUC individuellement. Il divise BUC en groupes | **BUC | / s QI**, où s (≥ m) est le nombre de valeurs dans la signature de BUC.

Chaque groupe a **s** tuples, prenant respectivement la valeur sensible **s** dans la signature. Le fractionnement optimise la qualité de la généralisation.

Étant donné BUC1 dans la figure 2c, notre algorithme de fractionnement crée trois groupes

QI: {David, Emily}, {Gary, Mary}, {Vince, Tom}.

Ils mènent aux groupes AQ 2, 4 et 6 du tableau 3a.

De même, BUC2, BUC3 et BUC4 donnent respectivement les groupes **QI 5, 1, 3.**

Quelques derniers mots concernent l'âge [21, 22] du groupe AQ 1 dans le tableau 3a. Ce groupe couvre le tuple de Bob (21 ans) et une contrefaçon (âge).

Nous aurions publié 21, si la généralisation de l'intervalle minimum avait été suivie. En pratique, cependant, les données personnelles ne doivent pas être divulguées directement.

Par conséquent, nous exigeons que chaque valeur de QI dans

T ∗ (n) soit un intervalle dont la longueur est au moins un seuil (par exemple, 2 pour l'âge). Ce seuil peut varier pour différents attributs QI (par exemple, 2k pour Zip code).

G.ID	Age	Zip	disease
1	[21,22]	[12k,14k]	Dyspepsia
1	[21,22]	[12k,14k]	bronchitis
2	[21,22]	[18k,25k]	flu
2	[21,22]	[18k,25k]	Gastritis
3	[21,22]	[20k,27k]	flu
3	[21,22]	[20k,27k]	Gastritis
4	[21,22]	[26k,35k]	dyspepsia
4	[21,22]	[26k,35k]	flu
4	[21,22]	[26k,26k]	Gastritis
5	[21,22]	[33k,34k]	dyspepsia
5	[21,22]	[33k,34k]	Gastritis

Tableau 9 généralisation de T(a)

G.ID	Age	Zip	Disease
1	[21,23]	[12k,25k]	Dyspepsia
1	[21,23]	[12k,25k]	Gastritis
2	[25,43]	[21k,33k]	Flu
2	[25,43]	[21k,33k]	Dyspepsia
2	[25,43]	[21k,33k]	Gastritis
3	[41,46]	[20k,30k]	Flu
3	[41,46]	[20k,30k]	Gastritis
4	[54,56]	[31k,34k]	Dyspepsia
4	[54,56]	[31k,34k]	Gastritis
5	[60,65]	[36k,44k]	Gastritis
5	[60,65]	[36k,44k]	flu

Tableau 10 généralisation de T(b)

Name	G.ID	Age	Zip	disease
Bob	1	[21,22]	[12k,14k]	dyspepsia
C1	1	[21,22]	[12k,14k]	bronchitis
David	2	[23,25]	[21k,25k]	Gastritis
Emily	2	[23,25]	[21k,25k]	Flu
Jane	3	[37,43]	[26k,33k]	dyspepsia
C2	3	[37,43]	[26k,33k]	Flu
Linda	3	[37,43]	[26k,33k]	Gastritis
Gary	4	[41,46]	[20k,30k]	Flu
Mary	4	[41,46]	[20k,30k]	Gastritis
Ray	5	[54,56]	[31k,34k]	Dyspepsia
Steve	5	[54,56]	[31k,34k]	Gastritis
Tom	6	[60,65]	[36k,44k]	Gastritis
Vince	6	[60,65]	[36k,44k]	Flu

Tableau 11 T*(b) avec contrefaçon

Group-ID	Count
1	1
3	1

Tableau 12 Statistiques de contrefaçons publiées

3. Des diagrammes des cas d'utilisations

Les diagrammes de cas d'utilisation sont des diagrammes UML utilisés pour donner une vision globale du comportement fonctionnel d'un système logiciel.

3.1 Propriétaire

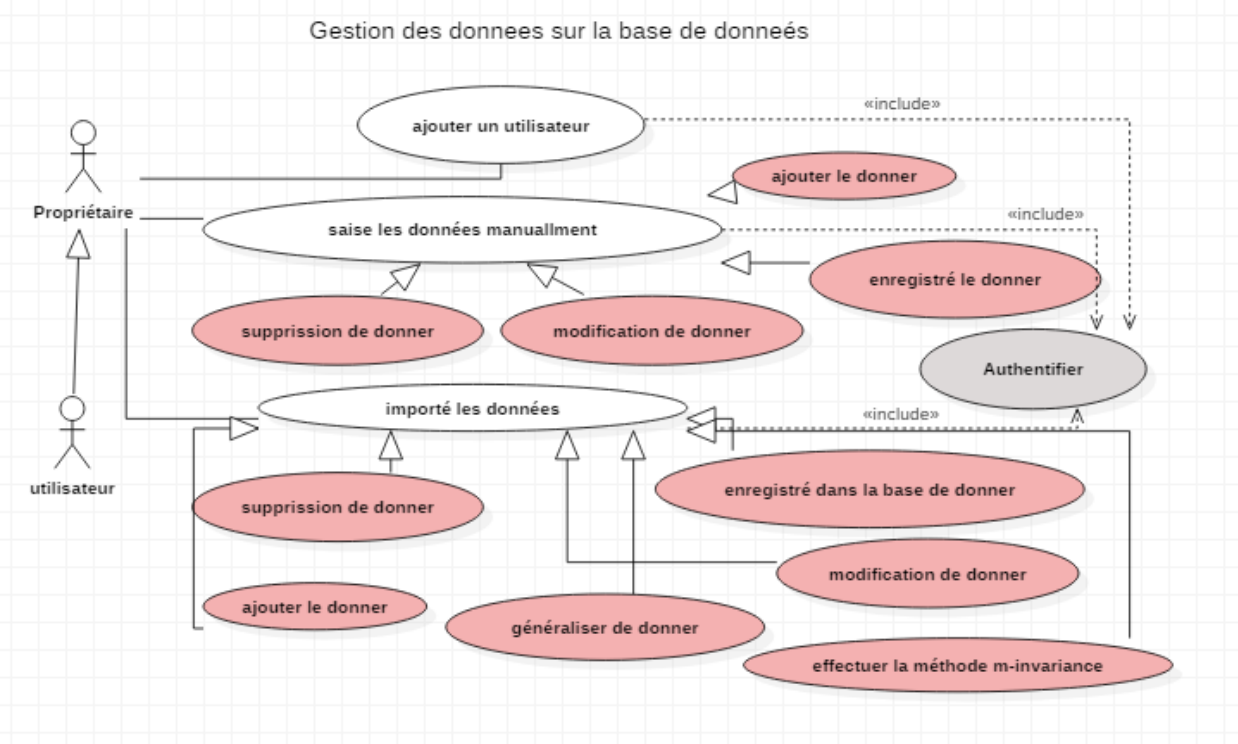

Figure 6 Digramme de cas d'utilisation 'Propriétaire'

3.2 Utilisateur

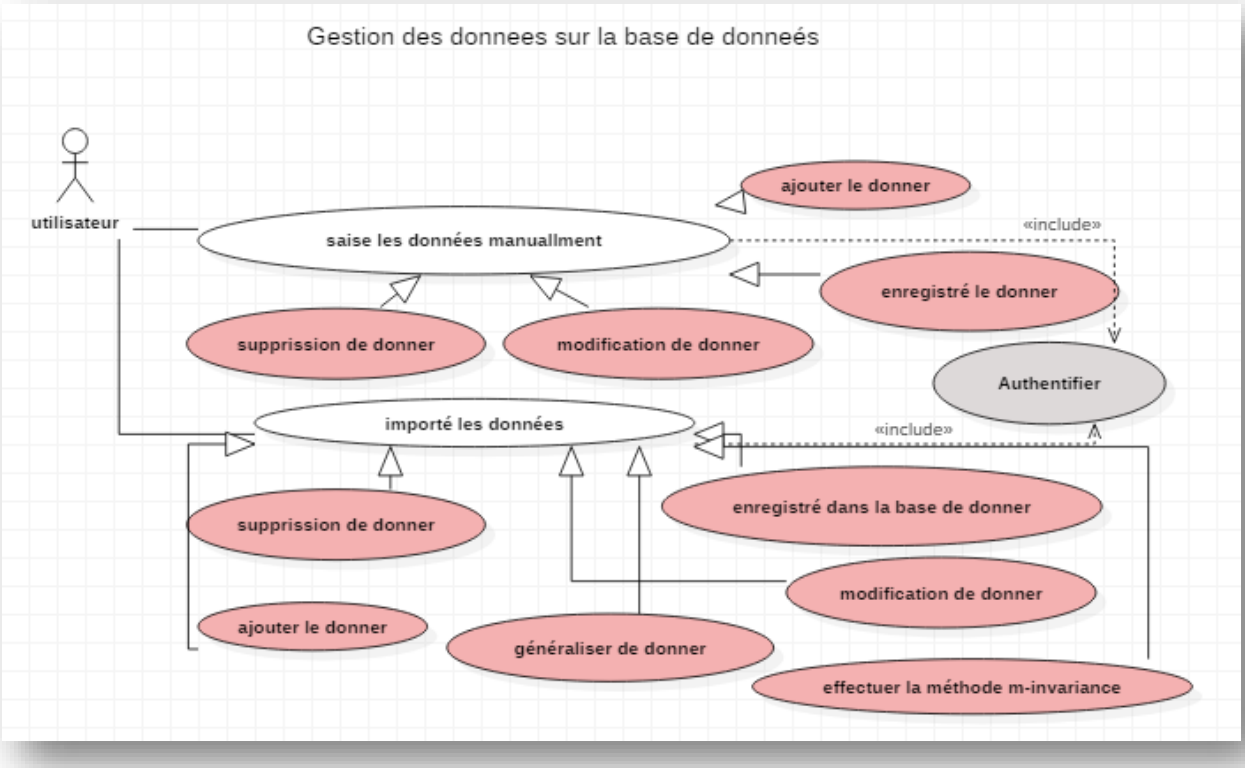

Figure 7 Digramme de cas d'utilisation 'utilisateur'

4. Description textuelle des cas d'utilisation et diagrammes des séquences :

4.1 Description textuelle de cas d'utilisation de S'authentifier à logiciel :

Cas d'utilisation	S'authentifier à logiciel
Acteur Principaux	Propriétaire, utilisateur
Acteurs Secondaires	Aucun
Objectif	Permettre à l'utilisateur d'accéder à sa session en fournissant son identifiant et son mot de passe.
Pré condition	L'accès aux applications est déjà fait.
Post-condition	Ouverture de la session.
Scenario Nominal	1. L'utilisateur veut connecter à son compte. 2. Le système affiche un formulaire d'authentification. 3. L'utilisateur saisit ses informations (nom d'utilisateur et mot de passe). 4. L'utilisateur envoie le formulaire. 5. Le système vérifier le contenu du formulaire. 6. Le système ouvre la session.
Scenarios Alternatifs	5.a) Si l'utilisateur saisit une information invalide le

	système affiche un message d'erreur. Aller à 2. 5.b) Formulaire vide. Aller à 2.

4.2 Diagramme de séquence de cas d'utilisation de S'authentifier à logiciel :

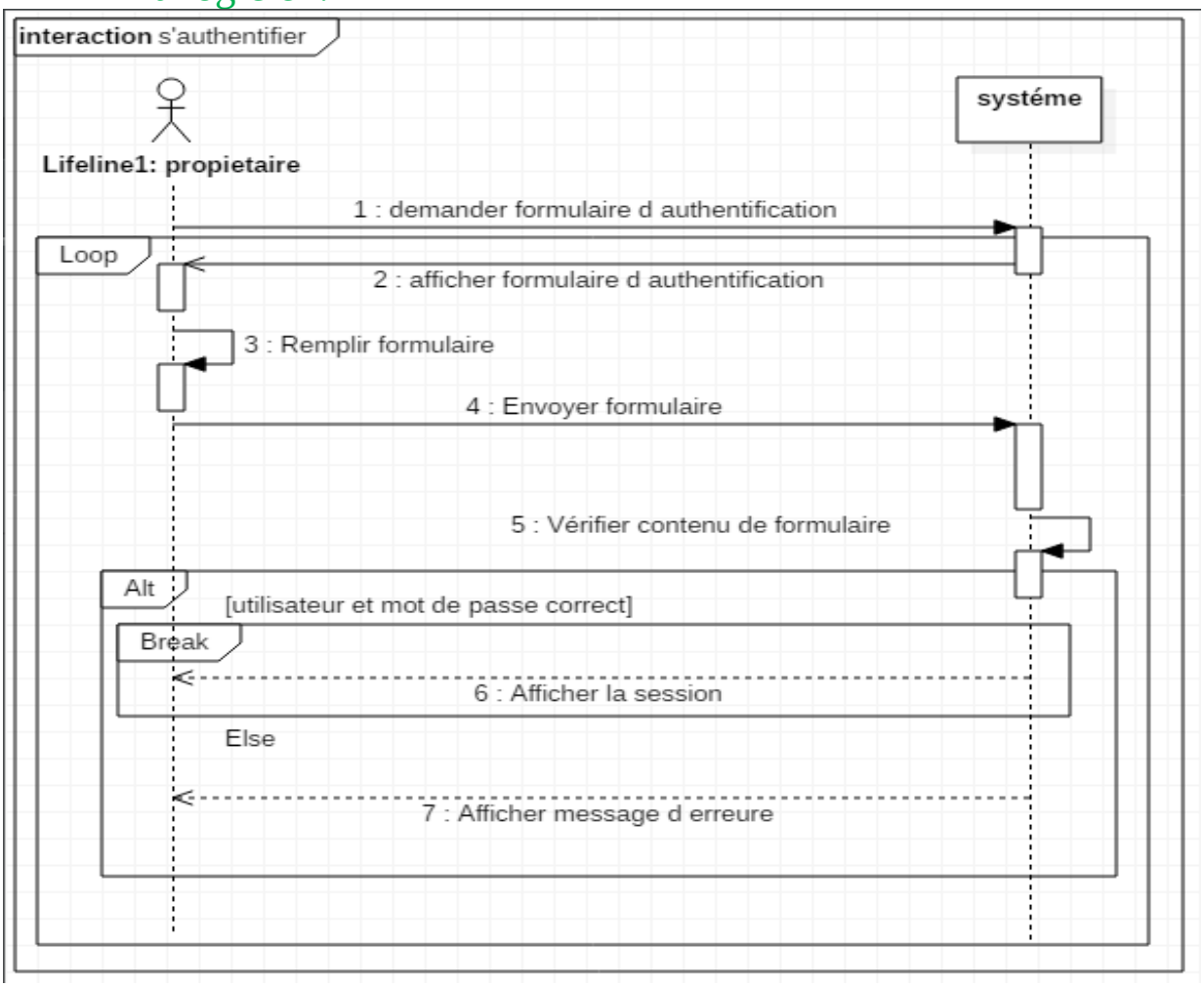

Figure 8 Diagramme de séquence s'authentifier

4.3 Description textuelle de cas d'utilisation Gréer un compte d'utilisateur

Cas d'utilisation	Gréer un compte d'utilisateur
Acteur Principaux	Propriétaire
Acteurs Secondaires	Aucun
Objectif	Permettre à propriétaire d'obtenir un compte.
Pré condition	Page d'accueil affiché.
Post-condition	Compte créé.
Scenario Nominal	1. propriétaire demandé de créer un compte. 2. Le système affiche un formulaire à remplir. 3. propriétaire saisit ses informations (email, mot de passe,

	confirmation de mot de passe et …etc.) Et propriétaire envoie le formulaire. 4. le système vérifier le formulaire. 5. le système ajoute un utilisateur et envoyer le code. 6. le système afficher un message d'avancement et un champ à remplir. 7. Propriétaire remplit le champ. 8. Propriétaire envoie la réponse. 9. le système vérifier la réponse. 10. le système valider le compte d'utilisateur. 11. Le système affiche un message de succès.
Scenarios Alternatifs	3.a) Si Propriétaire saisit une information invalide le système affiche un message d'erreur. Aller à 2. 3.b) Formulaire vide. Aller à 2.

4.4 Diagramme de séquence de cas d'utilisation Gréer un compte d'utilisateur

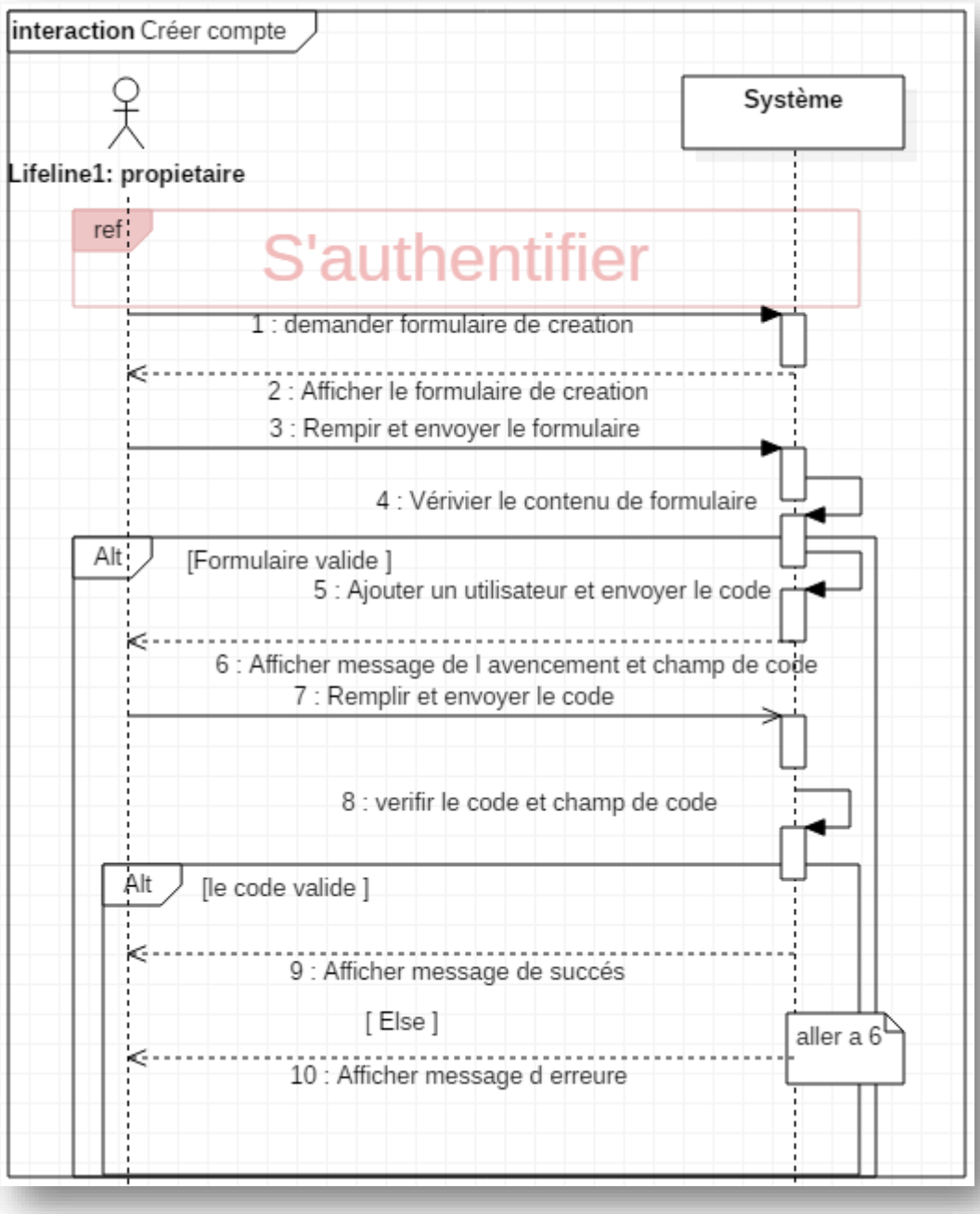

Figure 9 Diagramme de séquence 'créer un compte'

4.5 Description textuelle de cas d'utilisation ajouter les données

Cas d'utilisation	ajouter les données
Acteur Principaux	Propriétaire, l'utilisateur
Acteurs Secondaires	Aucun
Objectif	Dans ce cas, l'utilisateur peut ajouter les données dans la base de donner
Pré condition	Authentification réussie.
Post-condition	Les données ajouté à la base de donner. BD mise à jour.
Scenario Nominal	1. L'utilisateur veut ajouter les données. 2. Le système afficher le formulaire. 3. L'utilisateur Remplir le formulaire. 4. L'utilisateur envoie le formulaire. 5. Le système vérifier le formulaire. 6. Le système ajoute les données et MAJ la BD. 7. Le système affiche un message de succès.
Scenarios Alternatifs	5.a) S'il y a des champs vides : afficher message d'erreur et aller a 2.

4.6 Diagramme de séquence de cas d'utilisation ajouter les données

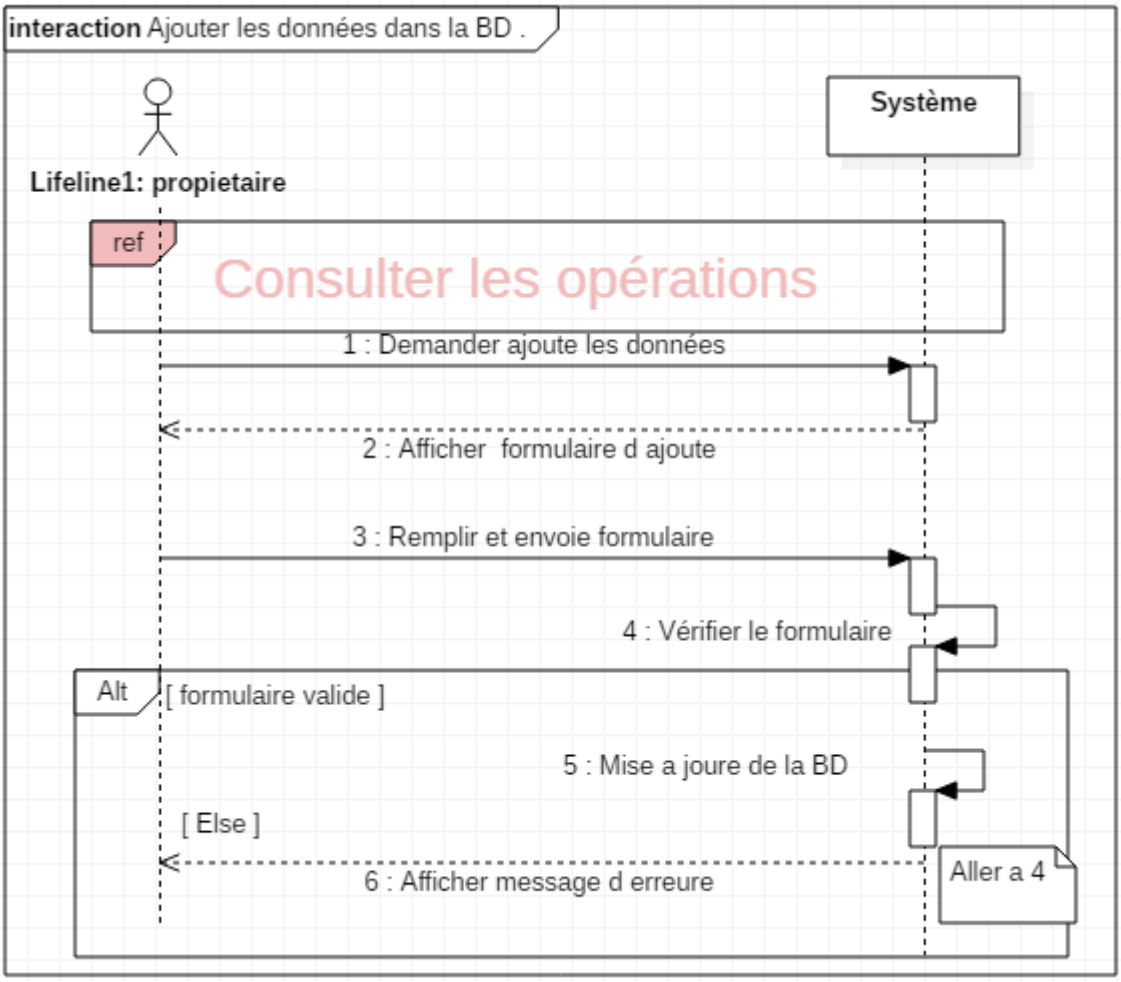

Figure 10 Diagramme de séquence 'ajouter de donner'

4.7 Description textuelle de cas d'utilisation Modifier les données

Cas d'utilisation	Modifier les données
Acteur Principaux	Propriétaire, utilisateur
Acteurs Secondaires	Aucun
Objectif	Dans ce cas, l'utilisateur peut modifier les données dans la base de donner
Pré condition	Authentification réussie.
Post-condition	Les données modifier dans la base de donner. BD mise à jour.
Scenario Nominal	1. L'utilisateur veut modifier une ligne ou plus. 2. Le système afficher la BD.

	3. le Propriétaire sélectionne la ligne. 4. le Propriétaire envoie son choix (pour clique sur bouton modifier). 5. Le système vérifie la sélection. 6. Le système affiche un formulaire avec les données actuel. 7. le Propriétaire modifie les champs de formulaire. 8. le Propriétaire envoie le formulaire 9. Le système vérifier le formulaire. 10. Le système enregistre la modification.
Scenarios Alternatifs	9.a) S'il y a des champs vides : afficher message d'erreur et aller a 6.

4.8 Diagramme de séquence de cas d'utilisation Modifier les données :

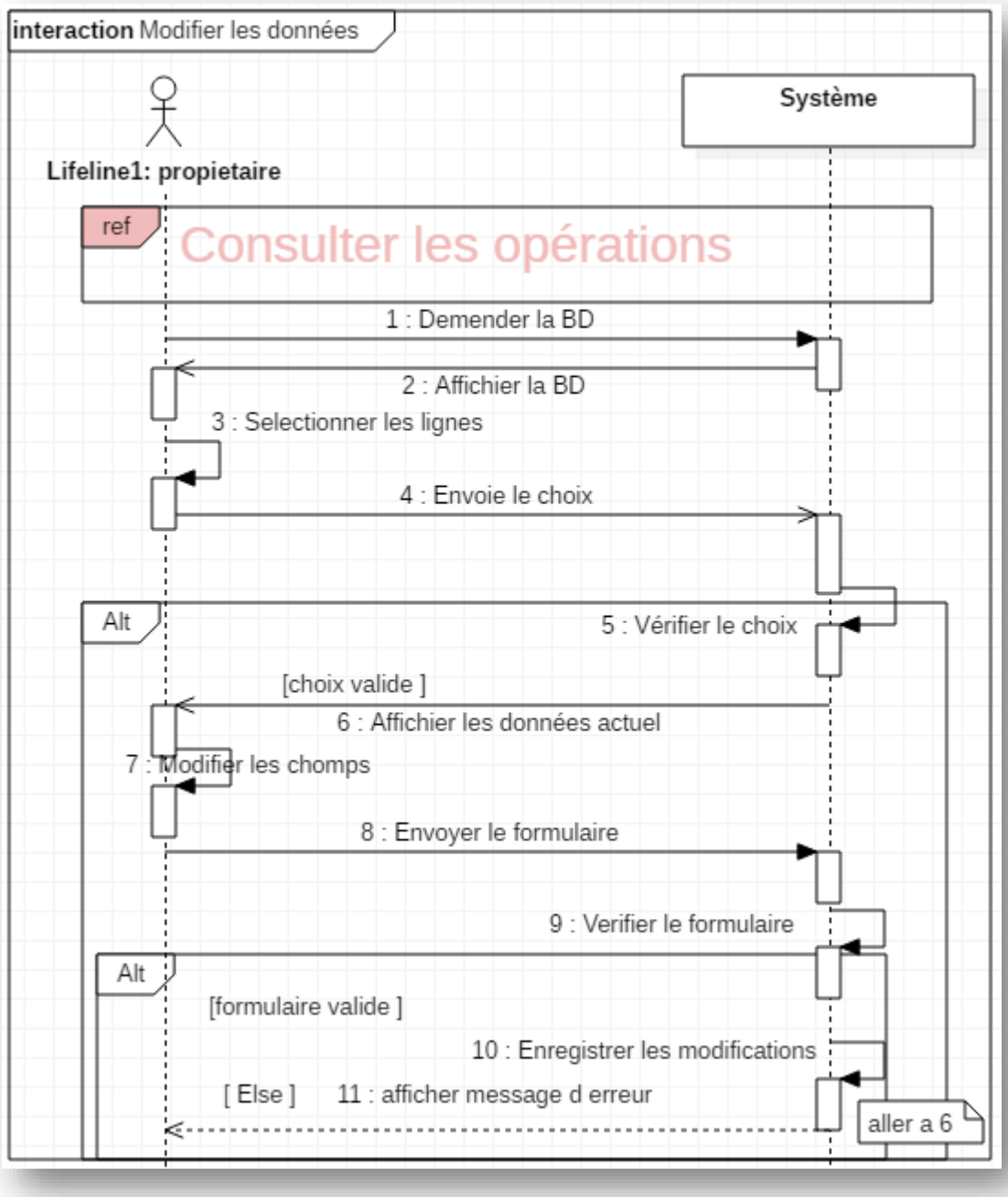

Figure 11 Diagramme de séquence 'modifier les données'

4.9 Description textuelle de cas d'utilisation Supprimer les données

Cas d'utilisation	Supprimer les données
Acteur Principaux	Propriétaire, utilisateur
Acteurs Secondaires	Aucun
Objectif	Dans ce cas, Propriétaire peut supprimer les données dans la base de donner
Pré condition	Authentification réussie.
Post-condition	Les données supprimer dans la base de donner. BD mise à jour.
Scenario Nominal	1. le propriétaire veut supprimer les données. 2. Le système affiche la BD. 3. le propriétaire sélectionne le donner. 4. le propriétaire envoie son choix. 5. Le système vérifie la sélection. 6. Le système afficher l'information d'entretien. 7. L'utilisateur demande la suppression. 8. Le système supprime le donner.
Scenarios Alternatifs	8.a) S'il y a des champs vides : afficher message d'erreur et aller a 3.

4.10 Diagramme de séquence de cas d'utilisation Supprimer les données

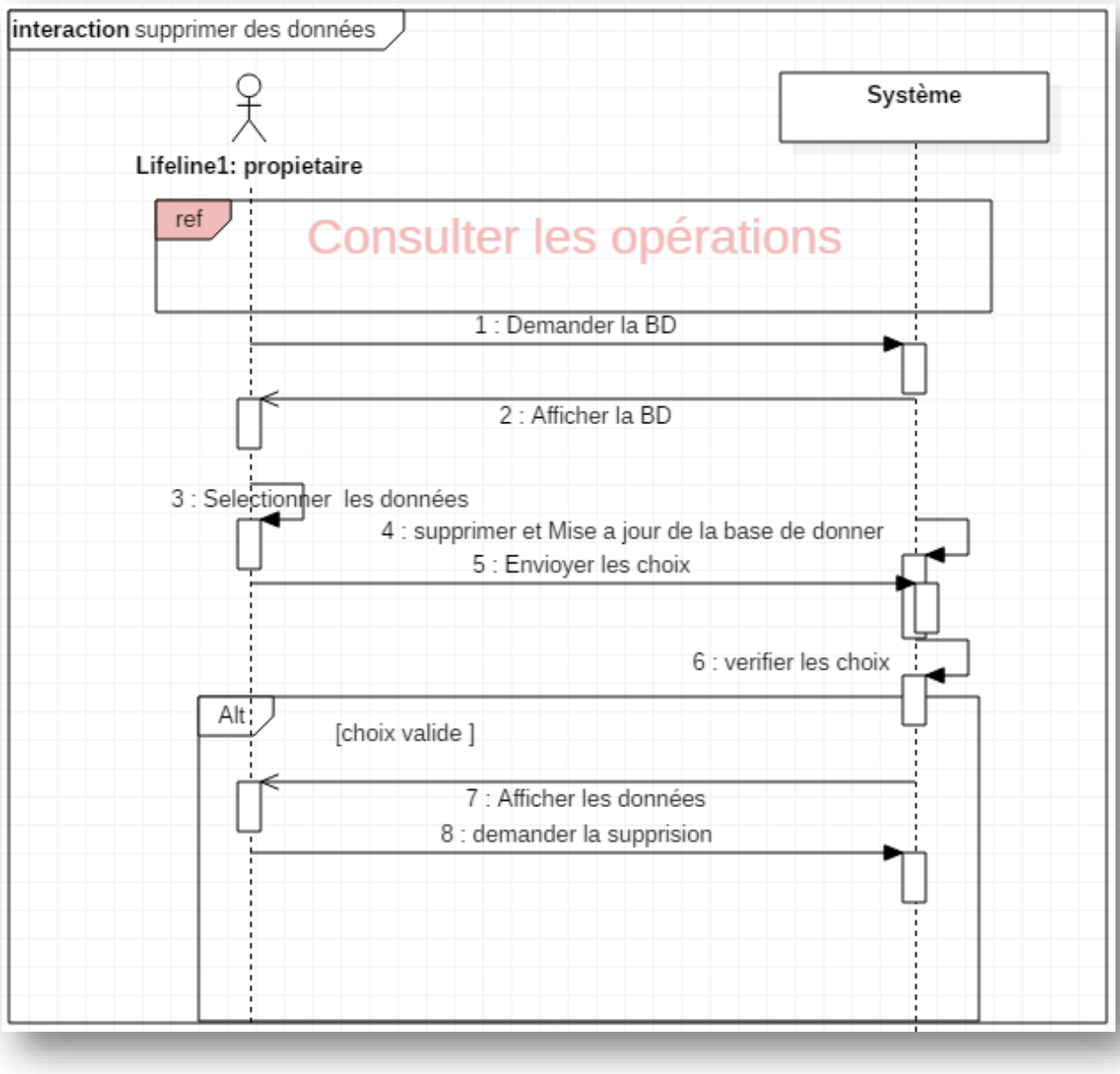

Figure 12 Diagramme de séquence 'supprimer les données'

4.11 Description textuelle de cas d'utilisation généralisé des données

Cas d'utilisation	Généralisé les données
Acteur Principaux	Propriétaire, utilisateur
Acteurs Secondaires	Aucun
Objectif	Dans ce cas, l'utilisateur peut généraliser les données dans la base de donner
Pré condition	Authentification réussie.
Post-condition	Les données généraliser dans la base de donner.
Scenario Nominal	1. L'utilisateur veut généraliser les données. 2. Le système afficher la BD. 3. L'utilisateur envoie son choix (clique sur bouton généraliser). 4. Le système vérifie la base sélectionné 5. Le système affiche les données généralisé. 6. Le système enregistre la base actuelle.
Scenarios Alternatifs	4.a) Si la base de donner est vides : afficher message d'erreur et aller a 2.

4.12 Diagramme de séquence de cas d'utilisation généralisation de donnée

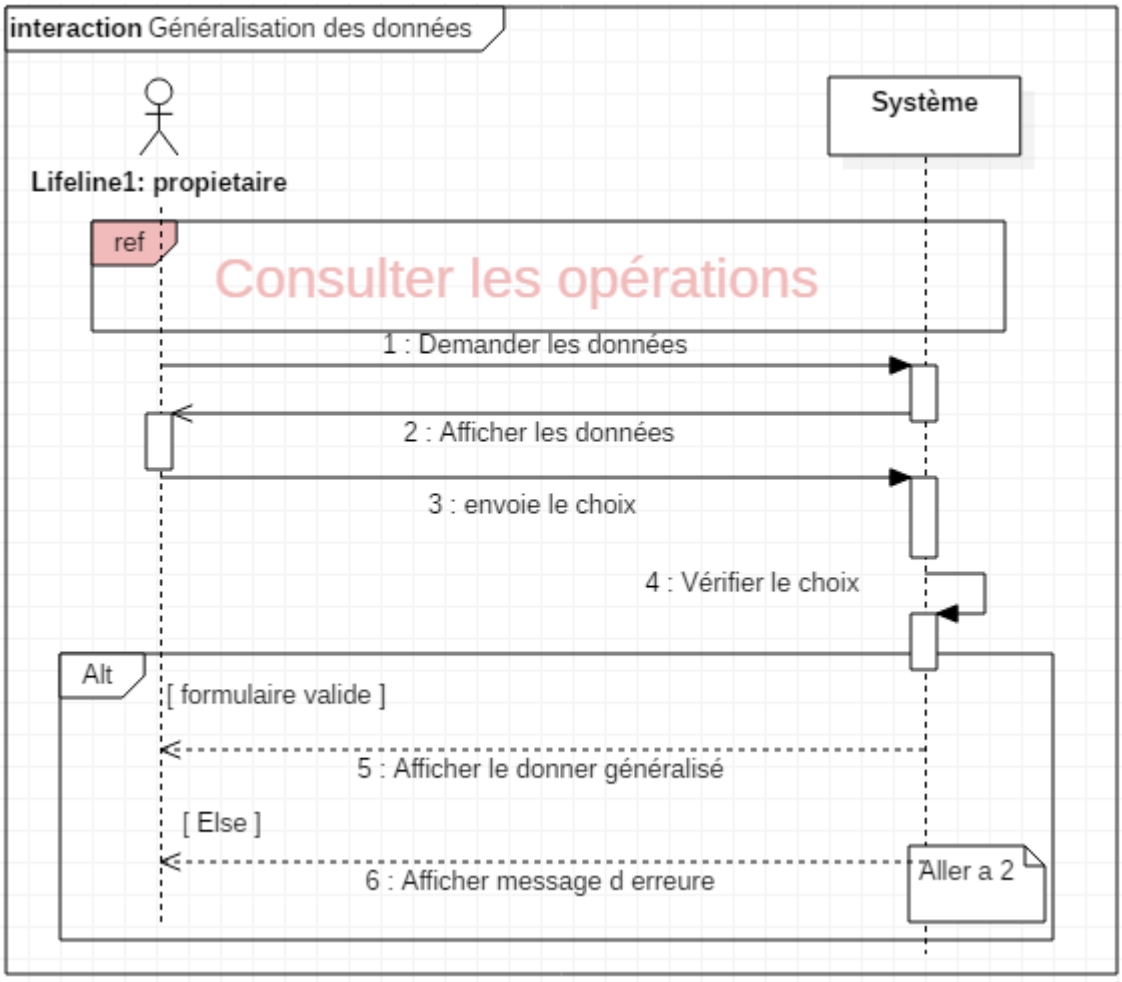

Figure 13 Diagramme de séquence 'Généraliser le donner'

4.13 Description textuelle de cas d'utilisation enregistre les données

Cas d'utilisation	enregistre les données
Acteur Principaux	Propriétaire, l'utilisateur
Acteurs Secondaires	Aucun
Objectif	Dans ce cas, l'utilisateur peut enregistre les données dans la base de donner
Pré condition	Authentification réussie.
Post-condition	Les données ajouté à la base de donner. BD mise à jour.
Scenario Nominal	1. L'utilisateur veut enregistre les données.

	2. Le système afficher un bouton. 3. L'utilisateur envoie son choix. 5. Le système vérifier le choix. 6. Le système enregistre les données dans la BD. 7. Le système affiche un message de succès.
Scenarios Alternatifs	5.a) S'il y a la table vide : afficher message d'erreur et aller a 2.

4.14 Diagramme de séquence de cas d'utilisation enregistre de donnée

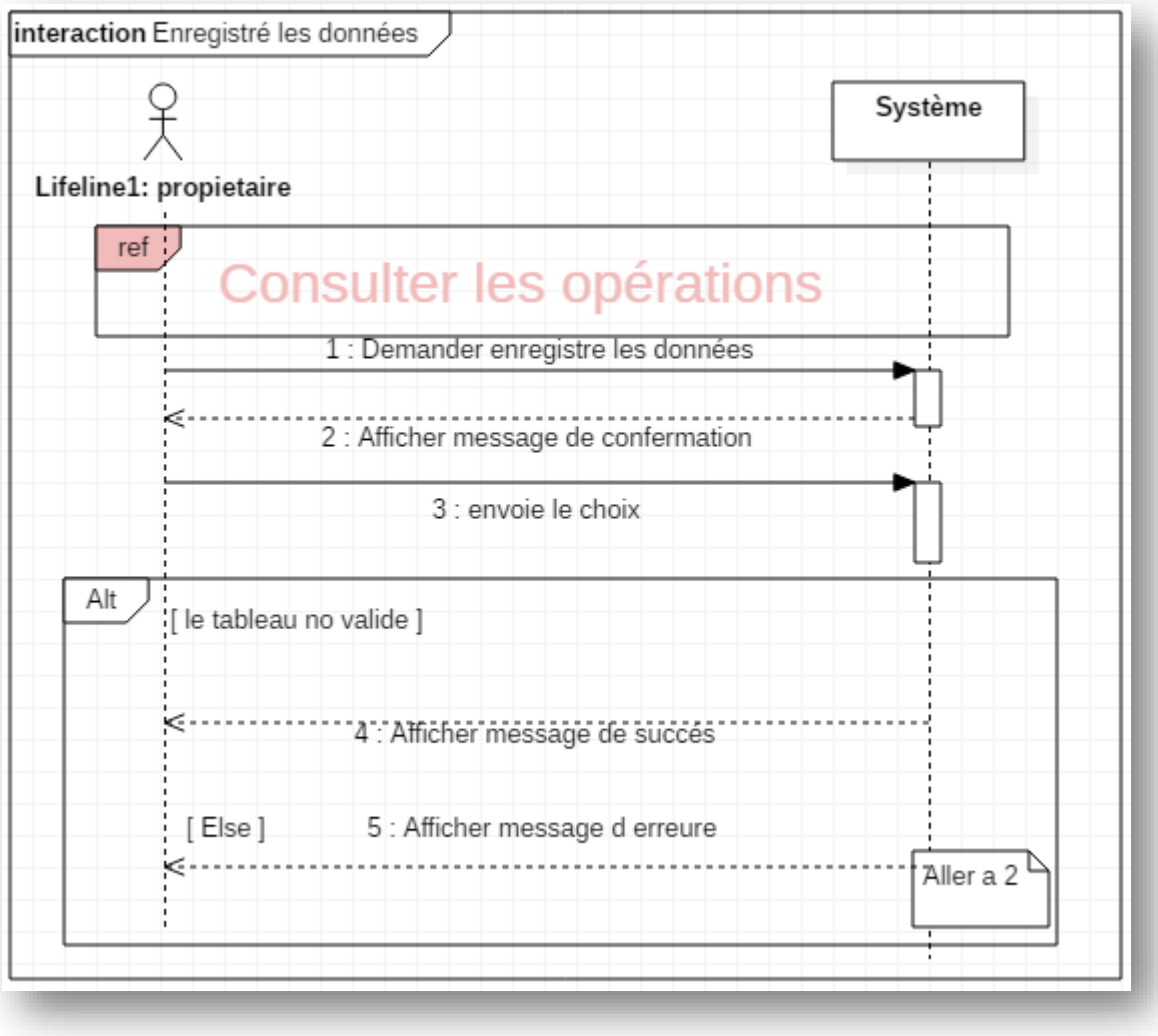

Figure 14 Diagramme de séquence de cas d'utilisation enregistre de donnée

Les Diagrammes des classes

Les diagrammes de clases sont sans doute les diagrammes les plus utilisés d'UML. Ils décrivent les types des objets qui composent un système et les différents types de relations statiques qui existent entre eux.

Les diagrammes de classes font abstraction du comportement du système.

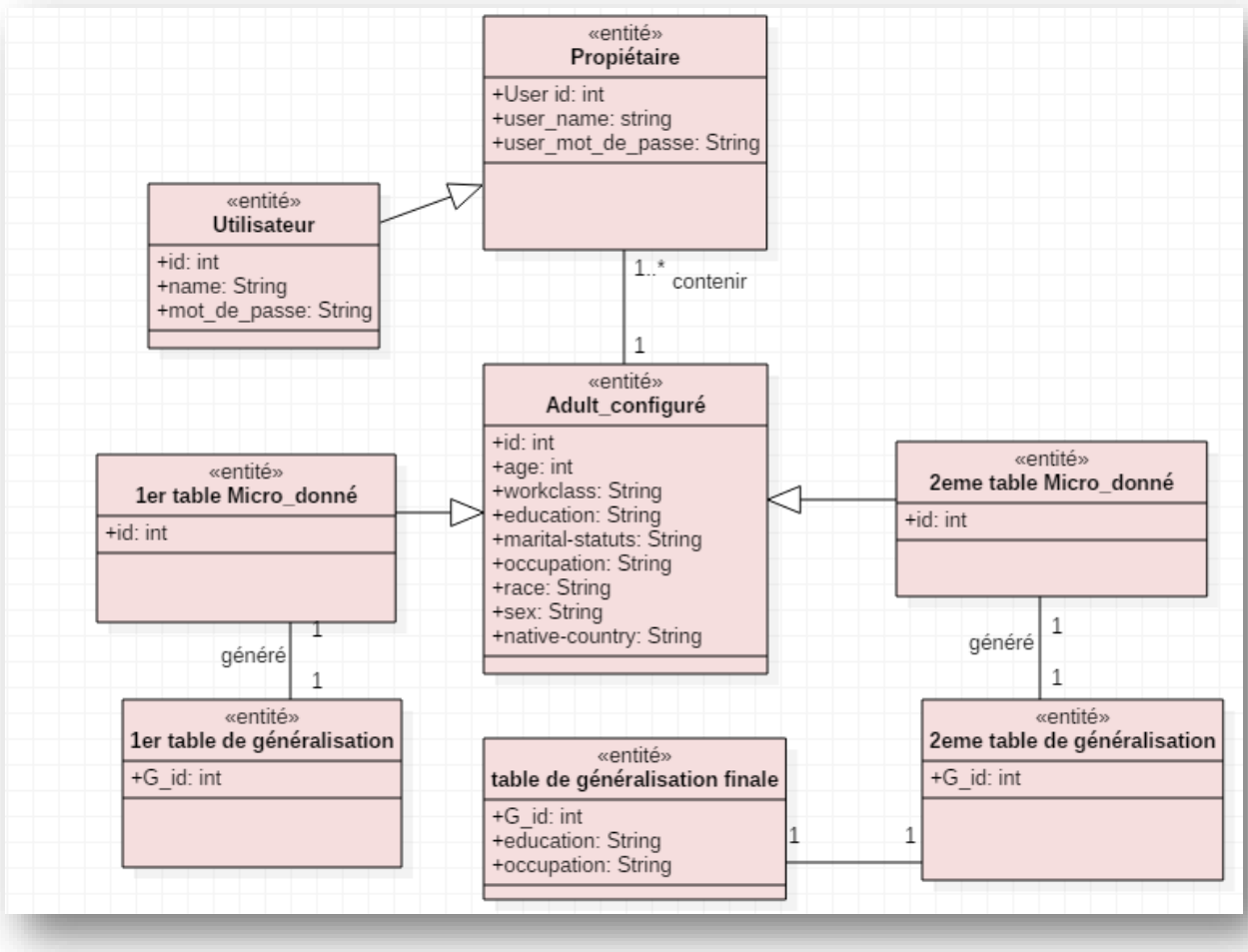

Figure 15 Diagramme de class

5. Passage du diagramme de classe au modèle relationnel

L'utilisation d'un SGBD impose un changement de représentation entre la structure des classes et celle structure des données relationnelles. Les deux structures ayant une analogie les équivalences exprimées dans un tableau sont utilisées pour en réaliser le rapprochement. Une classe définie une structure de données a laquelle souscrivant des instances elle correspond donc a une table du modèle relationnel :

- Chaque attribut donne lieu à une colonne
- Chaque instance stocke ses données dans une ligne (t_uplet) et son G_ID sert de clé primaire.
- Certains complexe ne correspondent a aucun des types SQL; on rencontre fréquemment ce cas pour les attributs représentant une structure de donnée.

Un type complexe peut être conçu :

- Soit avec plusieurs colonnes, chacune correspondant a une structure.
- Soit avec une table spécifique dotée d'une clé étrangère pour relier les instances aux valeurs de leur attribut complexe.

L'ensemble de règles utilisées pour le passage de l'orienté objet vers le relationnel est décrit ci-dessous :

Modèle objet	Modèle relationnel
Classe	table
Attribut de type simple	colonne
Attribut de type composé	Colonne ou clé étrangère
Instance	T_uplet
Héritage Clé primaire identique sur plusieurs tables	Association Clé étrangère ou table de liens
G_ID	Clé primaire

Dans une association 1 --- *

- Chaque classe devient une table, les attributs de la classe deviennent des attributs de la table et l'identifiant de classe devient la clé de la table.
- L'association est remplacé par une référence qui place l'identifiant de la classe à l'extrémité du 1 dans la table de la classe à l'extrémité du plusieurs * (ce sera une clé étrangère).

Dans une association * --- *

- Chaque classe devient une table, les attributs de la classe deviennent des attributs de la table et l'identifiant de classe devient la clé de la table.
- L'association (qui peut être une classe association) est remplacée par une table qui a comme clé la concaténation des identifiants des classes participantes. Dans le cas de classe association, les attributs sont rajoutés dans la nouvelle table.

Dans une association 1—1

Il y a différentes façons de d'implémenter selon les perspectives d'utilisation du concepteur :

- Règle1 Fusionner les deux classes dans une seule table, en gardant bien sûre la table la plus, importante sémantiquement.
- Règle 2 Garder les deux classes et les implémenter en deux tables. Sélectionner par la suite une table pour référencer l'autre par une clé étrangère. Ce qui revient à garder le sens de l'association qui est le plus important dans la conception (navigabilité).
- L'héritage L'implémentation de l'héritage demande un peu plus de considération pour choisir l'une des 03 règles de passage.
- Règle1 Beaucoup Conceptuellement parlant la classe mère est plus importante que les classes filles qui ne portent pas de spécificité informationnelle. Dans ce cas il faut garder la classe mère et l'implémenter par une table. Les classes filles, elles sont dégénérées et remplacées dans la table (classe mère) par un attribut.
- Règle 2 Conceptuellement parlant, Les classes filles sont plus importantes que la classe générique et sont porteuses d'information. La classe mère est dégénérée au profit des classes filles. Tous les attributs et opérations de la classe mère sont reportés au niveau des classes filles.
- Règle 3 Les classes mère et filles sont tout aussi importante et doivent être gardé, l'héritage est alors remplacé par une association 1 --1..* qui sera ensuite implémentée.

Nos relations

Adult_configuré (id, age, workclass,education,marital-status,occupation,race,sex,native-country **)**
Table Micro-donné (id, age, workclass,education,marital-status,occupation,race,sex **)**
Généralisation (id, age, workclass,education,marital-status,occupation,race,sex)
Table de generalization finale (G_id,education,occupation)

6. Conclusion

Dans ce chapitre nous avons présenté la méthode M-invariance et l'analyse et la conception du système de la préservation de la confidentialité pour les données dynamiques. En se basant sur le langage UML, en commencent par le diagramme le plus important qui est celui du cas d'utilisation ensuite nous représentons la structure de notre système avec le diagramme de séquence et diagramme de classe. En fin nous avons présenté les règles de passage relationnel.

Dans le prochain chapitre, nous allons présenter l'implémentation à l'aide des différents outils logiciels décrit dans le même chapitre

Chapitre 3

Réalisation

1. Introduction

Tout travail de recherche comporte deux parties : une partie théorique qui est expliqué au niveau du mémoire, et la partie pratique. Dans ce chapitre, nous allons présenter brièvement la structure de notre application, les langages de programmation et les outils utilisés pour la sa réalisation ainsi que la description des différentes interfaces.

2. L'Objectifs de l'application

L'objectif de ce projet, était de développer une application nécessaire à la préservation de la confidentialité pour les données publiées.

3. Environnement de développement

Dans cette partie on représente brièvement les interfaces de chaque logiciels qu'on a utilisé, on évoquera le système d'exploitions, le logicie PHP MyAdmin.

4. Système d'exploitation

L'environnement de base qui a constitué le support de notre travail est le système d'exploitation Windows 10.

Windows 10, offre la fiabilité et l'efficacité qui est optimisées .il en fait plus, en procurant un pouvoir de travail plus efficace.

5. StarUML

Est un logiciel de modélisation UML, cédé comme open source par son éditeur, à la fin de son exploitation commerciale, sous une licence modifiée de GNU GPL.

L'objectif de la reprise de ce projet était de se substituer à des solutions commerciales comme IBM Rational Rose StarUML gère la plupart des diagrammes spécifiés dans la norme UML 3.0.1.

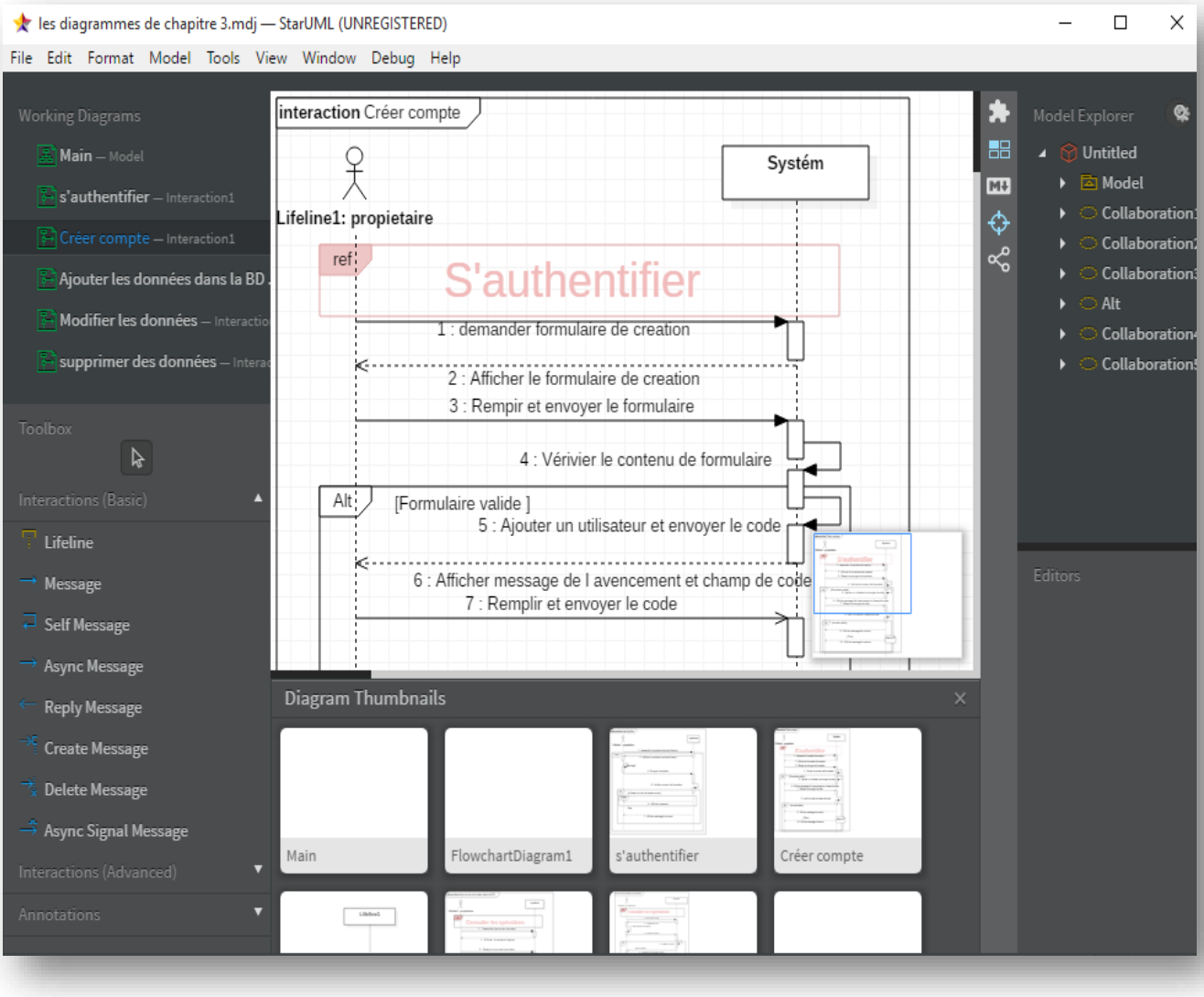

Figure 16 L'interface de StarUML

6. Xampp

XAMPP est un ensemble de logiciels permettant de mettre en place facilement un serveur Web et un serveur FTP. Il s'agit d'une distribution de logiciels libres (X Apache MySQL Perl PHP) offrant une bonne souplesse d'utilisation, réputée pour son installation simple et rapide. Ainsi, il est à la portée d'un grand nombre de personnes puisqu'il ne requiert pas de connaissances particulières et fonctionne, de plus, sur les systèmes d'exploitation les plus répandus.

Il possède également PHPMyAdmin pour gérer plus facilement vos bases de données.

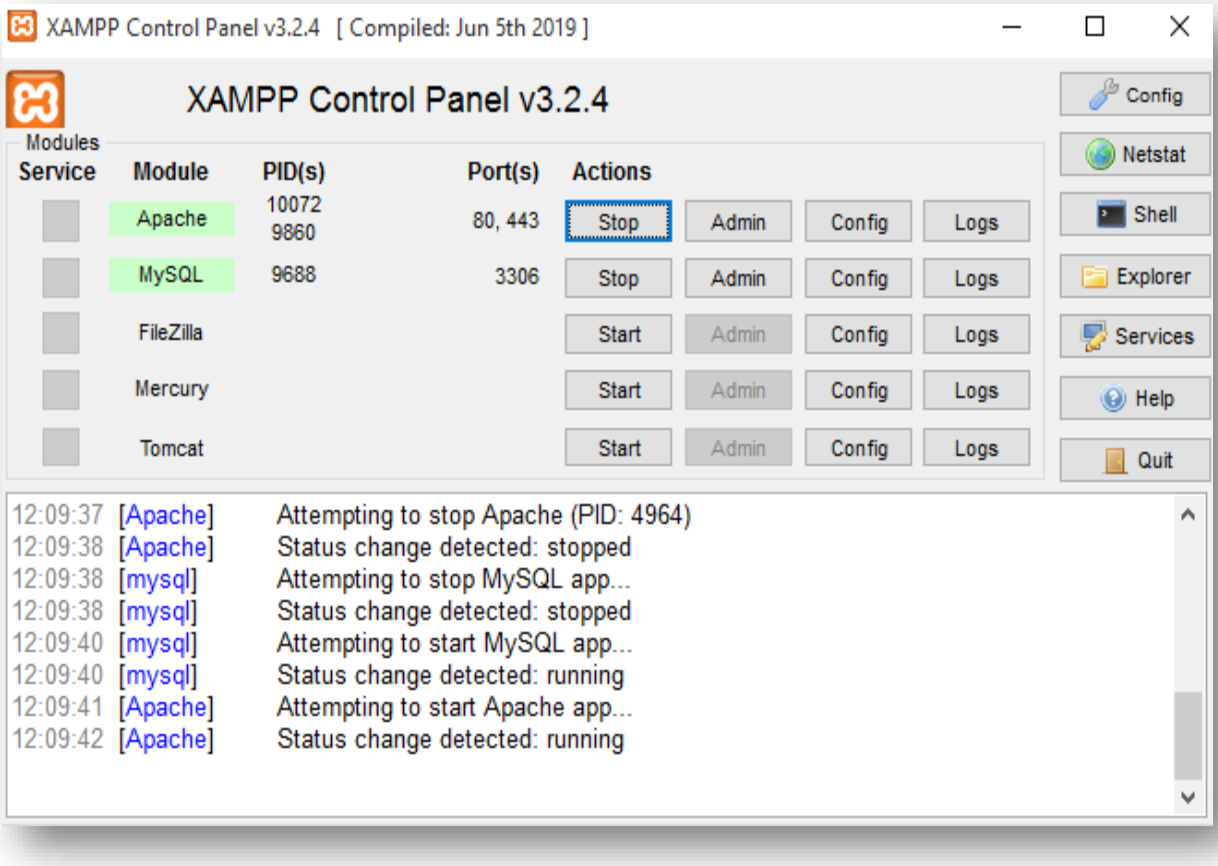

Figure 17 L'interface de Xampp

7. Net Beans 8.2

Net Beans est un IDE, Integrated Development Environment (EDI environnement de développement intégré en français), c'est-à-dire un logiciel qui simplifie la programmation en proposant un certain nombre de raccourcis et d'aide à la programmation. Il est développé par IBM, est gratuit et disponible pour la plupart des systèmes d'exploitation.

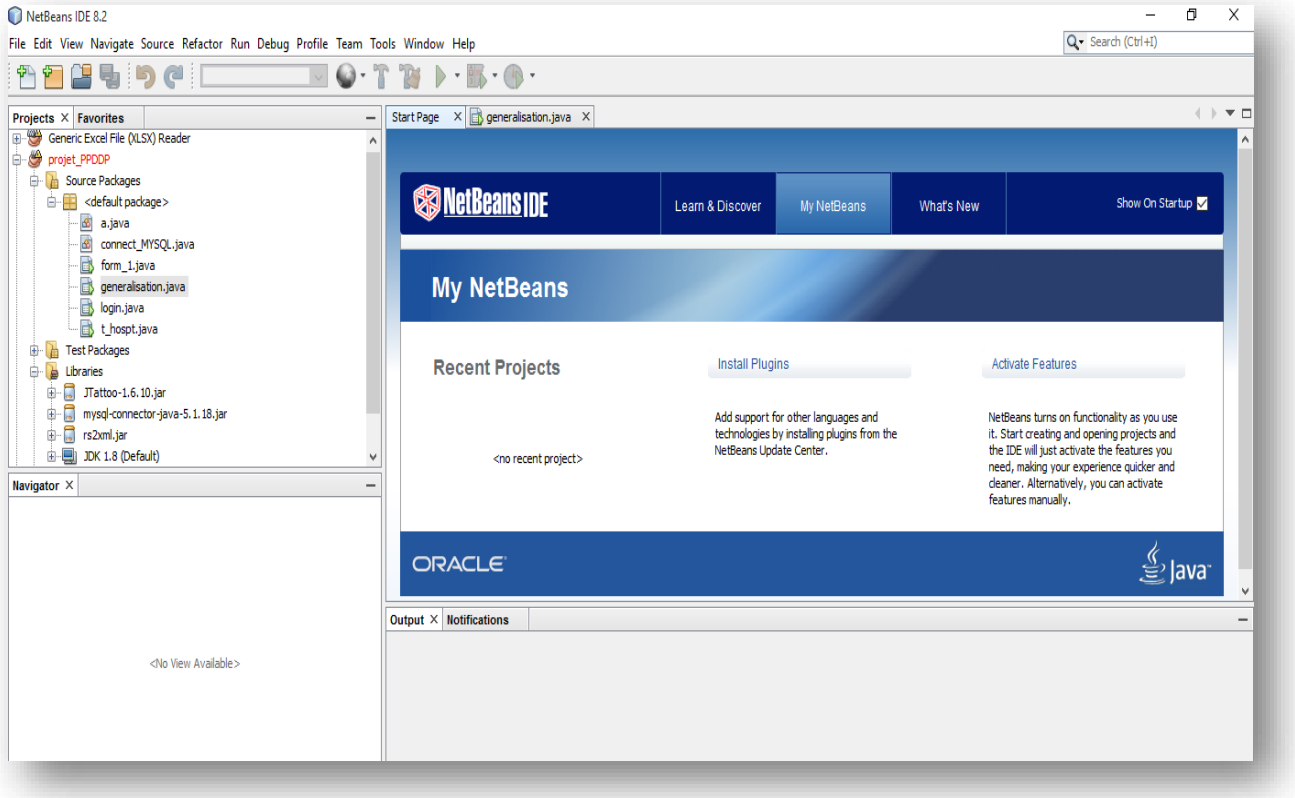

Figure 18 L'interface de NetBeans

8. Langage de programmation

8.1 Langage de programmation Java

Java est un langage de programmation orienté objet, développé par Sun Microsystems. Il permet de créer des logiciels compatibles avec de nombreux systèmes d'exploitation (Windows, Linux, Macintosh, Solaris). Java donne aussi la possibilité de développer des programmes pour téléphones portables et assistants personnels. Enfin, ce langage peut-être utilisé sur internet pour des petites applications intégrées à la page web (applet) ou encore comme langage serveur.

8.2 La base de donnée

Une base de données permet de stocker et d'organiser une grande quantité d'information. Les SGBD (Système de Gestion de Base de Données) permettent de naviguer dans ces données et d'extraire (ou de mettre à jour) les informations désirées au moyen d'une requête.

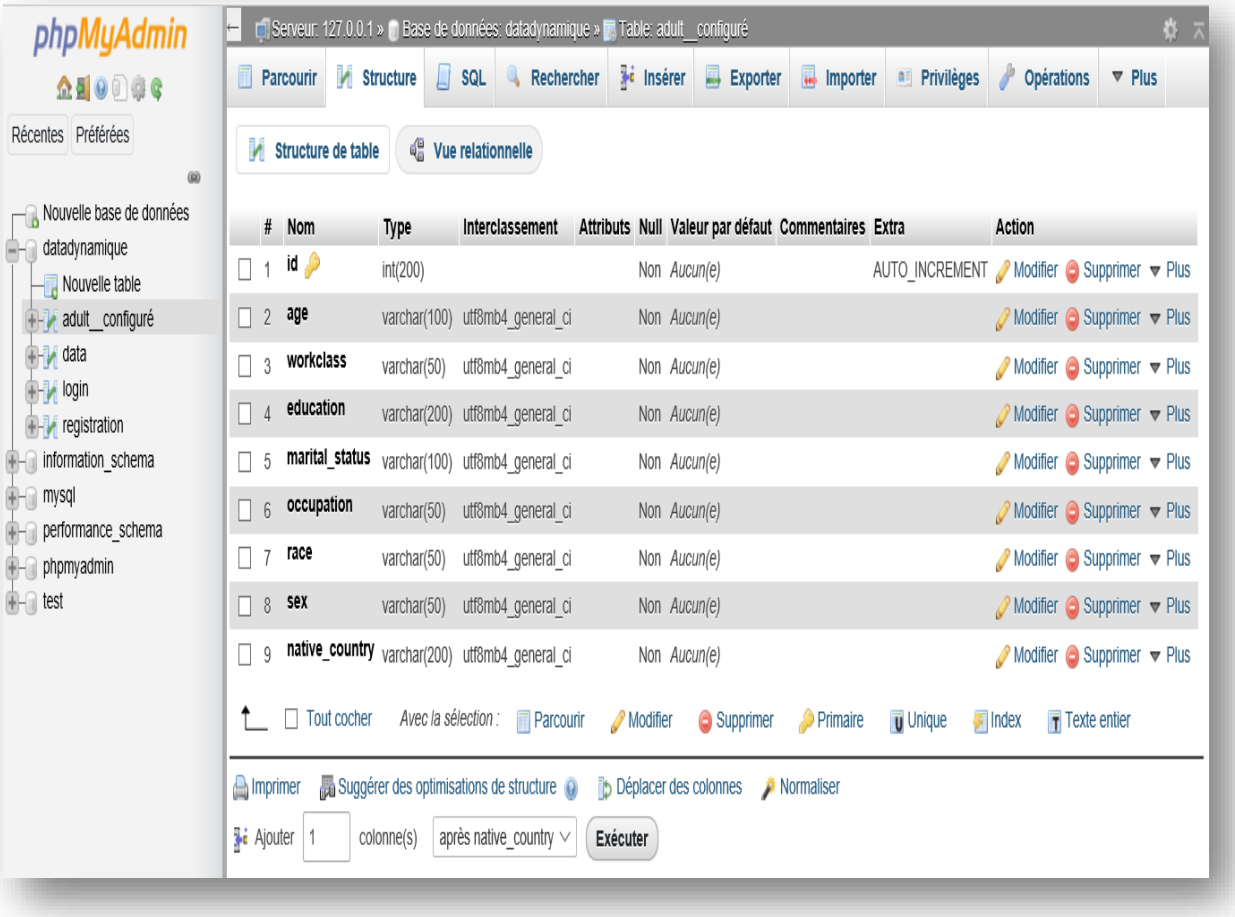

Figure 19 L'interface de la base de donner

8.3 SGBD

Un Système de Gestion de Base de Données (SGBD) est un **logiciel** (ou un ensemble de logiciels) permettant de manipuler les données d'une base de données. Manipuler, c'est-à-dire sélectionner et afficher des informations tirées de cette base, modifier des données, en ajouter ou en supprimer (ce groupe de quatre opérations étant souvent appelé "CRUD", pour Create, Read, Update, Delete).

Un SGBDR est un SGBD qui implémente la théorie relationnelle.

MySQL implémente la théorie relationnelle ; c'est donc un SGBD Relationnel (SGBDR).

MySQL est donc un SGBDR qui utilise le langage SQL. C'est un des SGBDR les plus utilisés. Sa popularité est due en grande partie au fait qu'il s'agit d'un logiciel Open Source, ce qui signifie que son code source est librement disponible et que quiconque qui en ressent l'envie et/ou le besoin peut modifier MySQL pour l'améliorer ou l'adapter à ses besoins.

9. Présentation de l'application

Dans cette partie, on va présenter quelques interfaces utilisateur sélectionnées parmi l'ensemble des interfaces constituant notre application :

9.1 La page d'authentification

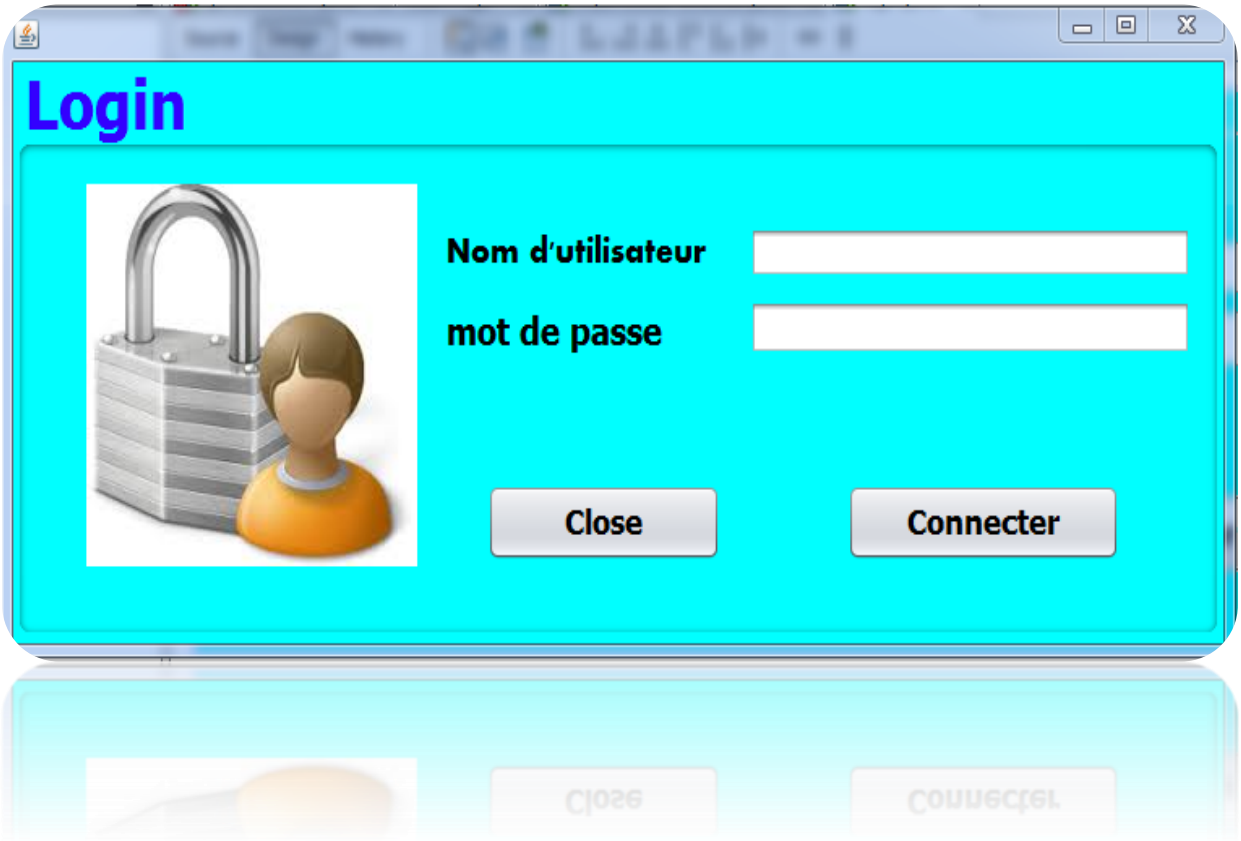

Figure 20 l'interface de la page d'authentification

Lorsque l'utilisateur remplir le formulaire et puis cliqué sur le bouton connecter , une nouvelle fenêtre apparait. Elle contient, les formulaires pour la saisie des données puis appliquer quellque opperations sur les données.

saisies manuellement de donnée :

Après avoir fait authentification alors apparait un ensemble de boutons, comme il est montré dans la figure cet ensemble contient les boutons suivants :

- Modifier et supprimer, ajouter.

- Enregistre le tableau
- Généralisation et M-invariance

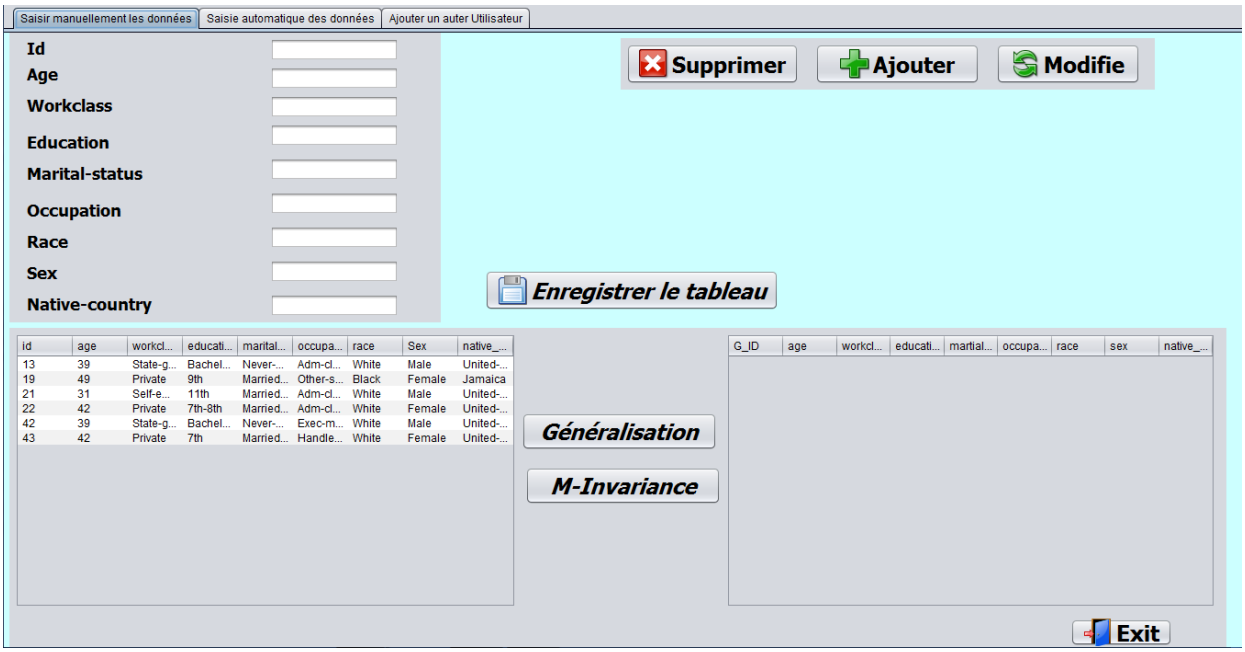

Figure 21 l'interface saisie manuellement de donnée

L'utilisateur peut supprimer les données.

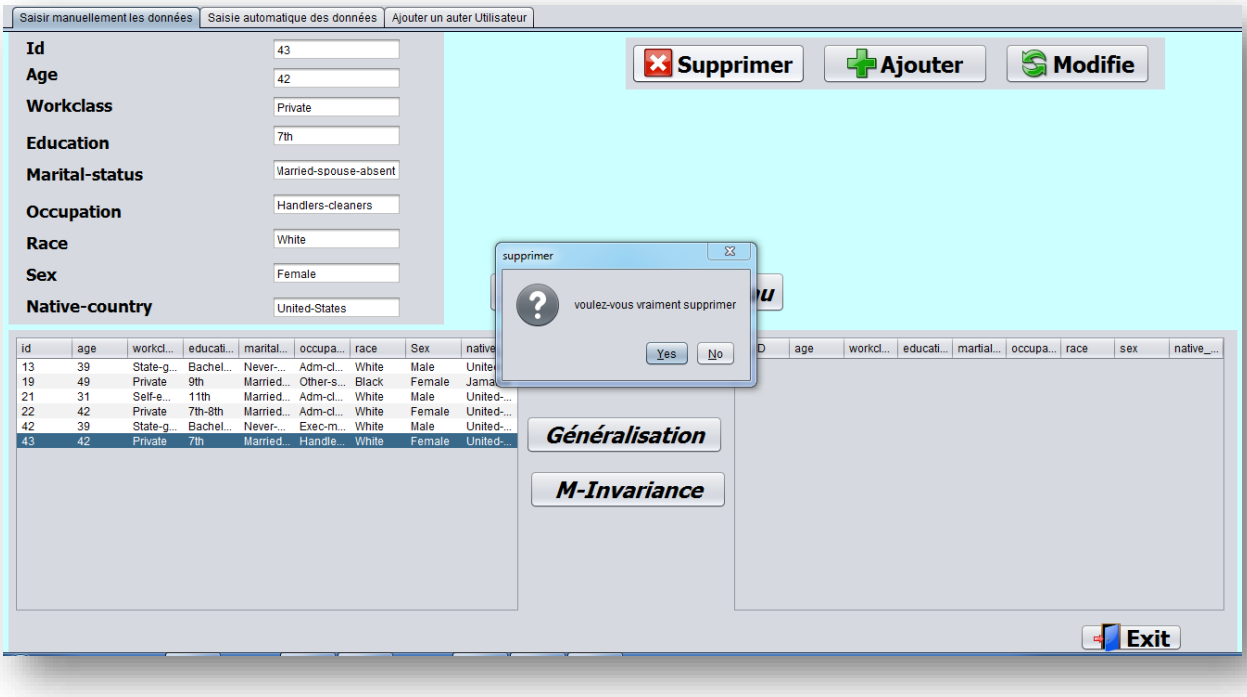

Figure 22 l'interface de suppression le donner

L'utilisateur lorsque clique sur le bouton Généralisation, une nouvelle table apparait a la droite. Elle contient Les données généralisées comme il est montré dans la figure.23

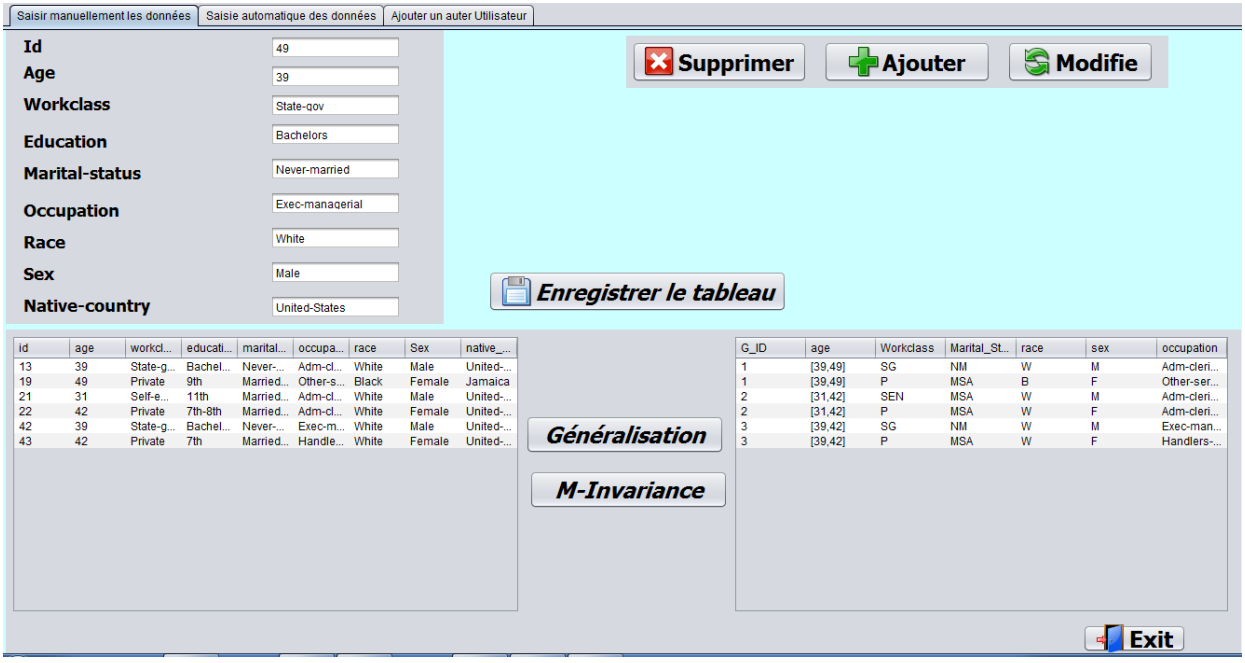

Figure 23 L'interface après de généralisation

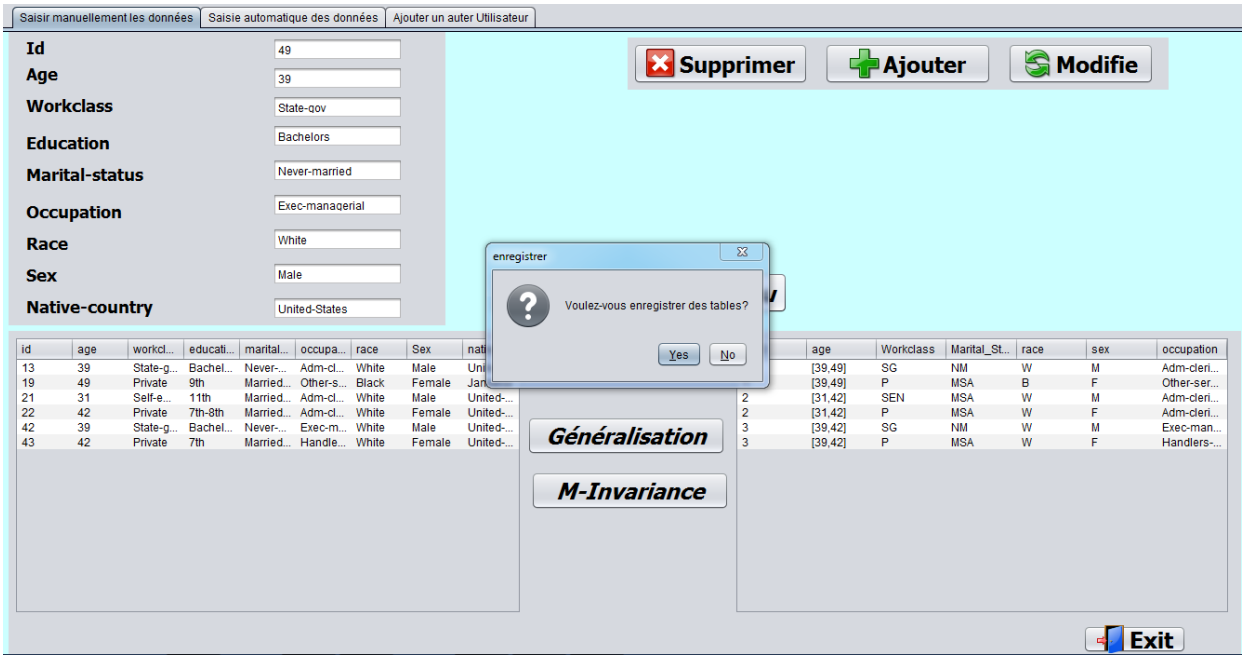

Figure 24 L'interface de enregistre les tableaux

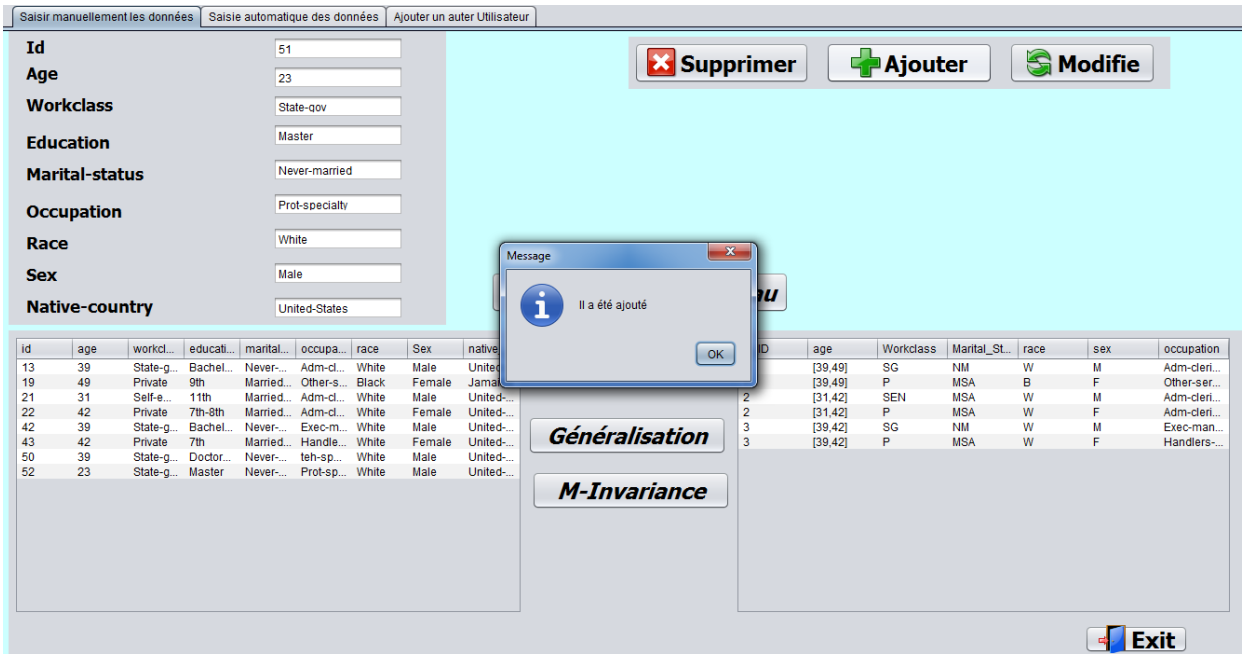

Figure 25 L'interface de modification de la table

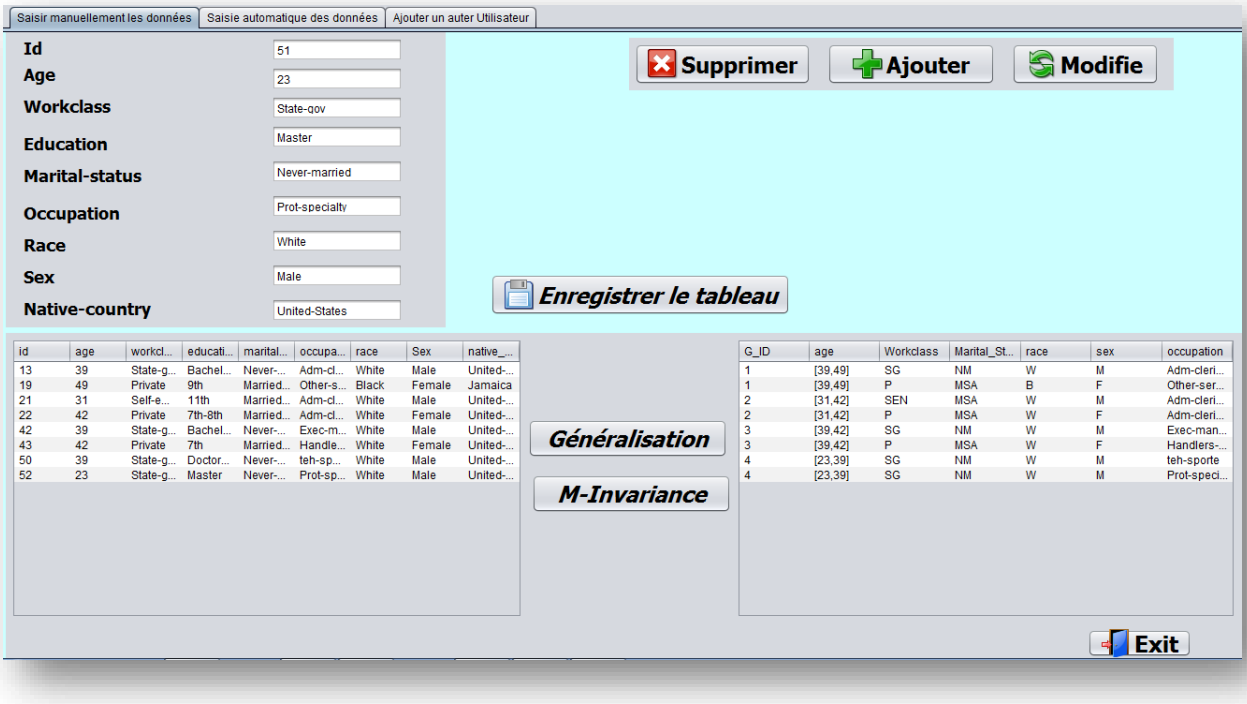

Figure 26 L'interface de généralisation de la table modifiée

L'utilisateur lorsque clique sur le bouton M-invariance, une nouvelle table apparait a la droite, comme il est montré dans la figure.24

Figure 27 L'interface après de fait de M-invariance

L’utilisateur peut aussi importe des données

L’utilisateur lorsque clique sur le bouton attacher, une nouvelle fenêtre apparait. Elle contient, le ficher de votre ordinateur.

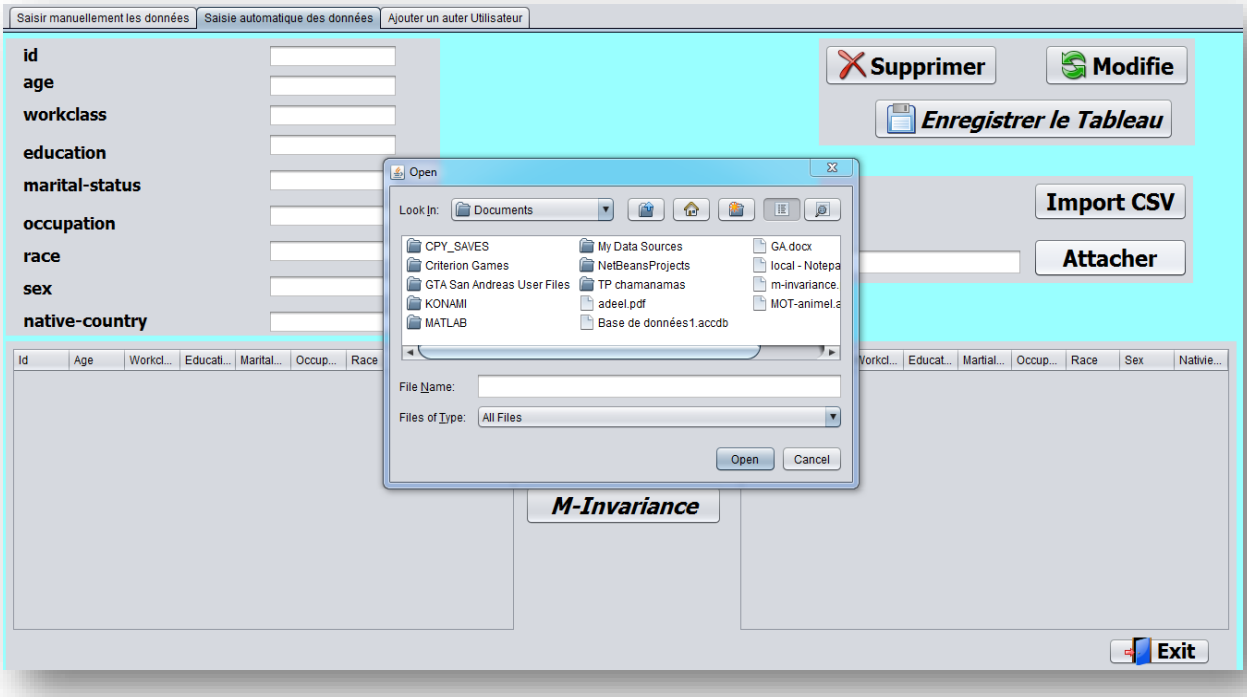

Figure 28 l’interface de importer le donnée

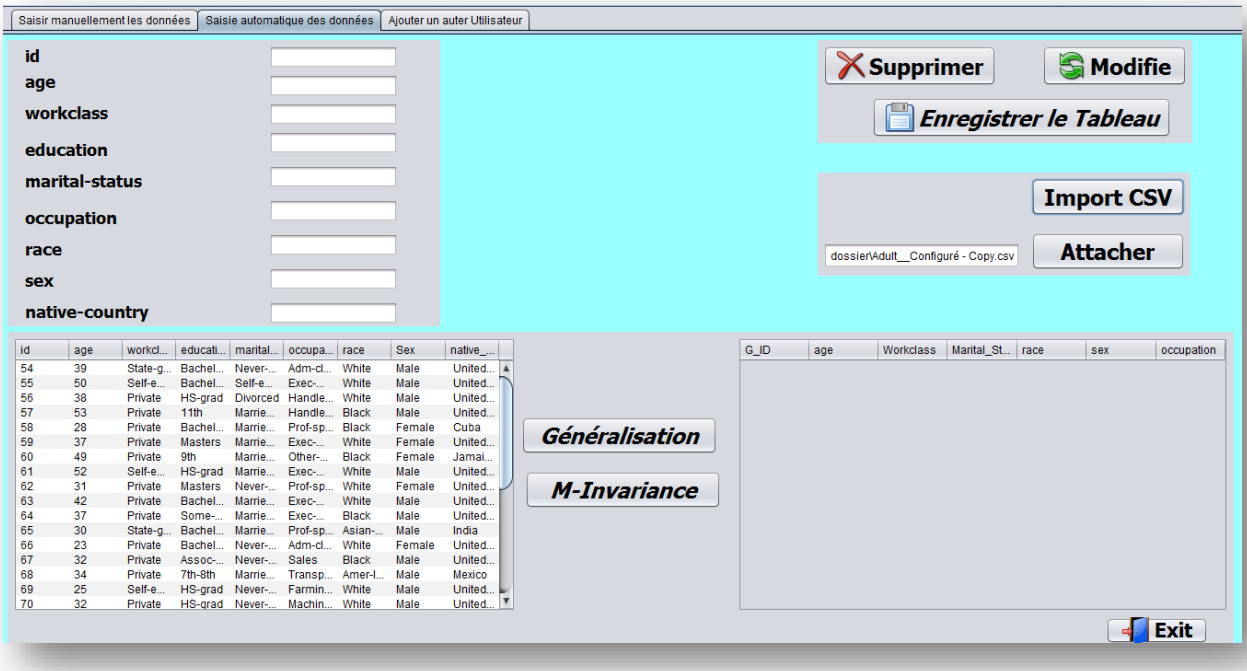

Figure 29 l'interface après de l'importation des données

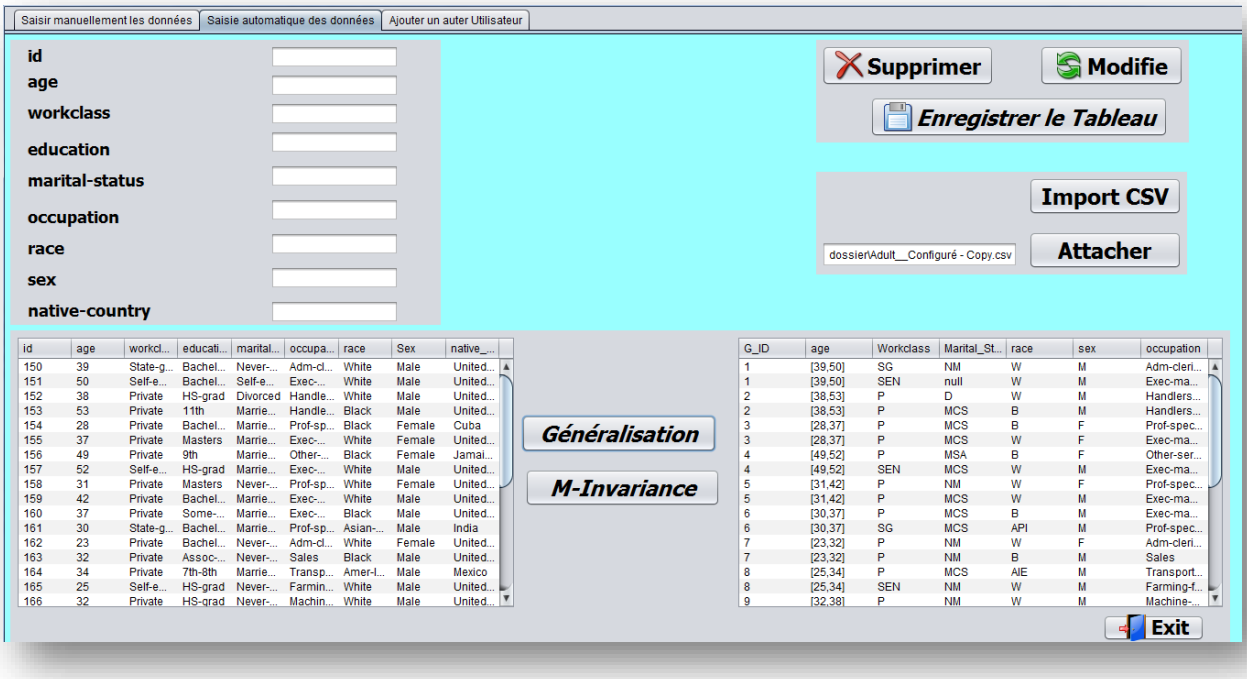

Figure 30 l'interface de généralisation de donnée importée

Administrateur peut ajouter et modifier, supprimer un utilisateur

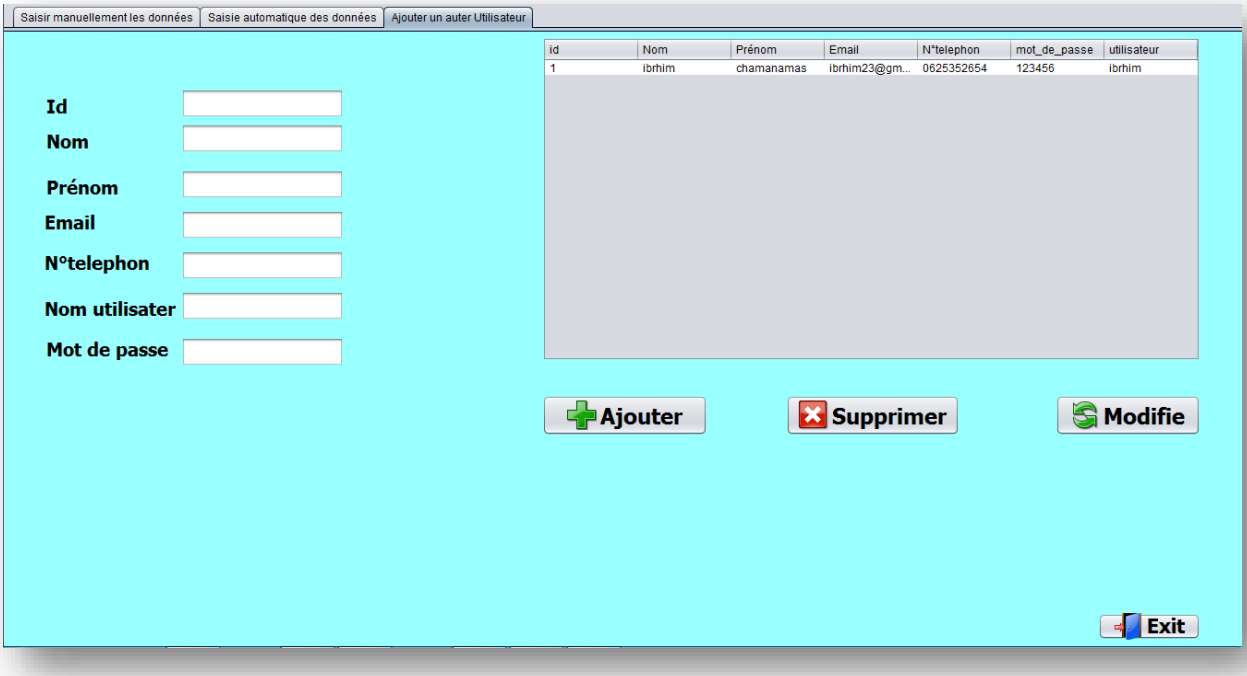

Figure 31 L'interface de ajouter un utilisateur

Discussion

Généralement, les tables de micro données d'origine ne satisfait pas les exigences spécifiques de la confidentialité et les tableaux doit être modifié avant d'être publié, la modification est faite par l'application des séquences d'opérations d'anonymisation. Ces opérations comprennent la généralisation, la suppression, la dissimulation et la permutation

L'anonymisation des données basée sur le modèle k-anonymisation a été largement étudiée au cours des dernières décennies. Cependant, dans notre étude, nous avons trouvé K-anonymisation limitée à un scénario où l'ensemble de données est supposé disponible au moment de la publication.

En d'autres termes, une grande partie du travail effectué sur le modèle K-anonymisation se concentre sur les données statiques. Cette hypothèse conduit à de graves lacunes en matière d'utilité et de confidentialité, car la collecte de données est actuellement effectuée en continu (et donc en constante augmentation) et il y a une demande croissante fréquente de données modernes.

La publication continue des données est plus compliquée. Les données de ce type de publication peuvent être modifiées en entrant de nouveaux enregistrements et en supprimant certains enregistrements précédents.

La principale raison de l'échec des modèles précédents (statiques) est que ces méthodes ne prennent en compte aucune limitation des valeurs sensibles dans les classes de parité.
Et que l'ensemble des valeurs sensibles dans toutes les versions continues est le même.

10. Conclusion

Ce chapitre décrit la finalisation de notre application qui permet de préservation de la confidentialité pour les données publiées, a travers l'implémentation des données de notre système qui été conçue selon les besoins des utilisateurs, et réalisée avec le langage JAVA sous l'environnement Eclipse.

Conclusion générale

L'utilisation et le partage des données collectées sont limités en raison de la présence d'informations personnelles identifiables dont la confidentialité des individus peut être violée lors d'un tel partage. La difficulté dans le partage des données provient principalement du fait que les données souvent ne sont pas statiques (il existe toujours des multiples pubications).

Dans ce mémoire, nos travaux de recherche ont porté sur la conception et l'implémentation d'une technique d'anonymisation (m-invariance) qui satisfait les exigences de la publication dynamique des données

Dans un premier temps, on a présenté une généralités sur la préservation de la confidentialité pour les données publiées. Au début, on a commencé par citer les définitions important qui concerne l'approche de l'anonymisation, les différents types, et les opérations sur les quelles, cette approche va se dérouler parfaitement.

On a passé par la suite à réaliser une étude analytique formelle qui clarifie la base théorique de la technique M-invariace qui vise à pallier tous les problèmes causés par les modèles de la préservation de confidentialité pour les données statique.

Ensuite et pour valider les contributions proposée, on a entamé la procédure d'expérimentation fournit par la méthode m-invariace

Références bibliographiques

[1] Salheddine kabou, Sidi mohammed benslimane, " La gestion de la confidentialité dans le cloud computing", Thèse de Doctorat, Université de Sidi Belabes, 2017.

[2] http://www.ericsson.com/research-blog/data-knowledge/ big-data-privacy-preservation/2015.

[3] Samarati P. Protecting respondent's privacy in microdata release. IEEE Trans Knowl Data Eng.

[4] http://www.ericsson.com/research-blog/data-knowledge/ big-data-privacy-preservation/2015.

[5] Sweeney L. K-anonymity: a model for protecting privacy. Int J Uncertain Fuzz

[6] Sweeney L. K-anonymity: a model for protecting privacy.

[7] http://www.ericsson.com/research-blog/data-knowledge/ big-data-privacy-preservation/2015

[8] Byun,J., Sohn,Y., Bertino,E., Li,N. (2006). Secure anonymization for incremental datasets. In Proc. of the VLDB Workshop on Secure Data Management (SDM),pp.48- 63

[9] Challenging More Updates: Towards Anonymous Re-publication of Fully Dynamic Datasets

[10] Challenging More Updates: Towards Anonymous Re-publication of Fully Dynamic Datasets Department of Computer Science and Engineering, Fudan University

[11] m-Invariance: Towards Privacy Preserving Re-publication of Dynamic Datasets Department of Computer Science and Engineering Chinese University of Hong Kong

[12] Towards Privacy Preserving Re-publication of Dynamic Datasets
Xiaokui Xiao Yufei Tao Department of Computer Science and Engineering Chinese University of Hong Kong

[13] Chang, V., Kuo, Y. H., & Ramachandran, M. "Cloud computing adoption framework: A security framework for business clouds." Future Generation Computer Systems, 57, 24-41. 2016

Site web

[14] https://www.cnil.fr/en/node/114427 L'anonymisation des données, un traitement clé pour l'open data 17 October 2019.