République Algérienne Démocratique Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université d'Ibn Khaldoun – Tiaret

Faculté des Mathématiques et de l'Informatique

Département Informatique

Thème

La sécurisation d'un Système de Filtrage d'Information

Pour l'obtention du diplôme de Master

Spécialité : Génie Informatique

Option : Système d'information

Dirigé par : Mdm. Lalia Benathmane

Réalisé par : Hocine Mediouni

Année Universitaire : 2015-2016

Remerciements

*Avant toute chose je tiens à remercier le grand « **Dieu »** de nous avoir donné le courage et la volonté qui nous ont permis de réaliser ce modeste travail.*

*Mon remerciement particulier à **Mdm Benatman Laila** pour ses remarques pertinentes et son optimisme qui nous fait parfois défaut. En outre ses qualités d'encadreur, m'a toujours permis d'avancer à un rythme régulier dans mon travail.*

J'exprime toute ma gratitude à mes examinateurs Monsieur …….. et …… pour avoir accepté d'examiner ce travail et leurs participations au jury.

Merci également à l'ensemble des enseignants du département Informatique Université Tiaret.

Enfin, je ne saurais terminer ces remerciements sans y associer toute personne qui, de près ou de loin, m'a apporté son aide ou sa sympathie.

Dédicace

Je dédie ce modeste travail très spécialement à mes chers parents pour tout ce que vous avez fait pour moi. Je ferai de mon mieux pour rester un sujet de fierté à vos yeux avec l'espoir de ne jamais vous décevoir.

À mes frères ma sœur et toute ma famille,

A mes très chers amis, J'espère de tout mon cœur que notre amitié durera éternellement.

A tous mes professeurs et les collèges d'études.

Liste de figures :

Sommaire :

Introduction générale :

La sécurité des applications informatique et des bases de données en particulier est devenue une priorité pour les citoyens ainsi que pour les administrations. Le besoin de partager et d'analyser des données personnelles est multiple : pour rendre plus simples et efficaces les procédures administratives et pour personnaliser les services dans un environnement spécifier.

Dans le cadre de ce PFE, on s'intéresse à la sécurité des bases de données dans les systèmes de filtrage d'information.

Notre objectif est de proposer des mécanismes de contrôles d'accès afin d'améliorer la sécurité d'une application informatique. Nous avons réalisé une application qui fait appel à des mécanismes des sécurités dans les bases de données.

Cette application contient deux parties : la première est réaliser afin de gérer les fonctionnalités de l'Administrateur, à ce stade nous avons proposé un contrôle d'accès sur les nombreux opérations réaliser par l'**admin** et on a intégré un mécanisme de chiffrement des données pour assurer la protection des données contre toutes attaques.

La deuxième partie de notre application est une implémentation des droits d'accès des utilisateurs **(user)** de cette deuxième partie, nous avons proposé un contrôle d'accès sur le nombre de tentatives erronés d'authentification ainsi qu'un mécanisme de chiffrement afin de protéger les informations pertinentes de la base de données.

Ce travail est structuré comme suit :

Le premier chapitre est consacré à présenter le filtrage d'information, nous avons détaillé les systèmes de filtrages d'informations. Le deuxième chapitre présent les bases de données, nous avons détaillé vers les SGBD les opérations sur les données et les différents types de base de données .Le troisième chapitre présent la sécurité dans les bases de données, ainsi que nous avant traité les moyens de sécurité selon les différents modèles de contrôle d'accès, et la protection d'une base de données.

Le quatrième chapitre présent est décomposé en deux parties :

La modélisation de notre application qui intègre des contrôles d'accès que nous avons proposés. La modélisation est basée sur le langage UML (Unified Modelisation Language). Notre application est basée sur deux types de fonctionnalités : la première implémente le rôle Administrateur et la deuxième pour gérer l'utilisateur avec pour chacune des propositions pour la gestion de la sécurité.

Et l'implémentation de notre application et la démarche de chaque fonctionnalité réalisée.

Chapitre 1 : Filtrage d'information

Introduction :

Ces dix dernières années, les systèmes de filtrage d'information ont connu une avancée significative, depuis les premiers systèmes collaboratifs classiques à ceux à base de contenu ou hybrides. Ils ont été largement investis dans divers domaines tels que le commerce électronique (livres, cinéma, musique, voyages, restauration, etc.….).

Dans ce chapitre nous commençons par présenter un aperçu global du filtrage l'information et ses techniques en particulier le filtrage collaboratif.

I. Système de filtrage d'information:

1. Définition :

Le filtrage d'information est un processus dual à la recherche d'information comme le montre N. Belkin dans **[1].** Il traite des documents provenant de sources dynamiques (News, Email, etc.) et décide a la volée, si le document correspond ou pas aux besoins en information des utilisateurs, besoins exprimes au travers du concept de profils utilisateurs. Dans les deux cas, l'objectif est de sélectionner les informations répondant aux besoins en information des utilisateurs**[1].**

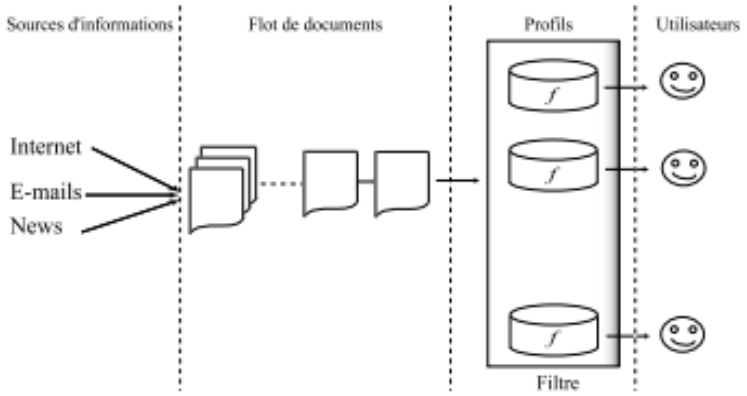

Figure I.1 : Principe de filtrage d'information**[1].**

Un SFI peut donc être vu comme un assistant personnel qui permet a des utilisateurs, ayant défini préalablement leur(s) centre(s) d'intérêt, de recevoir des documents pertinents provenant de sources dynamiques.

2. Profil [2] :

Un profil peut être modélise par différents types d'information permettant de caractériser un utilisateur ou un groupe d'utilisateurs. Ces types d'informations sont définis selon le contexte dans lequel le profil est utilise. On peut trouver par exemple des informations sur ces centres d'intérêts, des préférences, des connaissances sur l'utilisateur, etc.

Différentes formes de construction d'un profil ont été proposées dans la littérature. Un profil peut être construit de façon explicite `a travers une liste de termes pondères établie par l'utilisateur, de façon supervisée par le système en recueillant les jugements de l'utilisateur sur les documents déjà reçus ou d'une façon implicite par observation du comportement de l'utilisateur lors de ses interactions avec le système**.**

3. Caractéristiques d'un système de filtrage :

- Exploite un grand volume de données, entrant transmis par des sources distantes**[30],**
- Basé principalement sur le profil de l'utilisateur ou d'un groupe d'utilisateurs**[30],**,
- Diffuse que les informations en adéquation avec le profil l'utilisateur **[3].**
- Accès aux derniers documents arrivés or la rapidité avec laquelle ils arrivent **[4].**
- Intègre l'utilisateur, avec l'évaluation des ressources recommandées, pour la mise à jour de son profil **[5].**

4. Processus de filtrage d'information : [30]

Le principe de fonctionnement d'un système de filtrage d'information est d'acheminer des documents vers des groupes de personnes, depuis ces objectifs ou ces désirs définis préalablement qui sont relativement stables, à long terme ou périodiques. Ceci amène à des besoins réguliers d'information (exemple : être à jour sur un sujet) qui peuvent évoluer

lentement au cours du temps au fur et à mesure que les conditions, objectifs et connaissances changent.

La *figure I.2* illustre l'architecture de base des systèmes de filtrage d'information, telle qu'elle a été présentée par Belkin et Croft **[1].**

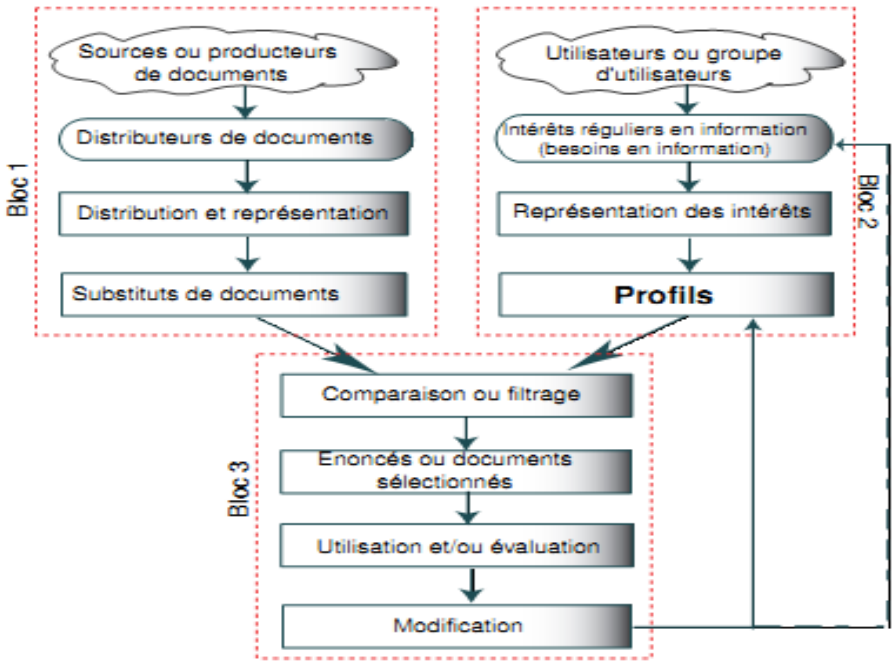

Figure I.2 : Architecture générale d'un SFI **[1].**

La figure est caractérisée par trois blocs :

- ***Bloc 1 :*** La création de substituts (représentation) de documents.
  - A l'arrivé d'un document, le système de filtrage associe une représentation de contenu à ce document.
- ***Bloc 2 :*** Création de profils :
  - Il représente les besoins en information d'un nouveau l'utilisateur par des profils
  - Il modifie les profils d'un ancien utilisateur.
- ***Bloc 3 :*** Le processus de comparaison et de filtrage :

- Il compare la représentation des documents et les profils de l'utilisateur actif et décide si les documents sont pertinents ou non pour les envoyer à l'utilisateur. Cette étape peut mener dans la plupart des cas à l'amélioration des profils et des domaines d'intérêt de l'utilisateur.

Ce processus de filtrage est déclenché l'arrivée d'un nouvel événement au Bloc 1 (respectivement Bloc2).

5. Grandes familles de filtrage d'information :

Malone dans **[6],** a identifie trois grandes familles de filtrage : le filtrage *cognitif, social* et *économique.* Ces types de filtrage se différencient principalement sur les indices qu'ils utilisent pour décider de sélectionner ou de rejeter un document.

Dans le filtrage cognitif, souvent appelé **filtrage base sur le contenu**, le filtrage tient compte seulement des contenus du document et du profil.

Dans le filtrage social, également appelé **filtrage collaboratif,** la décision de filtrage se base sur les annotations et commentaires attribuées par les utilisateurs aux documents.

Enfin, le filtrage économique se base sur des incitations additionnelles (**filtrage Hybride**), tels que des crédits attaches au document par son créateur. Les concepts du filtrage économique font souvent partie intégrante des filtrages bases sur le contenu et le filtrage collaboratif.

Ce sont ces deux filtrages qui présentent des caractéristiques spécifiques. Nous décrivons brièvement ces deux techniques dans les sections suivantes.

5.1 Filtrage d'information basé sur le contenu (cognitif) :

Les systèmes de recommandation bases sur le contenu s'appuient sur des évaluations effectuées par un utilisateur sur un ensemble de documents ou items. L'objectif est alors de comprendre les motivations l'ayant conduit à juger comme pertinent ou non un item donne.

Les techniques de filtrage basées sur le contenu fonctionnent par la caractérisation du contenu de l'information (document) à filtrer **[6].** Les représentations des documents et des

profils dans ce type de filtrage exploitent seulement les informations qui peuvent être d´étrivées de leur thème respectif **[7].** Autrement dit, la sélection de documents se base sur une comparaison des thèmes abordes dans les documents par rapport aux thèmes intéressant l'utilisateur. **La figure I.3** présente un processus de filtrage d'information base sur le contenu.

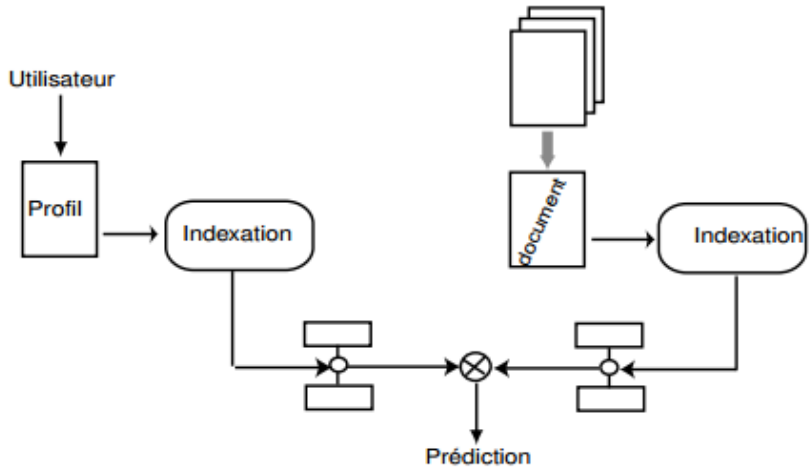

Figure I.3: Processus de filtrage base sur le contenu (filtrage cognitif) **[7].**

5.1.1. Avantages[7] [6]:

- L'utilisateur dans un tel système ne dépend absolument pas des autres
- Il peut répondre aux intérêts à long terme des utilisateurs
- Employant des techniques efficaces dans le domaine de l'intelligence artificielle pour la mise à jour des profils.

5.1.2 Problématiques du filtrage d'information cognitif [7][6]:

On pose quatre questions qui résument la problématique de ce type de système de filtrage :

- la première est comment représenter le profil ?
- la seconde, comment construire une fonction de décision ?
- la troisième, comment améliorer la représentation du profil ?
- et enfin la quatrième, comment adapter la fonction de décision ?

Ainsi, concernant les deux premières questions, dans la majorité des techniques de filtrage, le profil est représente par une liste de mots clés, éventuellement pondères, extraits du texte du profil ou élabores a partir de données d'apprentissage, et la fonction de décision est un seuil fixe arbitrairement ou appris sur une collection d'entrainement. En fait, quand on parle de la fonction de décision on fait principalement référencé a la manière d'identifier le seuil de décision, on parle souvent de fonction de seuillage.

Une grande partie des travaux effectues dans ce domaine s'est plutôt focalisée sur l'amélioration des profils et de la fonction de décision. Ceci est réalisé grâce a un processus d'apprentissage bas´e sur les éléments déjà filtres. Cet apprentissage peut se faire de manière incrémental, dans ce cas, le processus est déclenche a chaque réception d'un document (pertinent et/non pertinent), ou bien de manière différée, c'est à dire sur des ensembles de documents déjà filtres.

5.2 Filtrage d'information collaboratif :

Le filtrage collaboratif se base sur l'hypothèse que les personnes a la recherche d'information devraient pouvoir se servir de ce que d'autres ont déjà trouve et évalue. Le principe de base du filtrage collaboratif **[8]** est d'automatiser les processus sociaux, tel que le langage parle. Dans la vie quotidienne, les personnes communiquent par des recommandations entre elles : des mots, des lettres de recommandation, des films et des livres, des journaux, etc. Les systèmes de filtrage collaboratif permettent d'automatiser ceci en facilitant la prise de décision sur les informations qu'elles reçoivent.

Le filtrage collaboratif, d'après Goldberg et al. **[9],** décrit les techniques d'un groupe d'utilisateurs pour prédire la préférence d'un nouvel utilisateur ; les recommandations pour le nouvel utilisateur sont basées sur ces prédictions. Une préférence d'un utilisateur est d´écrite par un profil, qui est défini par un vecteur de dimension fini (correspondant au nombre de documents disponibles). Chaque valeur du vecteur représente l'´evaluation que l'utilisateur a attribue au document**.**

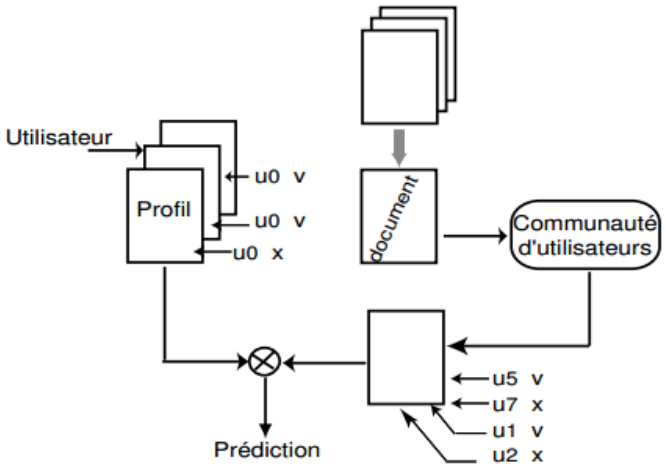

Figure I.4 : Processus de filtrage social (collaboratif) **[9].**

La figure I.4 présente un cas de processus de filtrage d'information collaboratif, ou la prédiction de l'opinion (ici, représente par v et x) qu'un utilisateur u0 sur un document donne, est calculée en rapprochant les évaluations passées de l'utilisateur des évaluations que d'autres utilisateurs (u1,u2,u5et u7) de la communauté ont donne par le passe sur les mêmes documents **[10]**.

5.2.1. Avantages [9] [10].:

- Filtrer tout type d'information (images, vidéos, etc.)
- L'utilisateur capable de découvrir divers domaines intéressants
- Permet d'exprimer des autres facteurs et critères tels que : la qualité de l'information, le public visé, la zone géographique, etc. Chose qui n'est pas possible dans un système de filtrage thématique
- L'effet entonnoir est atténué, car tout document évalué par une personne peut être recommandé. L'utilisateur peut bénéficier de nouveaux axes de recherche auxquels il n'a jamais pensé.

5.2.2. Limites[9] [10]:

- ***Démarrage à froid :*** Ce phénomène se produit en début d'utilisation de ce système, à l'arrivée d'un nouvel utilisateur, lorsqu'il s'inscrit pour utiliser le système, sa communauté

est encore inconnue, ce qui conduit à l'impossibilité de fournir des recommandations pertinentes.

- ***Masse critique :*** Afin de former des meilleures communautés, le système exige un nombre suffisant d'évaluations en commun (même opinions) entre les utilisateurs pour les comparer entre eux. Et pourtant, vu la taille énorme de l'ensemble des documents, achats, etc. dans les systèmes, le nombre des évaluations en commun entre utilisateurs risque d'être faible.

5.3. Filtrage hybride [30] :

Le principe d'hybridation s'effectue en deux phases :

1. Appliquer séparément le filtrage collaboratif et autres techniques de filtrage pour générer des recommandations candidates.
2. Combiner ces ensembles de recommandations préliminaires selon certaines méthodes telles que la pondération, la mixtion, la cascade, la commutation, etc.…, afin de produire les recommandations finales pour les utilisateurs.

Plus généralement, les systèmes hybrides gèrent des profils d'utilisateurs orientés contenus, et la comparaison entre ces profils donne lieu à la formation de communautés d'utilisateurs permettant le filtrage collaboratif.

6. Evaluation des performances des systèmes de filtrage d'information :

Plusieurs mesures standards utilisées dans l'évaluation des systèmes de recherche d'information (par exemple, les mesures de précision et rappel) ne sont pas applicables dans le cas de filtrage. Par exemple, dans le cas d'un système qui filtre, quotidiennement, des medias (images, vidéos) sur Internet, il est quasiment impossible de calculer le rappel, car l'utilisateur ne dispose pas des informations pertinentes non sélectionnées. Il y a plusieurs métriques de classification, et prédictives, et on base sur les prédictives c'est l'essentielle pour nous.

- ✓ **Les métriques prédictives [30] :**

Pour évaluer les algorithmes de filtrage collaboratif, on utilise généralement une technique statistique appelée « validation croisée » (*cross-validation*).elle consiste à séparer les données

disponibles en sous-ensembles. La première partie sert à faire la prédiction et la deuxième, à valider (Montrer la généralité du système de filtrage proposée c.à.d. attendre tous les objectives déclarer) l'algorithme. En pratique, cela se passe comme suit :

1. Nous choisissons aléatoirement un certain nombre d'individus, par exemple, la moitié de ceux-ci. Ces individus sont représentés par l'algorithme P dans les calculs (validation).

2. Les autres individus constituent un ensemble « test ».

Les individus de l'ensemble « test » sont alors pris un à un. Pour chaque individu x, on note leur évaluation réelle par « r_i » à l'article « i », et on fournit par l'algorithme P une évaluation prédite « p_i ».

Ensuite on mesure le pourcentage (%) d'erreur faite en prédisant la note accordée par les individus de « test » aux articles « i » par la fonction « MAE » :

$$MAE = \frac{\sum_{i=1}^{N} |p_i - r_i|}{N}$$

7. Quelques Systèmes de filtrage :

Quelques années plus tard, avec l'essor de l'Internet et des applications Web, il y a eu un engouement pour les systèmes de recommandation et surtaux celles collaboratives qui se sont développés dans différents domaines d'application. Nous pouvons en citer :

- **Tapestry :**

Le concept du filtrage collaboratif a été lancé avec le projet Tapestry à Xerox Parc. La gestion des e-mails est sa motivation première [Goldberg, 1992]. Tapestry repose sur une «recommandation commentée» basé sur des annotations de qualité ou d'appréciation des documents faites par les utilisateurs. De cette manière, les documents sont filtrés en fonction de ces annotations **[31].**

- **Le système de Malltz et Ehrlich :**

Le système de Maltz et Ehrlich est présenté comme un substitut au mail dans ces situations. Il est intégré à un système de recherche d'informations et permet à ses utilisateurs d'adresser des pointeurs aux personnes qu'ils jugent intéressées, sans avoir à interrompre leur session de recherche d'informations. D'un autre côté, l'ensemble de ces échanges est stocké pour constituer une base de références **[32].**

Conclusion :

Dans ce chapitre, nous avons présenté la définition, le rôle et les différentes méthodes du Processus de filtrage de l'information.

Ce chapitre a porte essentiellement sur l'´etude des systèmes de filtrage d'information, plus particulièrement sur le filtrage collaboratif, dit aussi basé sur le contenu. La majorité des approches proposées dans la littérature se basent principalement sur des modèles de recherche d'information augmentes par une fonction de décision. Les travaux dans le domaine de filtrage s'intéressent principalement à l'apprentissage du profil et l'adaptation de la fonction de décision.

L'apprentissage du profil se fait sur des bases de données ou on sauvegarde les données spécifiques des utilisateurs. On le chapitre suivant on va mieux détailler la notion des bases de données et ses notions de base.

Chapitre 2 : Base de données

Introduction :

L'informatique évolue vers le traitement de masses d'informations de plus en plus grandes dans des environnements répartis géographiquement où doivent cohabiter des matériels hétérogènes. Dans ce contexte, les bases de données sont utilisées de façon intensive pour de nombreux domaines d'application tels que le domaine médical, les administrations ou les associations. Les applications concernées par l'utilisation d'un SGBD (Système de gestion de base de données) possèdent des caractéristiques différentes tant au niveau du volume de données concernées qu'au niveau de la complexité de ces données et des traitements informatiques à réaliser. Néanmoins, le regroupement des données dans une base de données gérée par un système de gestion de base de données apporte de nombreux avantages dans la plupart des cas d'utilisation.

Le domaine informatique bien qu'étant jeune, a une évolution croisière. Jadis, la gestion et le traitement des données se faisaient par la méthode classique à laquelle l'on a pu dégager ces défauts suivants:

- La redondance de données ;
- La dépendance pleine entre données et traitement ;
- Le manque de normalisation au niveau de stockage de données.

Pour remédier à cette situation, il a été mis au point la notion de base de données répondant aux questions suivantes:

- L'accès aux données selon les multiples critères ;
- L'intégration des données ;
- La relation entre les données.

I. Les bases de données :

1. Définition [11]: Une base de données est une entité dans laquelle il est possible de stocker des données de façon structurée et avec le moins de redondance possible.

Ces données doivent pouvoir être utilisées par des programmes, par des utilisateurs différents. Ainsi, la notion de base de données est généralement couplée à celle de réseau, afin de pouvoir mettre en commun ces informations, d'où le nom de **base**. On parle généralement de système d'information pour désigner toute la structure regroupant les moyens mis en place pour pouvoir partager des données.

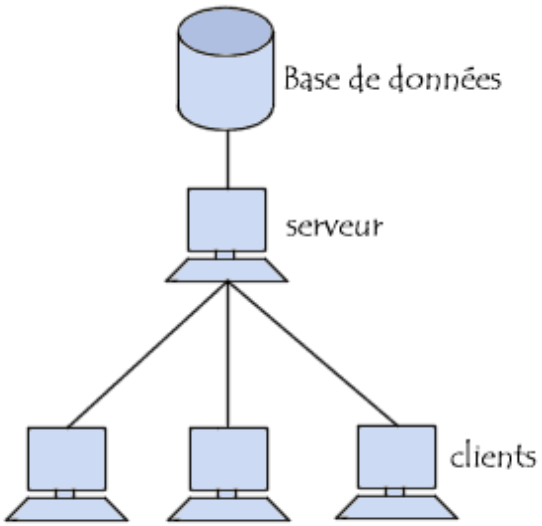

Figure II.1 : Une base de données reliée avec un serveur**[11].**

1.1. Critères d'une base de données [11]:

Une base de données doit répondre aux trois critères suivants :

- L'exhaustivité : C'est la présence dans cette base de tous les enseignements qui ont trait aux applications en question.
- Le non redondance des données : Non répétition d'une donnée plusieurs fois.
- La structure : C'est l'adaptation du mode de stockage de données au traitement structuration que la base doit avoir est liée à l'évolution de la technologie

1.2 Objectifs d'une base de données [11] :

La base de données a beaucoup d'adjectifs parmi lesquels nous pouvons citer :

- Eviter les redondances et les incohérences des données qui entraînaient fatalement une approche où les données seraient reparties dans des différents fichiers sans connexion entre eux.

- Offrir un langage de haut niveau pour la définition et la manipulation des données ;

- Contrôler l'intégrité entre plusieurs utilisateurs et la confidentialité des données ;

- Assurer l'indépendance entre les données et les traitements

2. Système de Gestion de Base de données (SGBD) [13] :

Un SGBD est un ensemble des programmes et des langages de commande qui permettent de :

- Définir des "bases de données", et des relations entre les éléments de chaque base ;
- Spécifier le traitement de ces données : interrogations, mises à jour, calculs, extractions...

Le SGBD reçoit des commandes aussi bien des programmes d'application que des utilisateurs : il commande les manipulations de données, généralement par l'intermédiaire d'un SGF. **[13]**

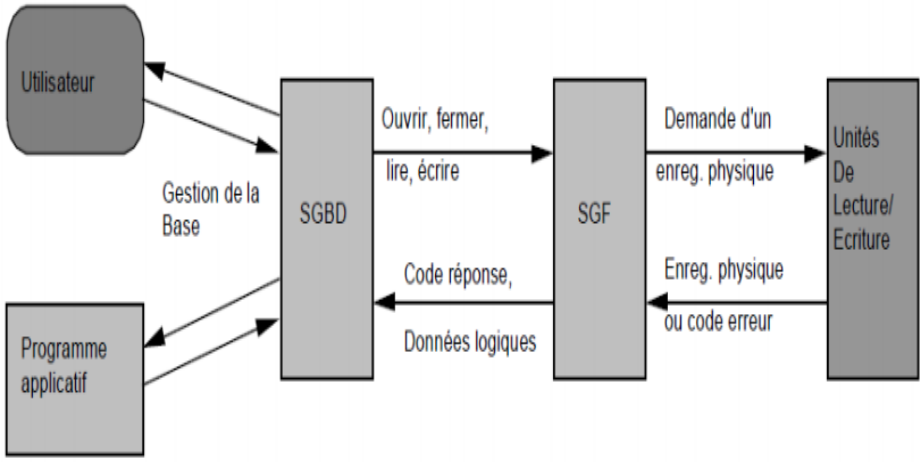

Figure II.2: Système de Gestion de Base de données « SGBD » **[13]**

2.1 Structure fonctionnelle d'un SGBD [13] :

Un SGBD est généralement composé de trois couches comme présente *la figure II.3* :

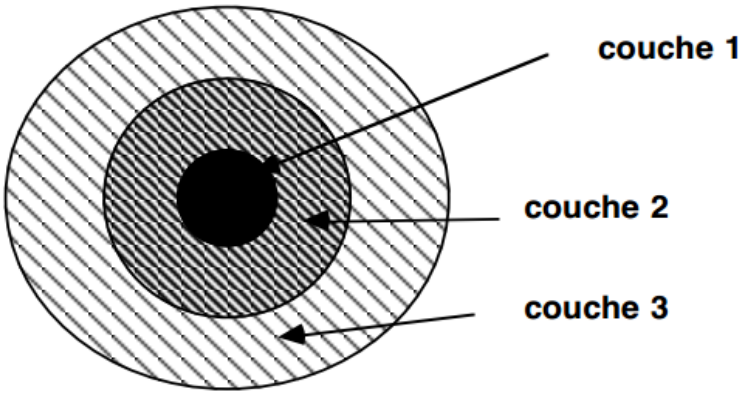

Figure II.3 : Structure fonctionnelle d'un SGBD **[13]**.

- **Couche 1 :** Gestion des récipients de données sur mémoire secondaire : système de gestion de fichiers (fonctions de base)
- **Couche 2 : SGBD interne :**
  - gestion des données stockées dans les fichiers
  - placement, assemblage de ces données
  - gestion des liens entre données et structures de recherche rapide (index)
- **Couche 3 : SGBD externe :** Présentation des données aux programmes d'applications et aux usagers ayant formulé leurs besoins en langage ± élaborés (requêtes, rapports, L4G...)

2.2 Objectifs de l'approche SGBD [13]:

- Orientés données :
  - Non redondance des données
  - Partagabilité des données
  - Sécurité des données
  - Cohérence des données

- Orientés traitements **:**
  - indépendance physique des données
  - indépendance logique des données
  - manipulation facile des données :
  - par informaticien

- par non informaticien
- cohérence physique (pannes, ...)

➢ Organisationnels :

- administration centralisée des données

2.3 Architecture de SGBD [12]:

2.3.1 Architecture Client-serveur : Depuis les années 80, les SGBD sont basés sur une architecture client-serveur.

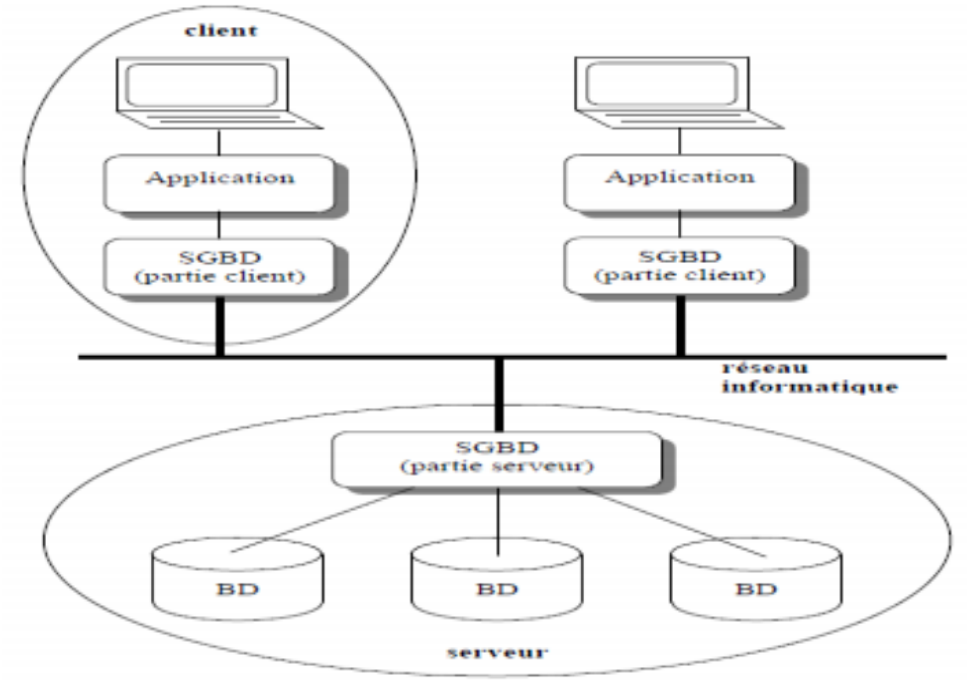

Figure II.4: Architecture Client-serveur. **[12]**

2.3.2 Architecture centralisée :

Ce type d'architecture est appelée solution sur site central (Mainframe). Historiquement, les applications sur site central ont été les premières à proposer un accès multiutilisateurs. Dans ce contexte, les utilisateurs se connectent aux applications exécutées par le serveur central à l'aide des terminaux se comportant en esclaves. C'est le serveur central qui prend en charge l'intégralité des traitements y compris l'affichage qui est simplement déporté sur des terminaux**. [11]**

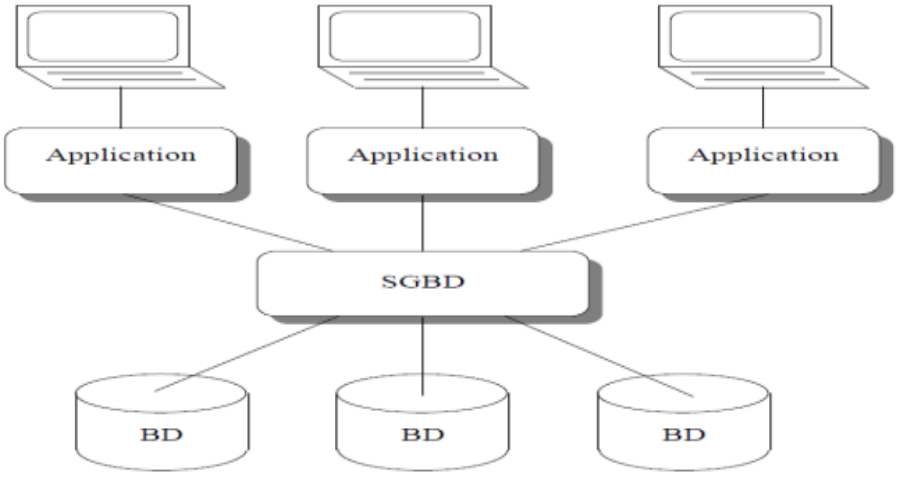

Figure II.5: Architecture centralisée. **[12]**

2.4 Les opérations sur les données : [14]

Il existe quatre opérations classiques (ou requêtes) :

- La création (ou insertion).
- La modification (ou mise-à-jour).
- La destruction.
- La recherche.

Ces opérations correspondent à des commandes du LMD (Langage de Manipulation des Données). La plus complexe est la recherche en raison de la variété des critères.

Pour l'utilisateur, une bonne requête a les caractéristiques suivantes. Tout d'abord elle s'exprime facilement : l'idéal serait de pouvoir utiliser le langage naturel, mais celui-ci présente trop d'ambiguïtés. Ensuite le langage ne devrait pas demander d'expertise technique (syntaxe compliquée, structures de données, implantation particulière ...).

Il est également souhaitable de ne pas attendre trop longtemps (à charge pour le SGBD de fournir des performances acceptables). Enfin, et peut-être surtout, la réponse doit être fiable. Une bonne partie du travail sur les SGBD consiste à satisfaire ces besoins. Le résultat est ce que l'on appelle un langage de requêtes, et constitue à la fois un sujet majeur d'étude et une caractéristique essentielle de chaque SGBD. Le Langage le plus répandu à l'heure actuelle est SQL.

3. Types de base de données :

On retrouve fréquemment les bases de données suivantes :

- Référence (qui regroupe des informations bibliographiques ou factuelles).

- Texte intégral (qui regroupe des documents à caractère textuel).
- Multimédia (qui regroupe documents sonores, visuels, etc.).

Une base de données peut se trouver sur n'importe quel support : disque dur, cédérom, sur un serveur et accessible en réseau interne ou en ligne (à distance).

4. SGBD réparti ou SGBD distribué [18]:

Système gérant une collection de BD logiquement reliées, réparties sur différents sites en fournissant un moyen d'accès rendant la distribution transparente.

Deux approches fondamentales sont à l'origine de la conception des bases de données réparties : la conception descendante 'Top down design' et la conception ascendante 'Bottom up design'.

4.1. Conception descendante[18] :

On commence par définir un schéma conceptuel global de la base de données répartie, puis on le distribue sur les différents sites en des schémas conceptuels locaux.

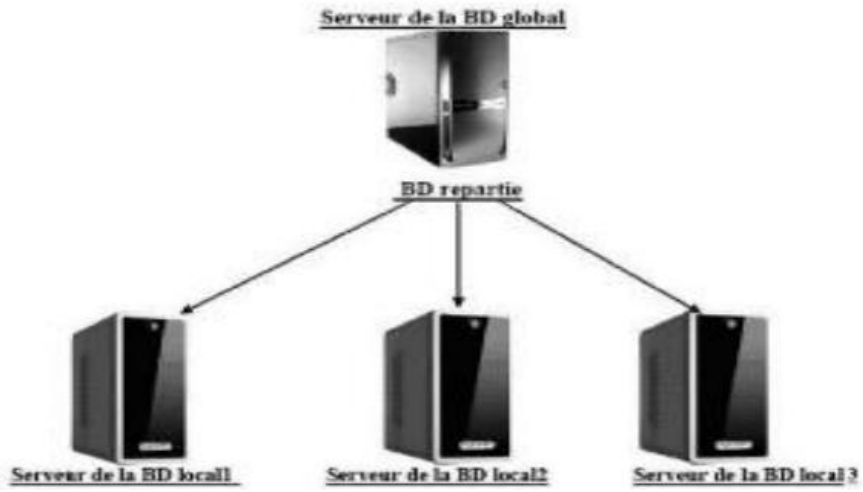

Figure II.6 : Architecture de la conception descendante. **[18]**

La répartition se fait donc en deux étapes, en première étape la fragmentation et en deuxième étape l'allocation de ces fragments aux sites. L'approche top down est intéressante quand on part du néant. Si les BDs existent déjà, la méthode bottom up est utilisée. **[18]**

4.2. Conception ascendante (Base de données fédérée):

Base de données fédérée (BDF) est une BD répartie hétérogène, c'est-à-dire constituée à partir de sources de données de nature variées : fichiers classiques, fichiers de textes, documents HTML, XML, BD relationnelle ou objet, etc.

L'objectif est de fournir aux utilisateurs une vue intégrée de différentes données de l'entreprise soit dynamiquement sur demande, soit en la matérialisant périodiquement dans un entrepôt de données. (Plusieurs BD hétérogènes capables d'inter opérer via une vue commune (modèle commun)).

Cette approche se base sur le fait que la répartition est déjà faite, mais il faut réussir à intégrer les différentes BDs existantes en une seule BD globale. En d'autre terme, les schémas conceptuels locaux existent et il faut réussir à les unifier dans un schéma conceptuel global. **[18]**

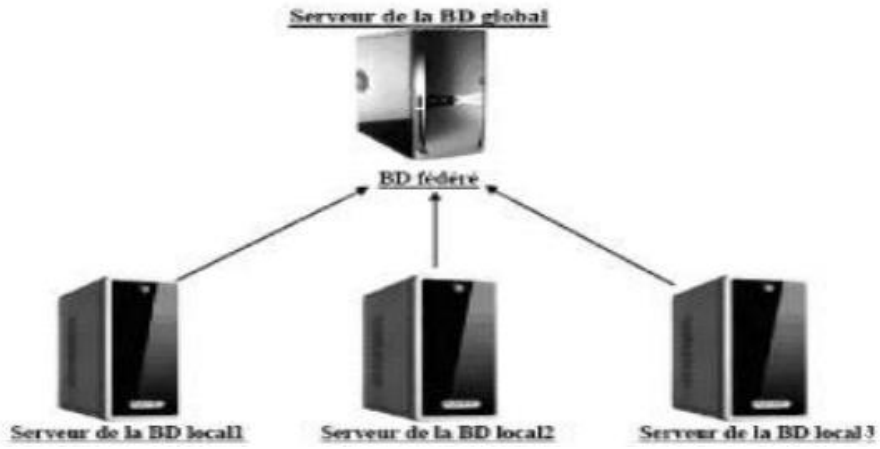

Figure II.7 : Architecture de la conception ascendante**[18]**.

5. Sécurité et confidentialité d'une base de données [11]:

La base de données doit être sécurisée contre :

- Les indiscrétions : Par un mot de passe.
- Les erreurs : Des contrôles doivent être mis en place pour vérifier que des contraintes d'intégrités sont respectées.

- Les destructions : En cas d'incident (panne logicielle, panne matérielle Ou panne d'électricité), des procédures de sauvegarde et reprise doivent être prévues afin de relancer le système sans avoir recommencé les saisies par la transaction**.**

6. Qui intervient sur une BDD [12]?

- **L'administrateur** (une personne ou une équipe) :

Il définit le schéma conceptuel de la BDD et le fait évolué, comme il fixe les paramètres de l'organisation physique de façon à optimiser les performances, et aussi il gère les droits d'accès et les mécanismes de sécurité. **[12]**

- **Les programmeurs d'applications :**

Ils définissent les schémas externes et construisent les programmes qui alimentent ou exploitent la BDD en vue d'applications particulières. Ils utilisent pour cela le langage de bases de données du SGBD, éventuellement couplé avec un langage de programmation classique. **[12]**

- **Les utilisateurs finals :**

Ils accèdent à la BDD au travers des outils construits par les programmeurs d'application ou pour les plus avertis au travers du langage de requêtes. **[12]**

7. Principaux modèles logiques de SGBD [13] :

Il existe cinq modèle de SGBD, les différenciés selon la représentation des données qu'elle contient :

7.1. Modèle hiérarchique :

Le modèle hiérarchique est une forme de système de gestion de base de données qui lie des enregistrements dans une structure arborescente de façon à ce que chaque enregistrement n'ait qu'un seul processeur. Il s'agit du premier modèle de SGBD.

- **Avantage :**
  - Rigueur des structures et des chemins d'accès.
  - Simplicité relative de l'implémentation.
  - Adéquation parfaite du modèle à une entreprise à structure arborescente.

- **Inconvénients :**
  - Les accès se font uniquement depuis la racine.
  - Indépendance logique très réduite : la structure du schéma doit refléter les besoins des applications.
  - Pas d'interface utilisateur simple.

7.2 .Modèle réseau :

Ce modèle utilise des pointeurs vers des enregistrements. Une base en réseau peut être décrite comme un certain nombre de fichiers comportant des références les uns Vers les autres. Les entités sont connectées entre elles à l'aide de pointeurs logiques.

- **Avantages et inconvénients du modèle**
  - Aucune restriction dans la conception : un type de "record" peut à la fois être propriétaire et membre de plusieurs sets.
  - Représentation naturelle des liens maillés N:M.
  - Pas d'anomalies pour les opérations de stockage
  - Procédural ite importante des langages de manipulation ; l'utilisateur doit "naviguer" dans le réseau logique constitué par les enregistrements et les chaînes de pointeurs.

7.3. Modèle relationnelle :

Le modèle relationnel a été défini par E.F. Codd en 1970 à IBM San José. Un modèle est dit relationnel dans la mesure où il permet de parcourir la structure des données empruntant des chemins non prédéfinis, constitués en dynamique par des requêtes Les concepts du modèle relationnel découlent de la théorie des ensembles. A ce type de modèle sont associées les notions suivantes:

- Domaine
- Table relationnelle
- Attribut
- tuple (ou n-uplet)

Caractéristiques du modèle :

- Schéma de données facile à utiliser : toutes les valeurs sont des champs de tables à deux dimensions.

- Améliore l'indépendance entre les niveaux logique et physique : pas de pointeurs visibles par l'utilisateur.
- Optimise les accès aux bases de données.
- Améliore l'intégrité et la confidentialité : unicité de clé, contrainte d'intégrité référentielle.
- Prend en compte une variété d'applications, en gestion et en industriel.
- Fournir une approche méthodologique dans la construction des schémas

7.4. Modèle déductif :

Un SGBD Déductif est un SGBD qui peut faire des déductions sur la base des Règles et des faits enregistrés dans la base de données .Le but est d'utiliser des méthodes semblables à celles pratiquées pour la déduction en intelligence artificielle. Une base de données déductive (BDD) est constituée de: **BDE** et **BDI**.

- **Base de données extensionnelle (BDE):** Ensemble des faits connus (tuples) Dans la BDD relationnelle.
- **Base de données intentionnelle (BDI**): Ensemble des faits ou règles déduits.

7.5. Modèle objet :

Modèle objet est l'annuaire, qui est capables de stocker une multitude d'informations. Il stocke l'information dans des objets, très souvent une fiche individuelle, une machine, une ressource, à laquelle on associe des valeurs, ses attributs. **[15]**

7.6. Modèle multidimensionnel :

Il permet de stocker différentes données numériques aux croisements des "n" axes correspondant aux "n" dimensions de la base .Il est alors possible de naviguer dans cet espace, à différents niveaux d'agrégats (zooms, rotation d'axes, etc.) : ces bases de données sont appelées cubes ou hyper cubes en informatique décisionnelle et sont souvent utilisés dans les métiers du contrôle de gestion. **[15]**

Les bases multidimensionnelles sont le plus souvent formées par agrégats de bases pouvant être relationnelles, en tout cas hétérogènes. **[16]**

7.7. Modèle semi-structuré :

La popularité du web et l'essor de XML ont contribué à l'émergence des bases de données semi-structurées et des bases de documents. Les modèles classiques de bases de données relationnelles ou objets supportent difficilement les données semi-structurées qui sont complexes, hétérogènes, distribuées, parfois incomplètes. La force des modèles semi-structures est de ne plus imposer de structure a priori dans le schéma mais de la définir a posteriori dans les données elles-mêmes. Plus récemment, les chercheurs s'intéressent à l'extraction de connaissances à partir de données semi-structurées du web. L'objectif de bases de données semi-structurées est de faire le point sur les avancées actuelles dans le domaine des bases de données semi -structurées d'un point de vue théorique et de recenser les applications originales qui s'appuient sur ce formalisme. **[17]**

Conclusion :

A la fin de ce chapitre on peut retenir que une base de données est une entité dans laquelle il est possible de stocker des données de façon structurée et avec le moins de redondance possible. Ces données doivent pouvoir être utilisées par des programmes, par des utilisateurs différents.

Une base de données permet de mettre des données à la disposition d'utilisateurs pour une consultation, une saisie ou bien une mise à jour, tout en s'assurant des droits accordés à ces derniers. Pour cela il faut assurer la sécurité des bases données qui est l'objectif du chapitre suivant.

Chapitre 3 : Sécurité dans les Bases De Données.

Introduction :

Le terme sécurité dépend du contexte ou il est utilisé, par exemple, la sécurité d'un véhicule. D'un point se vue, la sécurité d'une voiture est principalement concentrée sur la sûreté des personnes à l'intérieur de la voiture, d'un autre point de vue, la sécurité d'une voiture peut être contre les voles.

La sécurité est souvent interprétée de manière subjective, elle correspond à une protection qui n'est pas forcement la même pour tous, en effet, elle change en fonction de ses besoins.

Cependant dans le contexte de la sécurité des bases de données, il convient de matérialiser objectivement la notion de la sécurité. La sécurité des BDDs n'est qu'une partie de la sécurité des informations. Les informations existent à l'intérieur d'une BDD doivent être protégées chaque fois qu'elles soient utilisées, transmises [19]. La sécurité des bases de données traite en premier lieu et essentiellement les informations et leurs traitements.

I. Sécurité informatique :

1. Définition :

La sécurité informatique est la capacité d'un système de protéger ses objets contre leur modification ou leur utilisation par des sujets non autorisés **[20].**

2. Les objectifs de la sécurité informatique [20]:

La sécurité informatique à plusieurs objectifs, bien sûr liés aux types de menaces ainsi qu'aux types de ressources, etc... Néanmoins, les points principaux points sont les suivants :

- empêcher la divulgation non-autorisée de données
- empêcher la modification non-autorisée de données
- empêcher l'utilisation non-autorisée de ressources réseau ou informatiques de façon générale.

3. Terminologie de la sécurité informatique :

La sécurité informatique utilise un vocabulaire bien défini que nous utilisons dans nos articles. De manière à bien comprendre ces articles, il est nécessaire de définir certains termes **[20]** :

- ***Les vulnérabilités*** *:* ce sont les failles de sécurité dans un ou plusieurs systèmes. Tout système vu dans sa globalité présente des vulnérabilités, qui peuvent être exploitables ou non.
- ***Les attaques (exploits):*** elles représentent les moyens d'exploiter une vulnérabilité. Il peut y avoir plusieurs attaques pour une même vulnérabilité mais toutes les vulnérabilités ne sont pas exploitables.
- ***Les contre-mesures :*** ce sont les procédures ou techniques permettant de résoudre une vulnérabilité ou de contrer une attaque spécifique (auquel cas il peut exister d'autres attaques sur la même vulnérabilité).
- ***Les menaces :*** ce sont des adversaires déterminés capables de monter une attaque exploitant une vulnérabilité.

4. Services principaux de la sécurité informatique :

Pour remédier aux failles et pour contrer les attaques, la sécurité informatique se base sur un certain nombre de services qui permettent de mettre en place une réponse appropriée à

chaque menace. A ce niveau, aucune technique n'est encore envisagée; il ne s'agit que d'un niveau d'abstraction visant à obtenir une granularité minimale pour déployer une politique de sécurité de façon optimale (les aspects pratiques tels qu'analyses de risques, solutions technologiques et coûts viendront par la suite **[21]**. Décrivons les principaux services de sécurité :

a. Confidentialité : La confidentialité est ainsi définie par (ISO) **[21]** comme : "*le fait de s'assurer que l'information est seulement accessible qu'aux entités dont l'accès est autorisé*". Cette définition implique que l'information ne doit pas être accessible par certaines entités, mais doit être accessible par d'autres.

b. Intégrité : Les données doivent être celles que l'on attend, et ne doivent pas être altérées de façon fortuite, illicite ou malveillante. En clair, les éléments considérés doivent être exacts et complets.

c. Disponibilité : Le système doit fonctionner sans faille durant les plages d'utilisation prévues et garantir l'accès aux services et ressources installées avec le temps de réponse attendu.

D'autres aspects peuvent aussi être considérés comme des objectifs de la sécurité des systèmes l'information, tels que :

- *La traçabilité* (ou « **Preuve** ») : garantie que les accès et tentatives d'accès aux éléments considérés sont tracés et que ces traces sont conservées et exploitables.
- *L'authentification* : L'identification des utilisateurs est fondamentale pour gérer les accès aux espaces de travail pertinents et maintenir la confiance dans les relations d'échange.

- *La non-répudiation* et *l'imputation* : Aucun utilisateur ne doit pouvoir contester les opérations qu'il a réalisées dans le cadre de ses actions autorisées, et aucun tiers ne doit pouvoir s'attribuer les actions d'un autre utilisateur.

Notez que le chiffrement, les signatures digitales et autres techniques correspondent au niveau d'abstraction inférieur, décrit comme l'ensemble des mécanismes de sécurité permettant de réaliser les services décrits ci-dessus.

II. Sécurité des Bases de données[22] :

Bien que la sécurité soit l'une des raisons de l'architecture trois couches, plusieurs challenges praticables sont enlevés lors de la construction du système tel que l'authentification des utilisateurs, le contrôle des accès et Audit les actions des utilisateurs, la protection des données entre les couches, la limitation des privilèges de l'intermédiaire, et la construction des systèmes extensibles.

1. Processeur de sécurité :

1.1. Autorisation, interdiction et obligation :

- Les règlements les plus simples ne contiennent que des **autorisations**: ce qui n'est pas autorisé est interdit.
- Certains règlements incluent des **interdictions** à fin de spécifier des exceptions à des permissions générales. Exemple: les patients ont droit de consulter leur dossier médical sauf Jean Dupont.
- D'autres enfin, plus sophistiqués, incluent des **obligations**: difficiles à implanter dans les systèmes informatiques. **[22]**

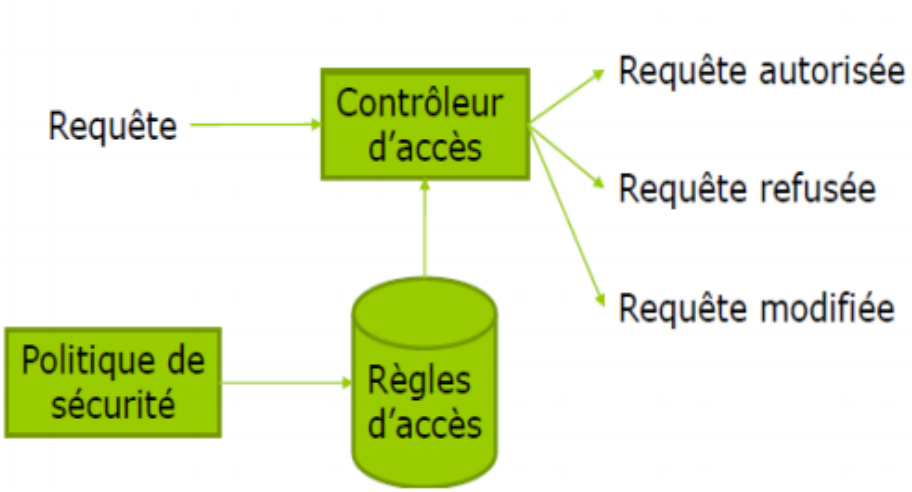

Figure III.1 : Processus de sécurité. **[22]**

2. Attaque [22]:

Les violations de la sécurité d'une BD consistent en des lectures ou des mises à jour illicites. Les événements qui portent ces violations sont appelés des attaques.

Les attaques à une BD peuvent exploiter les failles des applications opérant sur cette BD

- Stockage des mots de passe dans les fichiers de configuration de l'application,
- Scripts de connexion à la BD accessibles dans le code source de l'application,
- Attaques par injection SQL,
- Attaques exploitant les débordements de tampons. **[22]**

2.1. Types d'attaques [22] : On distingue:

a. Les attaques **non frauduleuses:**
   - Catastrophes naturelles,
   - Pannes de logiciel ou de matériel,
   - Erreurs humaines…

b. Les attaques **frauduleuses**:
   - Utilisation abusive de leurs droits par les utilisateurs,
   - Agents hostiles exécutant des actions de destruction du logiciel ou du matériel, ou lisant ou mettant à jour des données protégées,
   - Ces agents peuvent être cachés dans des actions légales :
   - chevaux de Troie. **[22]**

2.2. Types d'attaquants[23] :

Dans cette partie on présente les différents attaquants :

- ***Pirate externe*** : il est capable de s'infiltrer sur le serveur BD et de lire ses fichiers, il peut aussi casser une clé de chiffrement avec un texte connu.
- ***Pirate utilisateur*** : ce type de pirate est reconnu par le SGBD et à accès à une partie des données suivant le mode de chiffrement, il a accès à certaines clés.

- ***Pirate administrateur (DBA)*** : employé peu scrupuleux ou pirate s'étant octroyé ces droits ; A accès à des données inaccessibles aux autres pirates (journal) et aussi peut espionner le SGBD pendant l'exécution**. [23]**

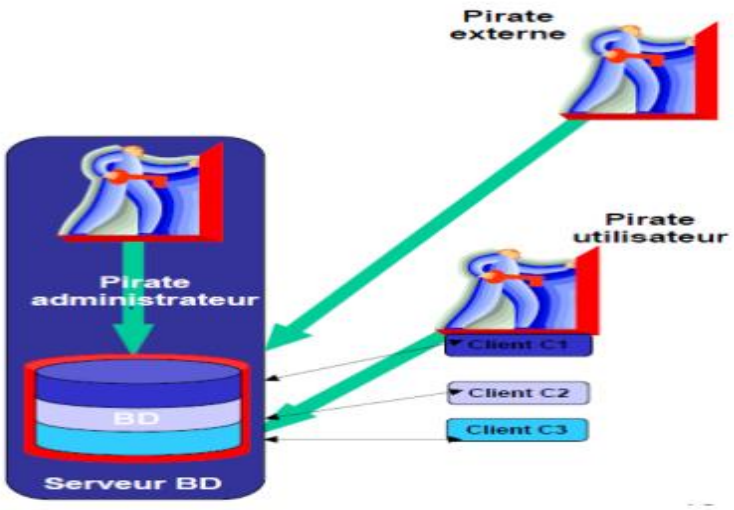

Figure III.2 : Différents types d'attaquants. **[23]**

2.3. Les risques encourus :

Les risques propres à une source de données sont les suivants :

a. **Le vol de données** induit la perte de confidentialité des données stockées. La divulgation de données financières hautement confidentielles peut avoir un impact néfaste sur l'activité d'une entreprise : risque juridique, atteinte à l'image de marque, perte de confiance des partenaires industriels.

b. **L'altération de données** induit une perte d'intégrité, c'est-à-dire que les données ne sont plus dignes de confiance. En fonction de la rapidité de détection et de la qualité des sauvegardes, les conséquences peuvent en être réduites. Mais une application fonctionnant sur des données falsifiées peut voir son comportement fortement influencé : par exemple, un site de commerce électronique pourrait débiter le compte d'un autre client que celui réalisant la commande.

c. **La destruction de données** remet sérieusement en cause la continuité de l'activité de l'entreprise concernée. Privée de ses données clients, sans sauvegarde, c'est le dépôt de bilan garanti.

d. **L'augmentation du niveau de privilèges** d'un utilisateur d'une application est plus insidieuse que les risques précédents, car comme pour l'altération de données, il n'est remarqué qu'après un certain laps de temps durant lequel le pirate peut réaliser un grand nombre d'actions malveillantes. Il peut ainsi s'attribuer le droit d'accès à des informations confidentielles, le droit d'accès à des opérations sensibles, voire même prendre le contrôle d'une application.

e. Selon le SGBD utilisé, **des ressources systèmes** peuvent être attribuées à chaque utilisateur (nombre de requêtes par unité de temps...). Ces ressources peuvent être limitées par l'administrateur système afin d'éviter l'écroulement des capacités de traitement du serveur (déni de service) par un utilisateur malveillant. De plus, ceci permet de limiter la portée d'une attaque par altération ou vol de données en limitant le nombre d'opérations réalisables en un temps donné. La conséquence d'un tel risque peut être la paralysie du serveur (perte de disponibilité).

3. Les types d'utilisateurs : [24]

Les utilisateurs ayant besoin d'un accès à la base de données peuvent être de différents types :

3.1. L'administrateur: est une personne physique ayant tous les droits sur le SGBD, mais pas forcément sur le contenu des bases de données : il peut réaliser des opérations de gestion des droits d'accès et des ressources systèmes mais on pourra choisir d'exclure ou non les droits d'accès en lecture et/ou écriture au contenu des bases de données.

Bien que parfaitement logique d'un point de vu métier, pour la protection de données sensibles par exemple, retirer à un administrateur les droits de lecture et d'écriture sur le contenu d'une base de données n'a pas de sens d'un point de vu technique puisqu'il possède les capacités techniques de s'octroyer ses droits là. De plus, les opérations de 40 sauvegardes, de

restauration et de maintenance après incident peuvent l'amener à devoir accéder au contenu d'une base de données.

Bref, normalement, c'est l'utilisateur qui a tous les droits sur le SGBD et les bases de données hébergées. C'est normalement une personne de confiance, compétente et prudente.

3.2. L'utilisateur: est une personne physique se connectant directement à la base de données (commande mysql sous Linux) ou via une interface graphique (script phpMyAdmin sur un Intranet) ou utilisant une application qui va se connecter à la base de données sous l'identité de l'utilisateur (client lourd MySQLQuery Browser**). [24]**

3.3. Une application : peut être une application web, un outil de synchronisation entre sources d'informations ou tout programme accédant pour lui-même à la base de données. Ce type d'utilisateur logique n'a rien à voir avec l'utilisateur réel dénotant une personne physique ayant des besoins particuliers. Même si une application est utilisée par des personnes physiques, on pourra choisir de déléguer à l'application la gestion des droits d'accès à l'information en fonction des habilitations qu'elle décide de lui attribuer.

Ainsi, une application peut être vue comme un utilisateur de base de données auquel on attribue des droits qu'elle pourra restreindre de façon transparente pour l'utilisateur final de l'application ainsi que pour le SGBD.

4. Politique de sécurité :

Les ITSEC (Information Technology Security Evaluation Criteria) définissent la politique de sécurité comme l'ensemble des lois règles ou pratiques qui régissent la façon dont l'information sensible et les autres ressources sont gérées, protégées et distribuées à l'intérieur d'un système d'information. **[22]**

5. Les moyens de sécurité :

5.1. Protections contre les attaques :

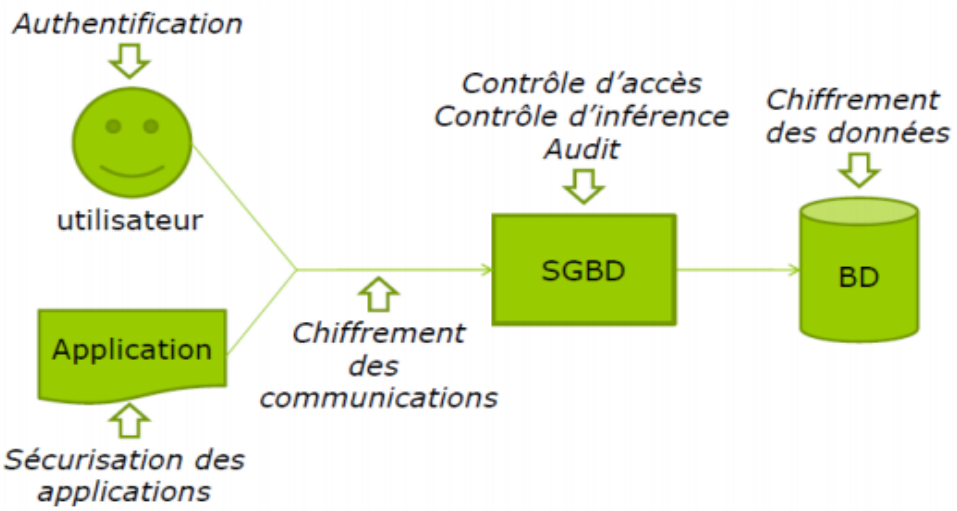

Figure III.3 : Protections contre les attaques. **[25]**

Les SGBD fournissent différents moyens pour garantir la sécurité :

5.1.1. Vues :

La base de données du fournisseur SQL Server comprend des vues prédéfinies qui permettent d'accéder aux données d'une fonctionnalité particulière sans accéder directement aux tables de base de données. L'accès aux vues fournies est en lecture seule. Vous ne devez pas essayer de mettre à jour les données de la base de données à partir des vues.

Importance des vues : elles permettent de définir de façon précise les portions d'une BD sur lesquelles des privilèges sont accordés**. [25]**

5.1.2. L'authentification des utilisateurs :

L'authentification a pour objectif d'assurer que l'utilisateur qui se connecte à la BD est:

- Autorisé à se connecter,
- Bien celui qui s'annonce.

L'authentification repose sur :

- La sécurité des mots de passe,

- Des techniques d'identification biométriques. **[22]**

5.1.3. Le contrôle d'accès des utilisateurs : Afin de permettre l'implantation de politiques de confidentialité et d'intégrité en leur sein, les systèmes d'exploitation disposent de mécanismes de contrôle d'accès. Typiquement, ceux-ci fonctionnent sur le modèle suivant :

- **Un sujet :** est une entité active inclut souvent les utilisateurs et les processus travaillant pour le compte des utilisateurs (qui peuvent être classés par groupes).
- **Un objet** est une entité passive, un conteneur d'information à protéger, sur lequel un sujet peut effectuer une action (les fichiers, Données, programmes, périphériques matériels) ;
- **Un règlement de sécurité** constitué d'un ensemble de règles d'accès traduisant la politique de sécurité du système d'information.
- **Un processeur de sécurité** qui vérifie que les requêtes adressées au système ne violent pas les règles d'accès et selon le cas autorise, modifie ou interdit la requête **[22] [27].**

Le contrôle d'accès est configuré par un ensemble de règles spécifiant un **suje**t, un **objet** et des **droits d'accès**.

Une fois que l'utilisateur est authentifié par l'intermédiaire, le système doit contrôler quelles données, applications et ressources l'utilisateur peut accéder dans le système. Les données ne doivent pas être protégées seulement contre les intrusions mais aussi les accès des utilisateurs ayant des limites qui doivent être respecté. Pour contrôler l'accès il faut d'abord renforcer la manière dont les utilisateurs font accès.

- **Modèles de contrôle d'accès :**

Définition : une politique de contrôle d'accès est un ensemble de règles.

Les modèles de contrôle d'accès permettent de définir le cadre d'expression d'une politique de sécurité.

Cette partie on présente l'état de l'art des principaux modèles de contrôle d'accès statiques : DAC, MAC et RBAC. Les parties suivantes présentent les différentes familles de modèles théoriques.

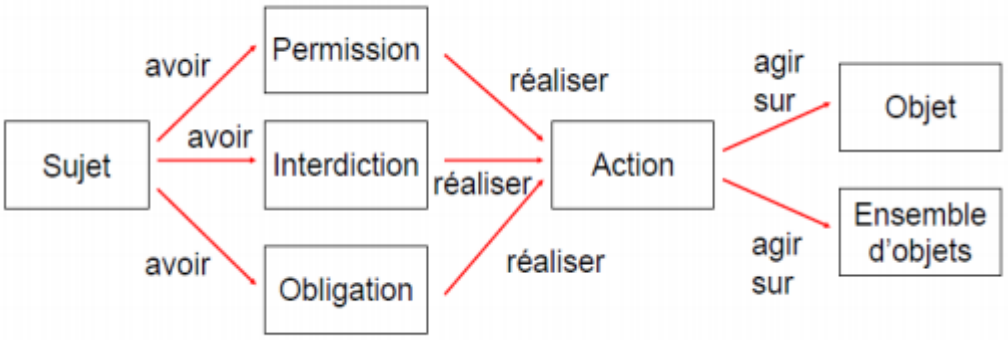

Figure III.4 : Format des règles. **[27].**

5.2. Protection des données de l'utilisateur :

Le changement des données entre les trois tiers doits être protégé contre les révélations et les modifications non attendus, le cryptage est le mécanisme standard pour ce but.

La sécurisation des données permet de crypter ou de limiter la vision des données pour un utilisateur au niveau d'une base, d'une table, d'une colonne.

La norme SQL ANSI permet déjà à travers la gestion des droits (GRANT, REVOKE) d'autoriser un utilisateur à voir ou modifier des informations sur une table, ou sur certaines colonnes. Par contre les données sont stockées en clair, non cryptées.

Dans le cas de données sensibles, il est important de proposer des solutions de cryptage des données. Pour répondre à ces problématiques, les éditeurs de SGBD proposent essentiellement deux solutions :

- La première consiste à utiliser des fonctions de cryptage directement dans le code SQL. L'emploi des fonctions de cryptage oblige à une modification du code applicatif. Les principaux algorithmes disponibles suivant les SGBD sont DES, 3DES, AES, MD5, MD4, SHA et SHA-1, et dans notre travaille on va appliqué l'algorithme MD5 qui va être bien détailler dans le chapitre suivant.
- La deuxième consiste à mettre en place un système de cryptage connu sous le nom générique de TDE (Transparent Data Encryption). Cette solution de cryptage est transparente pour les applications (Il n'y a pas besoin de modifier le code applicatif). En général, cette solution permet de protéger les fichiers de la base ainsi que les sauvegardes.

5.3. Audit des accès de l'utilisateur :

L'audit des accès est nécessaire pour déterminer l'utilisateur responsable de telle action dans la base de données, et comme les utilisateurs accèdent à partir d'un intermédiaire, il est difficile à un système audit de garder la trace et de corréler les activités qui peuvent être sensitives à la sécurité. **[26]**

Un outil indispensable pour assurer la sécurité d'une BD est l'audit basé sur un journal des différents types d'accès à la BD :

- Audit des entrées dans la base.
- Audit des utilisations de la BD en dehors des heures ouvrables.
- Audit de la manipulation du schéma.
- Audit des erreurs.
- Audit des modifications des sources des procédures stockées et des
- triggers.
- Audit des modifications des attributs de sécurité (login, privilèges...). **[22]**

5.4. Limitation du privilège de l'intermédiaire [21]

5.4.1. Principe du moindre privilège :

Ce Principe stipule qu'un sujet ne doit disposer que des droits d'accès minimum pour assurer l'exécution des tâches qui lui sont assignées, pas un de plus.

Ex : ne pas donner les droits de l'administrateur à tout utilisateur d'un système (système d'exploitation, SGBD).

5.4.2. Politique de gestion des privilèges :

- **Les privilèges :**

Il convient pour chaque compte d'accès d'identifier les privilèges minima à accorder ainsi que le niveau de granularité adéquat. Ici ; le terme utilisateur désigne une application aussi bien qu'une personne physique.

A. Classes d'objets et granularité

Les SGBD permettent généralement de spécifier assez finement les privilèges d'un utilisateur en fonction des objets manipulé :

- Base de données.
- Table (relation).
- Colonne (attribut).

Ainsi, un utilisateur peut se voir attribuer un privilège pour toute une base de données, ou seulement pour quelques tables, ou encore sur uniquement quelques colonnes de certaines tables.

B. Classes de privilèges :

Les privilèges s'organisent autour de plusieurs classes :

- Accès au contenu de l'information.
- Gestion du schéma de la base de données.
- Gestion des privilèges utilisateurs.
- Gestion des paramètres systèmes.

C. Règles d'attribution des privilèges :

Règle fondamentale n°1 : attribution du moindre privilège.

Les utilisateurs ne doivent avoir que le minimum de droits, ceux strictement nécessaires à l'accomplissement de leurs tâches.

Règle n°2 : contrôle de la population.

Les privilèges doivent êtres synchrones avec la réalité de la population : il faut supprimer les comptes des utilisateurs quittant l'entreprise et de ceux n'étant plus affectés à telle ou telle tâche.

Règle n°3 : supervision de la délégation des tâches d'administration.

Un administrateur peut être amené à déléguer auprès d'une autre personne les tâches d'attribution des privilèges de tout ou partie de la population des utilisateurs. Un contrôle a posteriori doit être réalisé afin de vérifier que le résultat de cette délégation est conforme à la politique adoptée.

Règle n°4 : contrôle physique des connexions.

Il est nécessaire de restreindre les connexions à des hôtes spécifiques connus.

***Règle n°5* :** limitation des ressources utilisées.

Le SGBD offre souvent la possibilité de restreindre les ressources de calcul disponibles pour un utilisateur. Il est recommandé de configurer ces limitations de ressources en fonction de la charge maximale attendue pour un utilisateur.

Règle n°6 : journaliser les comportements suspects.

Certains SGBD permettent de conserver dans des journaux de log les requêtes non conformes aux privilèges accordés à un utilisateur. Il peut être intéressant de les surveiller afin de détecter toute anomalie dénotant des tentatives de piratage.

Règle n°7 : restrictions sur une application en fonction du public.

Une même application web peut avoir plusieurs interfaces différentes selon le contexte d'utilisation : internet / intranet.

D. Contrôle des privilèges :

La principale question qui se pose lors du développement d'une application, c'est quelle stratégie adopter vis à vis des utilisateurs : contrôle de leurs droits d'accès par l'application ou par le SGBD ?

- ***Par le SGBD :*** Dans le cas où toute l'information métier repose sur une base de données comportant également toutes les procédures stockées de contrôle de l'intégrité, du logique métier et des actions utilisateurs, il est logique de déléguer au SGBD le contrôle d'accès et les habilitations. Ceci suppose que l'administrateur de bases de données réalise les opérations d'attribution des privilèges et de synchronisation avec l'annuaire des utilisateurs du système d'information de l'entreprise. L'application ne devient alors qu'une interface graphique ergonomique d'interrogation de la base de données métier.
- ***Par l'application :*** Dans le cas où l'application gère elle-même le niveau d'accréditation des utilisateurs, elle va se connecter sous sa propre identité logique à la base de données et décider des informations et des opérations que l'utilisateur peut voir, modifier et réaliser. C'est la stratégie employée par les applications dont la logique métier n'est pas intégrée directement dans la base de données et qui gèrent plusieurs sources de données.

E. Contrôle d'inférence : L'objectif du contrôle d'inférence est protéger une BD des attaques consistant à déduire des données non autorisées à partir de données autorisées.

Ex : Interdire l'accès à des données individuelles dans une BD statistiques à partir de requêtes agrégatives (comptage, somme, moyenne).

6. La protection d'une base de données :

Selon la référence **[28]** la protection d'une base de données suit les étapes suivantes:

6.1 Connaître son besoin :

La sécurité de la base de données commence par une réflexion sur les usages et la population d'utilisateurs accédant à celle -ci, ainsi que sur la manière dont la connexion s'effectue. Est-ce directement par les utilisateurs ou par le biais d'un applicatif (interface Web, progiciel, etc.). Il est indispensable de connaitre la méthode et la nature des accès afin de définir une politique de sécurité adaptée.

"La connexion d'un SGBD avec un progiciel, qui nécessite une méthode d'interconnexion spécifique, peut avoir pour effet d'abaisser le niveau de sécurité. Les équipes sécurité et intégration, dont les missions ne sont pas forcément en accord, doivent souvent trouver un accord".

6.2. Une sécurité en amont :

Le déploiement d'une base de données est souvent la brique d'un projet plus global. La sécurité doit donc être pensée pour l'ensemble des éléments, surtout dans le cas d'un applicatif accédant à la base. Celle-ci peut être protégée mais si l'outil utilisé pour s'y connecter est vulnérable, il ouvrira des portes. Un SGBD ne pourra pas faire la différence entre une connexion légitime et une attaque par le biais d'un frontal Web.

6.3. Supervision :

Un suivi des indicateurs de la base de données doit être assuré afin de détecter les anomalies, prévenir les interruptions de service et intervenir dans les meilleurs délais. La majorité des SGBD du marché embarquent désormais des systèmes de supervision. Charge ensuite à l'administrateur de base de données (DBA) de concevoir des filtres appropriés pour diagnostiquer toute évolution du mode de fonctionnement de la base.

6.4. Sensibiliser les DBA :

L'administrateur doit être sensibilisé aux problématiques de sécurité, aux risques, à la criticité des contenus dont il a la charge et pas seulement à la performance. Un DBA peut

avoir à superviser une dizaine de bases sans bénéficier de visibilité sur les données qu'elles hébergent et risquer par conséquent de ne pas avoir les bons réflexes.

6.5. Durcir le socle système :

Une base de données repose sur une couche système. Cette dernière ne doit donc pas être négligée et faire l'objet d'un durcissement fort. Une base de données ne sera pas en mesure de se défendre contre une personne détenant des droits administrateur sur l'OS. Ce durcissement comprend l'application d'une politique de gestion des correctifs et du moindre privilège, la l imitation des services (réseau et système) et applicatifs, la segmentation des droits ou encore une authentification via des mots de passe fort. Attention au paramétrage des SGBD lors de migration de versions.

6.6. Renforcer la couche BD :

Tout comme le système, les correctifs de sécurité doivent être appliqués à la base de données. Pour des exigences de disponibilité, le patch management est cependant complexifié. Il faut veiller en outre au durcissement de l'installation par défaut.

6.7. Gestion des comptes :

Les comptes par défaut doivent être verrouillés et les mots de passe remplacés pour respecter les normes de sécurité. La notion de politique du moindre privilège s'applique. C'est-à-dire qu'un utilisateur n'ayant par exemple besoin que de consulter les données ne doit en aucune façon disposer de droits en écriture. De même, la base doit être correctement segmentée pour qu'une habilitation ne concerne qu'un périmètre défini des données.

6.8. Méthodes d'accès :

L'entrée sur la base de données doit être autorisée selon des méthodes précises.

C'est à ce niveau que le filtrage sera défini. Si la connexion se fait depuis une application Web, alors seule celle-ci et le DBA seront autorisés à accéder. Ce filtrage est toutefois complexifié lors de l'intégration avec un PGI ou de connexion depuis une application en client lourd installée sur de nombreux postes.

Interviennent alors des aspects de gestions des profils et des utilisateurs, d'évolution des droits. Une cartographie des données et des habilitations doit être dressée pour définir les types de populations accédant à la base et les parties de celle-ci qu'ils sont autorisés à consulter.

6.9. Chiffrer les flux de données :

Les informations envoyées en réponse à une requête ne doivent pas circuler en clair sur le réseau. Nul besoin de durcir l'accès et l'OS, s'il suffit d'écouter le trafic réseau. Les flux seront par conséquent chiffrés entre la base et les différents composants.

Conclusion :

La sécurité des accès à une base de données est une préoccupation de tous les instants. Les privilèges doivent être restreints à l'indispensable et être actualisés régulièrement.

Pour cela, dans ce chapitre, on a essaie de voie les notions de base pour la sécurité des accès à une base de donnée.

Chapitre 4 : Modélisation et Conception.

Introduction :

De nos jours, les bases de données sont des composants incontournables de serveurs Web et d'applications en ligne, qui fournissent du contenu dynamique. Ils permettent d'organiser efficacement la sauvegarde et la lecture des données d'un programme, des données secrètes ou critiques peuvent être stockées dans les bases de données, il est donc important de les protéger efficacement.

Dans le cadre de notre PFE est afin d'implémenter des mécanismes de sécurité, nous proposerons de réaliser une application qui intégrera un ensemble de contrôle d'accès. Pour faire cela, et dans le cadre de ce chapitre, nous présenteront la modélisation de notre application qui intègre des contrôles d'accès que nous avons proposés. La modélisation est basée sur le langage UML (Unified Modelisation Language).

I. Environnement du projet et outils utilisés :

Dans ce projet, nous avons utilisé le langage de programmation Java sous NetBeans8.0.1.

1. Java sous NetBeans :

Java est un langage robuste qui peut être exploité pour développer un large éventail de programmes utilisant une interface utilisateur graphique, pouvant être appliqués en réseau pour se connecter à des bases de données, et offrant d'autres fonctionnalités toutes plus sophistiquées les unes que les autres.

Le ramasse-miettes intégré à Java détecte automatiquement les objets inutilisés pour libérer la mémoire qu'ils occupent. Aussi le Java intègre la gestion des exceptions pour faciliter la mise au point des programmes (détection et localisation des bugs). Concernant la sécurité, Java protège les informations sensibles de l'utilisateur et le système d'exploitation de sa machine en empêchant l'exécution des programmes conçus de façon malintentionnée.

La bibliothèque fournie en standard avec Java couvre de nombreux domaines (gestion de collections, accès aux bases de données, interface utilisateur graphique, accès aux fichiers et au réseau, utilisation d'objets distribués, XML…, sans compter toutes les extensions qui s'intègrent sans difficulté à Java) donc la bibliothèque très riche. **[23]**

Java est enfin nativement doté d'un ensemble complet de primitives de gestion du multitâche simplifiant grandement l'écriture de programmes par exemple devant exécuter des requêtes sur une base de données. L'API JDBC (Java DataBase Connectivity), apparue dès la JDK 1.1, permet de développer des applications capables de se connecter à des serveurs de bases de données (SGBD), par le biais d'un pilote. Nous avons utilisé un tel pilote (com.mysql.jdbc.Driver) de manière à pouvoir se connecter à un serveur mySQL.

2. MySQL :

L'emploi de bases de données était absolument indispensable afin de sauvegarder de manière efficace l'ensemble des tables, qui à long terme peuvent constituer un bloc très volumineux de données.

Nous avons opté pour MySQL (My Structured Query Language) pour plusieurs raisons parmi les quelles la disponibilité de driver pour se connecter à une base de données à partir de Java. Mais aussi, car il offre un système optimisé de gestion de base de données, et ses commandes sont plutôt faciles d'emploi.

3. UML :

Pour dessiner des diagrammes UML, il existe plusieurs outils disponibles en Open Source (ArgoUML, StarUML, Poséidon, etc.) ou sous forme de plug-in pour Eclipse ou NetBeans. Pour ce qui suit nous avons utilisé StarUML. Pour exprimer nos besoins, nous avons utilisé le formalisme UML des cas d'utilisation et des séquences.

4. L'algorithme de cryptage MD5 :

MD5 (*Message Digest 5*) est une fonction de hachage cryptographique qui calcule, à partir d'un fichier numérique, son *empreinte numérique* (en l'occurrence une séquence de128 bits ou 32 caractères en notation hexadécimale) avec une probabilité très forte que deux fichiers différents donnent deux empreintes différentes.

En 1991, Ronald Rivest améliore l'architecture de MD4 pour contrer des attaques potentielles qui seront confirmées plus tard par les travaux de Hans Dobbertin. **[29]**

Comme toute fonction de hachage cryptographique, MD5 peut aussi être utilisé pour calculer l'empreinte d'un mot de passe avec la présence d'un sel permettant de ralentir une attaque par force brute. Cela a été le système employé dans GNU/Linux. Ainsi, plutôt que de stocker les mots de passe dans un fichier, ce sont leurs empreintes MD5 qui sont enregistrées, de sorte que quelqu'un qui lirait ce fichier ne pourrait pas découvrir les mots de passe. La commande *enable secret* des commutateurs et routeurs Cisco, utilisait le hachage MD5 (5 pour indiquer MD5) pour stocker le mot de passe du mode privilégié dans le fichier de configuration de l'équipement.

Exemple :

Voici l'empreinte (appelée abusivement *signature*) obtenue sur une phrase : MD5("Wikipedia, l'encyclopedie libre et gratuite") = d6aa97d33d459ea3670056e737c99a3d

En modifiant un caractère, cette empreinte change radicalement :

MD5("Wikipedia, l'encyclopedie libre et gratuit**E**") = 5da8aa7126701c9840f99f8e9fa54976

II. Modélisation avec UML :

Dans le cadre de notre PFE, on s'intéresse à la sécurité des bases de données dans les systèmes de filtrage d'information.

Notre objectif est de proposer des mécanismes de contrôles d'accès afin d'améliorer la sécurité d'une application informatique ainsi qu'un mécanisme de chiffrement afin de protéger les informations pertinentes de la base de données. Cette application contient deux parties : la première est réaliser afin de gérer les fonctionnalités de l'Administrateur **(admin)**, et La deuxième partie est une implémentation des droits d'accès des utilisateurs **(user).**

À ce stade, nous avons réalisé une modélisation qui est présenter dans cette section, pour que l'utilisateur (ou admin) accède a l'application, il doit passer d'abord par la phase d'authentification, il entre le nom d'utilisateur et son mot de passe qui vont être transformé en mot miroir et crypté par le MD5 le système les comparer avec les données crypter et sauvegarder dans la base de données.

Si le résultat de la comparaison est négatif le système affiche un message d'erreur, Sinon le système génère un code de sécurité et l'envoie à la boite email de l'utilisateur pour qu'il le récupère et le saisie, le système va vérifier si le code saisie est le même code générer, si oui l'utilisateur (admin) accède a l'application, sinon un message d'erreur va être afficher et même on a intégrer un mécanisme de blocage d'accès après trois essaie d'accéder a l'application.

1. Diagramme des cas d'utilisation :

Les diagrammes des cas d'utilisation identifient les fonctionnalités fournies par le système (cas d'utilisation), les utilisateurs qui interagissent avec le système (acteurs), et les interactions entre ces derniers. Les cas d'utilisation sont utilisés dans la phase d'analyse pour définir les besoins de "haut niveau" du système **[24].**

Les acteurs humains qui utilisent le système sont les suivants : Administrateur et utilisateurs. Les diagrammes des cas d'utilisation se présentent comme ci-dessous selon chaque acteur. (Utilisateur, administrateur).

2.1. Premier acteur : utilisateur

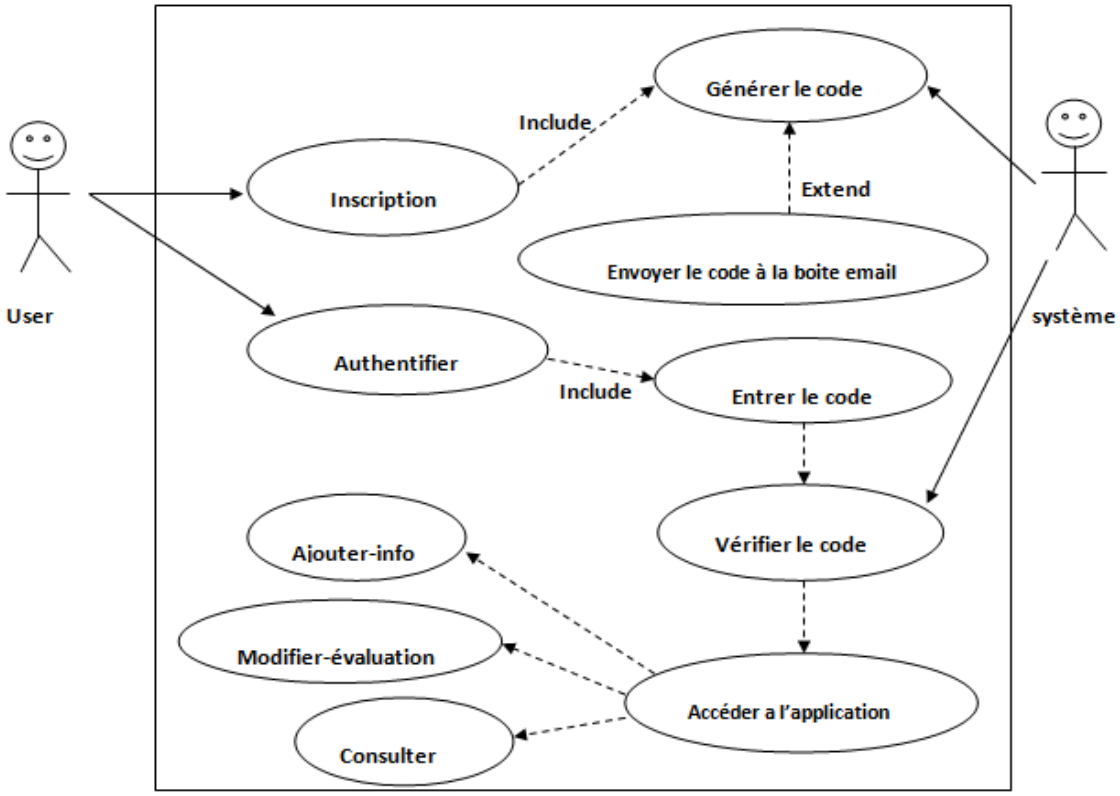

Figure IV.1 : Diagramme de cas d'utilisation « **acteur utilisateur**».

L'utilisateur entre le nom d'utilisateur et son mot de passe, et le système le génère un code qui sera réenvoyer a la boite email de l'utilisateur, il récupère ce code et il va le saisie, si le code est juste l'utilisateur peut accéder a l'application si non il a le droit de le saisir seulement 2 fois puis le système lui affiche un message d'erreur.

Un utilisateur peut ajouter ses information, modifier une évaluation sur un film non évaluer, et consulter ses informations.

2.2. Deuxième acteur : Administrateur

Un administrateur peut ajouter, modifier, supprimer, un utilisateur et filtrer les items.

Apres que l'admin sera authentifier, et le système génère un code de sécurité composer de 6 chiffres, qui sera renvoyer a sa boite email , l'admin va saisir ce code et le système lui vérifier si il est juste il donne la main a l'administrateur pour accéder a l'application si non il le donne la possibilité de saisir le code seulement 2 fois puis le système lui affiche un message d'erreur.

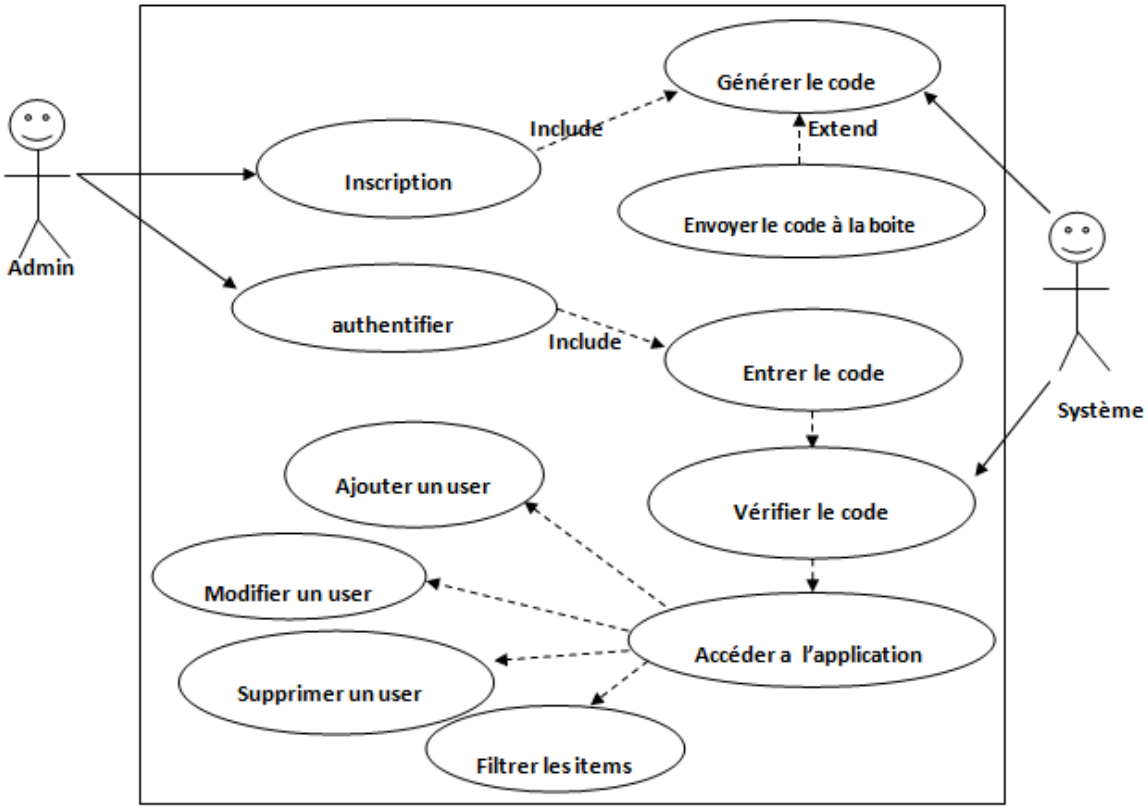

Figure IV.2 : Diagramme de cas d'utilisation « **acteur administrateur**».

Chaque cas d'utilisation représenté dans le diagramme précédent doit être complété par un diagramme des séquences.

1. Diagramme des Classes :

Le diagramme des classes identifie la structure des classes d'un système, y compris les propriétés et les méthodes de chaque classe**. [24]**

Le diagramme des classes est le diagramme le plus largement répandu dans les spécifications d'UML car il fait abstraction des aspects temporels et dynamiques.

Dans ce qui suit, nous avons présenté d'une façon globale les classes qui sont liées avec elles par des relations, selon certaine condition *figure IV.3*

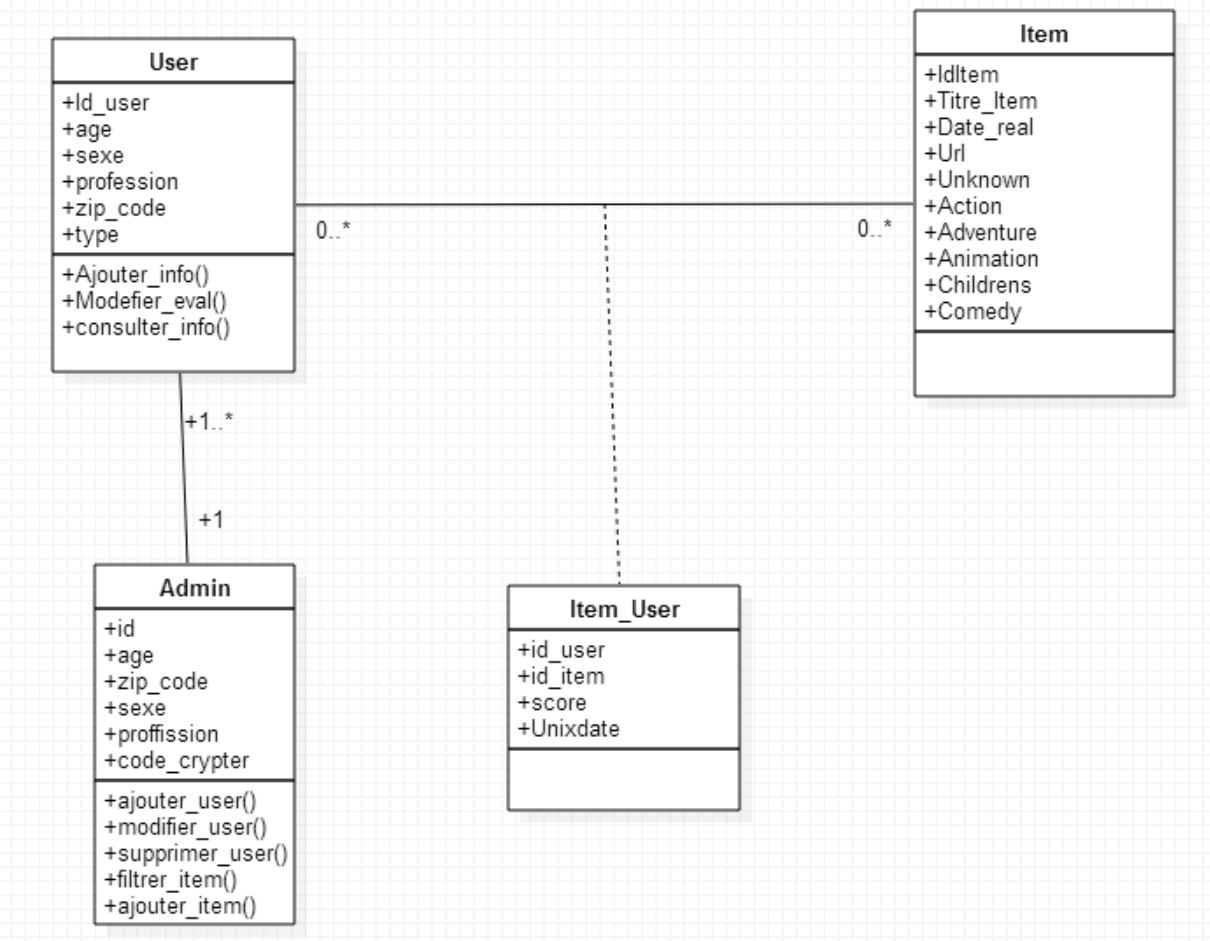

Figure IV.3 : Diagramme des classes.

3. Diagramme des séquences :

Les diagrammes des séquences documentent les interactions à mettre en œuvre entre les classes pour réaliser un résultat, un cas d'utilisation. UML étant conçu pour la programmation orientée objet, ces communications entre les classes sont reconnues comme des messages. Le diagramme des séquences énumère des objets horizontalement, et le temps verticalement. Il modélise l'exécution des différents messages en fonction du temps**. [24]**

Dans ce qui suit, nous intéresses a présenter le diagramme de séquence pour le cas d'utilisation « authentification » :

3.1. Diagramme de séquence de cas d'utilisation : « Authentification »

Le digramme est comme suit :

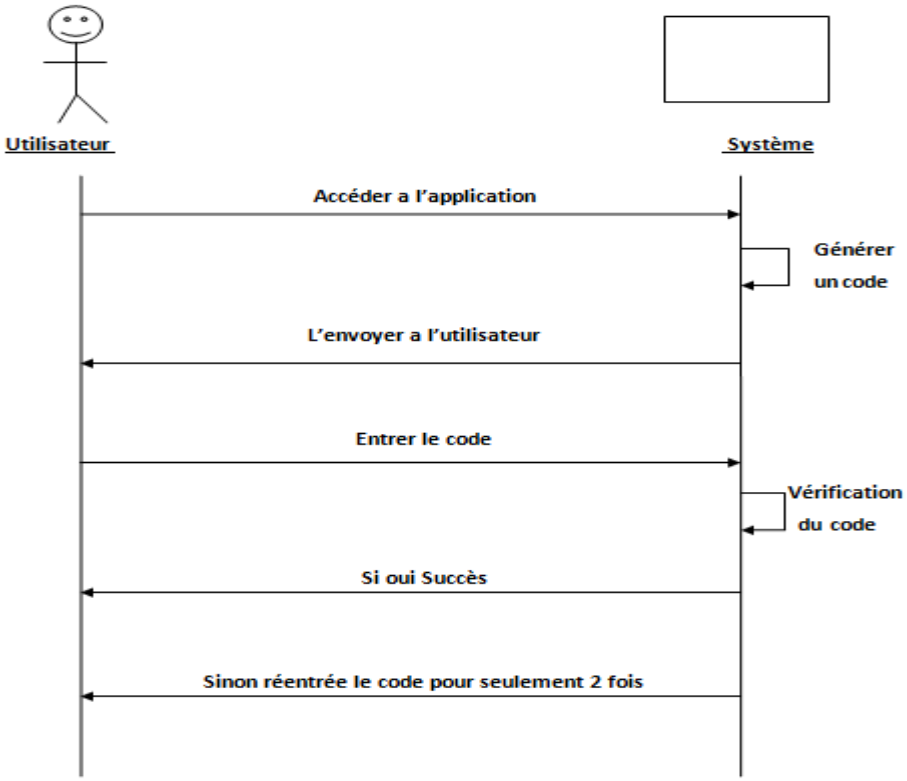

Figure IV.4 : Diagramme de séquence de cas d'utilisation « Authentification utilisateurs ».

Dans ce diagramme, on propose un contrôle d'accès pour authentifier un utilisateur, notre idée est basée sur la généralisation d'un code de sécurité après entrer le nom d'utilisateur et le mot de passe correct.

4. Diagramme d'activité :

Les diagrammes d'activités permettent de mettre l'accent sur les traitements. Ils sont donc particulièrement adaptés à la modélisation du cheminement de flots de contrôle et de flots de données. Ils permettent ainsi de représenter graphiquement le comportement d'une méthode ou le déroulement d'un cas d'utilisation.

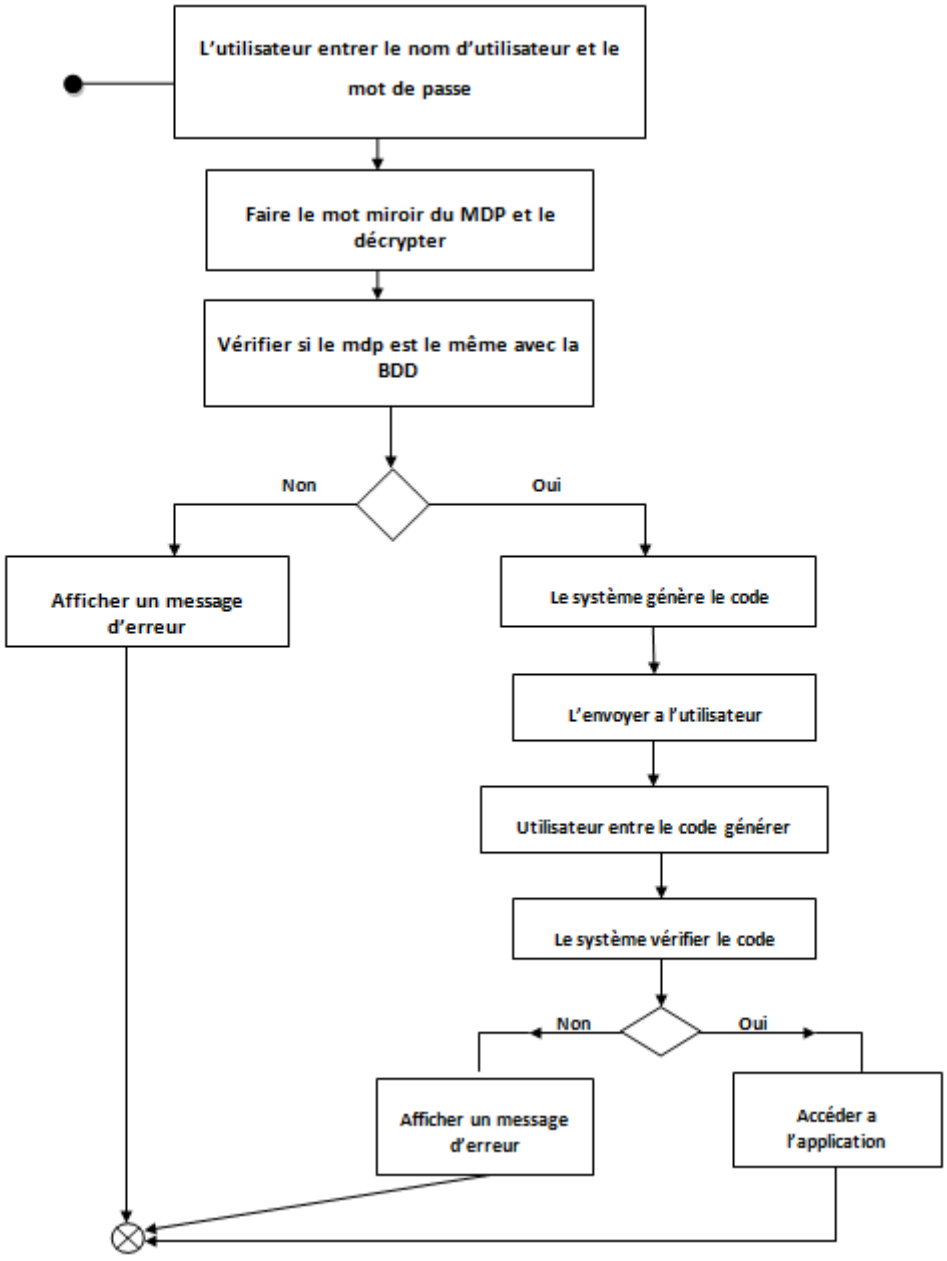

Figure IV.5 : Diagramme d'activité.

Dans notre travaille, l'utilisateur ou admin accède a l'application, il entre le nom d'utilisateur et son mot de passe qui vont être transformé en mot miroir et crypté par le MD5

le système les vérifier si ils correspondent au données crypter a la base de données (on a choisie d'utiliser l'algorithme MD5 pour le cryptage de données).

Si non le système affiche un message d'erreur, si oui le système génère un code de sécurité et l'envoie à la boite email de l'utilisateur.

Apres que l'utilisateur le récupère, il va le saisie, le système vérifier si le code saisie est le même code générer, si oui l'utilisateur accède a l'application, sinon le système lui affiche un message d'erreur et le donne la possibilité pour ré-entrer le code seulement deux fois puis il le bloque.

III. Réalisation de l'application :

Dans cette dernière partie, nous allons présenter l'implémentation de notre application et la démarche de chaque fonctionnalité réalisée, nous allons développer tout au long de cette étude une application pour la sécurisation d'une base de données d'un système de filtrage MovieLens.

Notre application se compose d'une partie liée à l'Administrateur, et d'une autre partie concernant l'accès utilisateurs.

1. Authentification :

Avant que l'admin accède à l'application il doit d'abord passé par l'authentification :

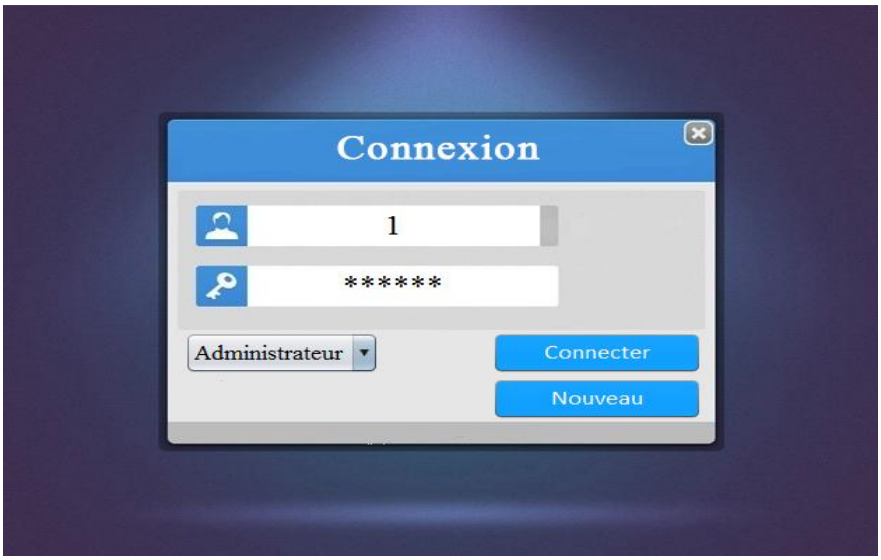

Figure IV.6 : authentification 1.

A ce stade l'admin ou le user entre son nom d'utilisateur et mot de passe et il va choisi son type (par exemple administrateur), le système va faire le mot miroir de mot de passe et le crypter par l'algorithme MD5, et il va le comparé par le mot de passe crypter sauvegarder a la basse de données, si le résultat est négatif il donne la possibilité de ré-entrer son mot de passe seulement deux fois puis il bloque l'accès.

Si le résultat est positif, le système va générer un code de sécurité qui va être envoyé a la boit email de l'admin (user) préalablement configurer a l'application.

L'admin (user) récupère son code de sécurité et il le saisie, le système va vérifier si le code saisie est le même code envoyer, puis il donne l'accès a l'admin (user).

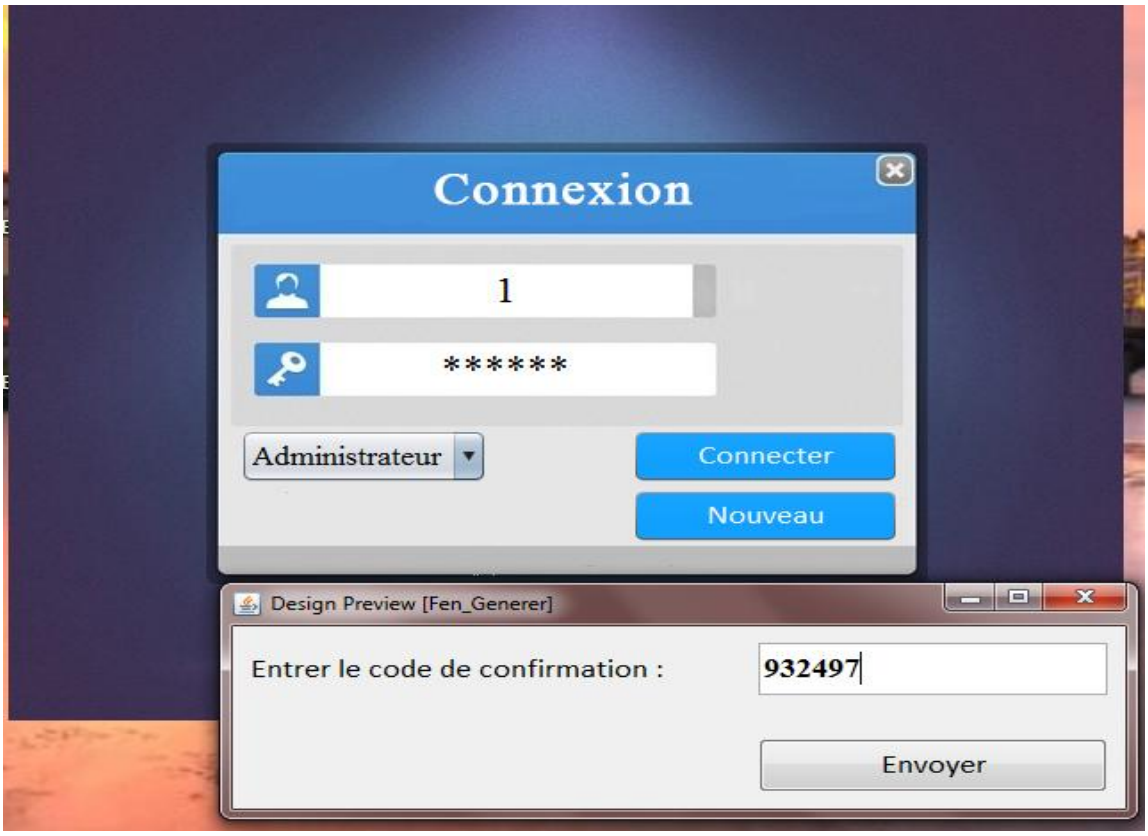

Figure IV.7 : authentification 2.

Si un utilisateur n'a pas un compte, il clique sur nouveau pour inscrire et la fenêtre suivante va être affichée, elle contient l'âge, sexe, profession, code_zip ou mot de passe, et la confirmation de code_zip et en fin l'adresse email pour la configurer avec l'application afin de retenir un code de sécurité et l'ID qui est définit comme clé primaire et qui va être remplie automatiquement de la base de données et non modifiable.

Quant le nouveau utilisateur remplie ce formulaire il clique sur valider pour conserver le données.

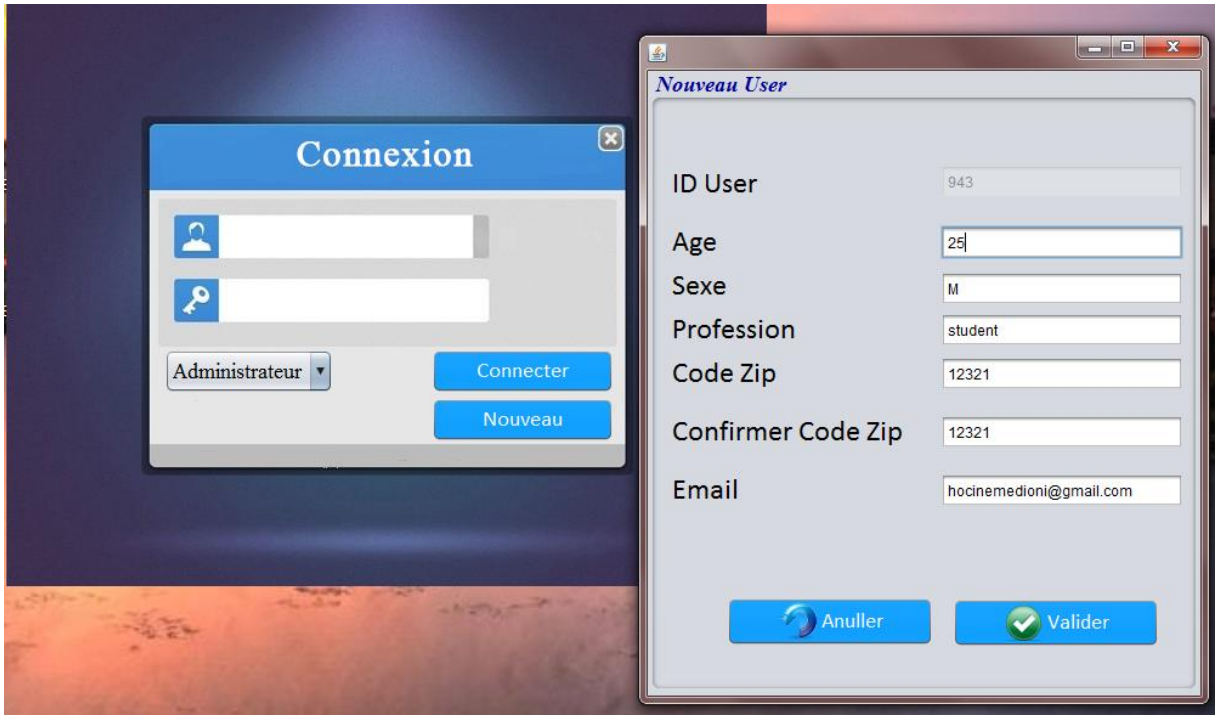

Figure IV.8 : Inscription d'un nouveau user.

2. Partie administrateur :

Après la phase d'authentification, l'admin accède a son espace, ou il peut ajouter, modifier, supprimer un user et filtrer les films qui sont évalués d'un user comme montre les figures suivantes :

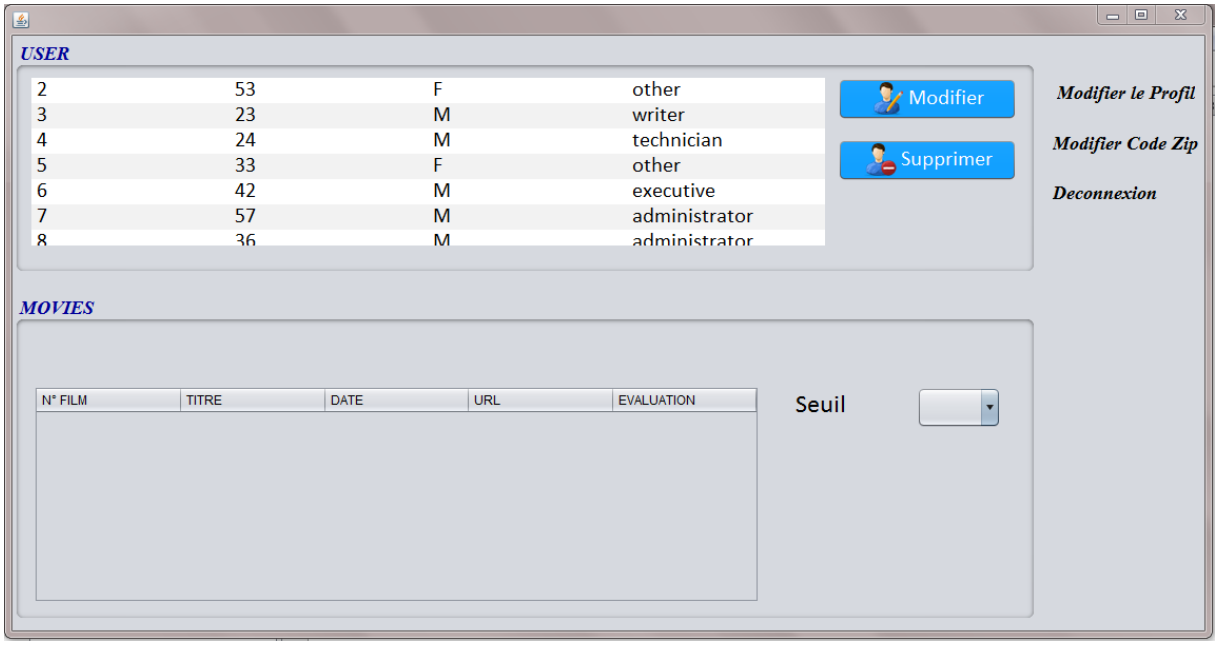

Figure IV. 9: Espace administrateur.

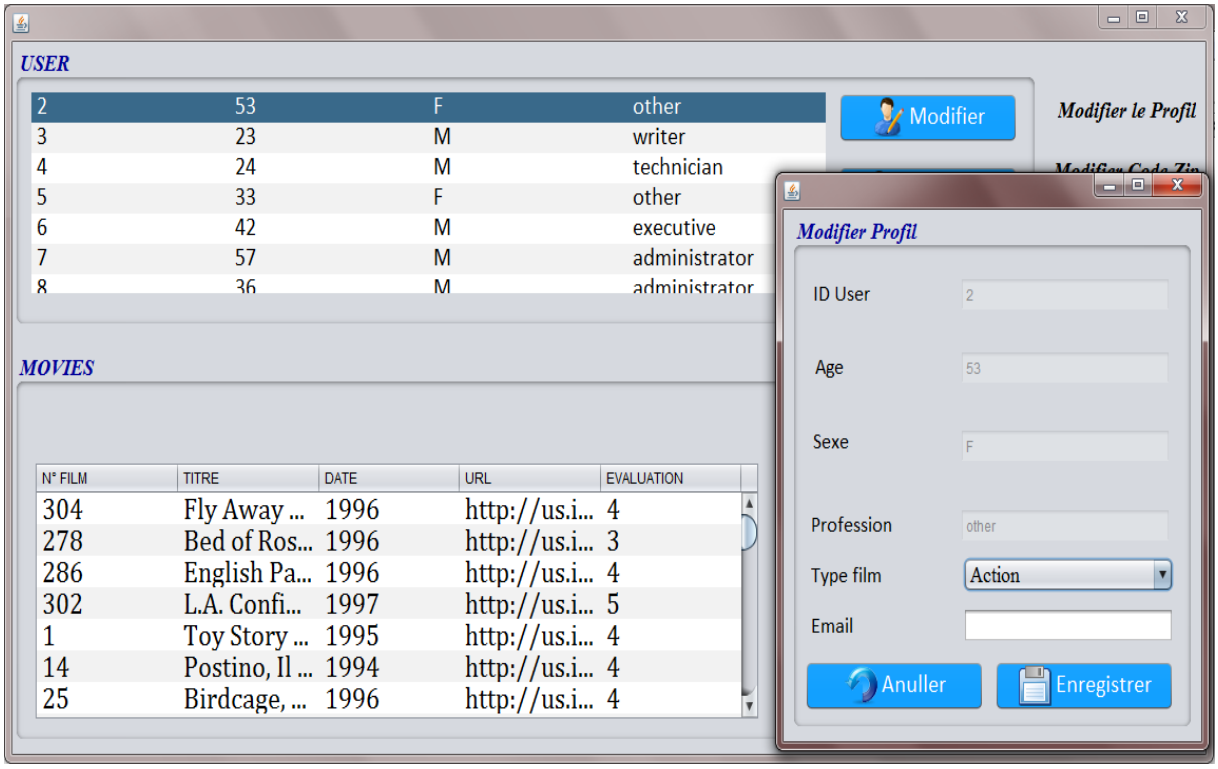

Figure IV.10 : Espace administrateur « Modifier un user ».

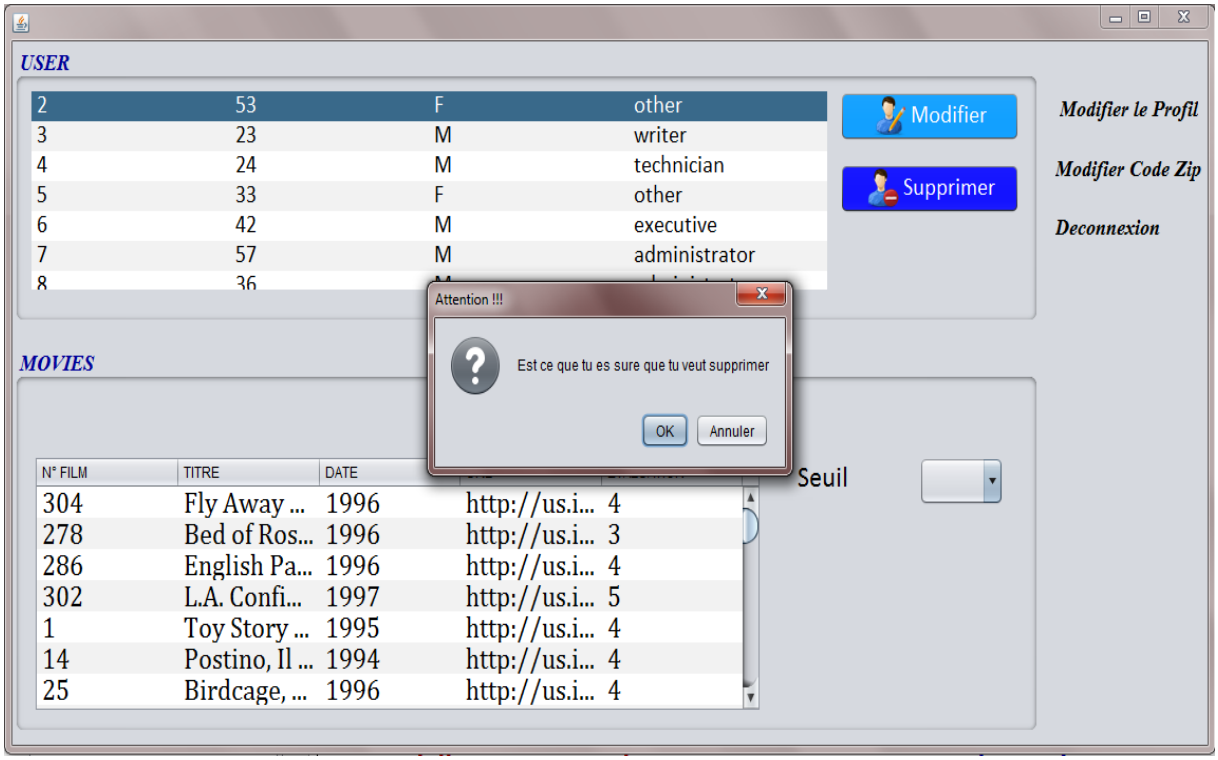

Figure IV.11 : Espace administrateur « Supprimer un user ».

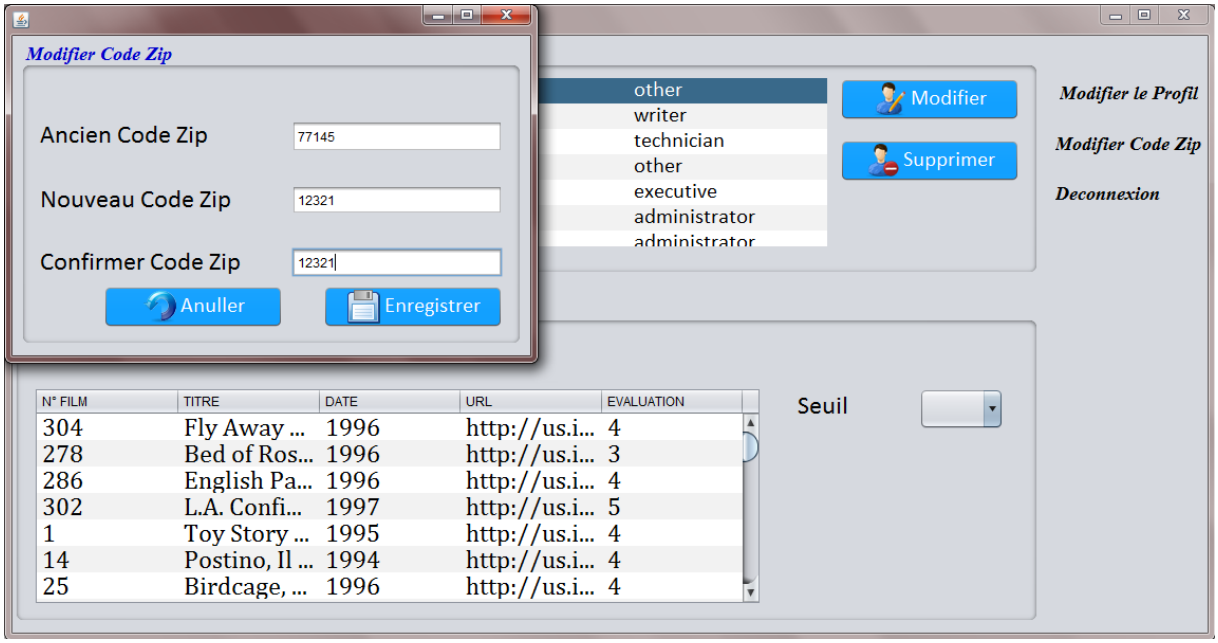

Figure IV.12: Espace administrateur « Modifier code zip ».

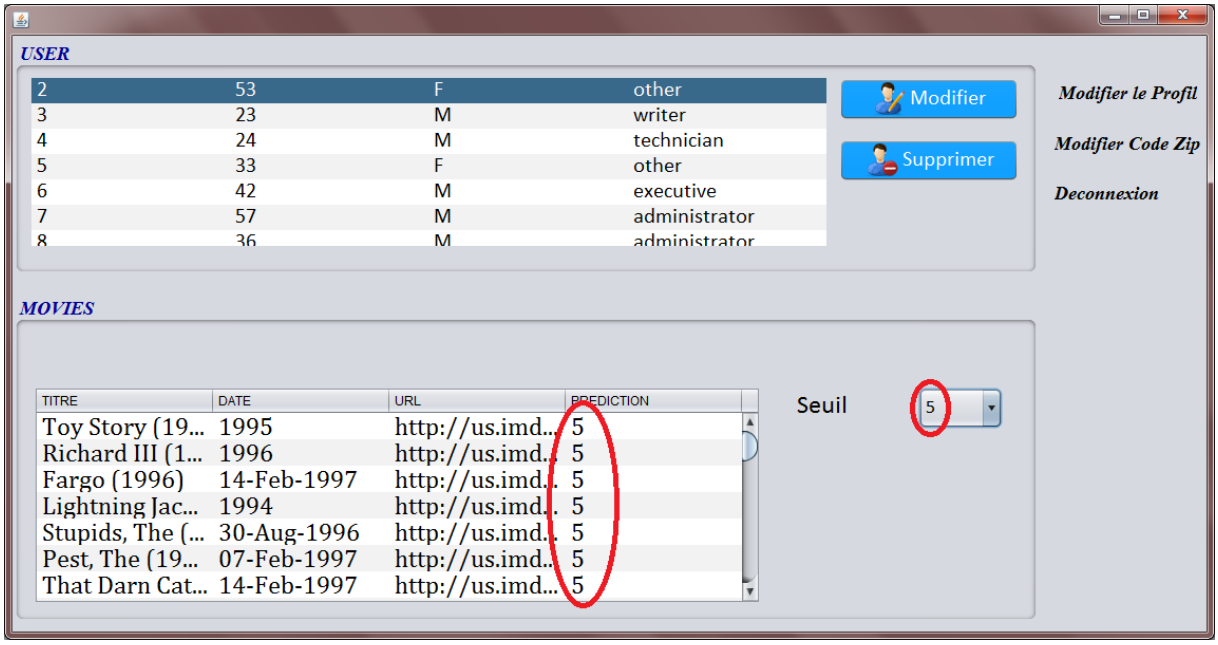

Figure IV.13: Espace administrateur « filtrer les items d'un user ».

Dans cette étape, le filtre se fait selon le seuil d'évaluation de ses items, qui sont compris entre le **1 et 5**. L'admin ne peut pas changer l'évaluation d'un film (item) car il n'a pas le droit de la modifier par contre le user a tout le droit de le faire dans un système de filtrage dynamique (les droits d'accès).

3. Partie utilisateur :

Pour que l'utilisateur accède à l'application il doit passer par l'authentification comme le cas avec l'administrateur. Un user peut ajouter des information ou les modifier, modifier l'évaluation d'un film, et consulter les information dans son profil et d'autre opération présenter dans les figures suivantes :

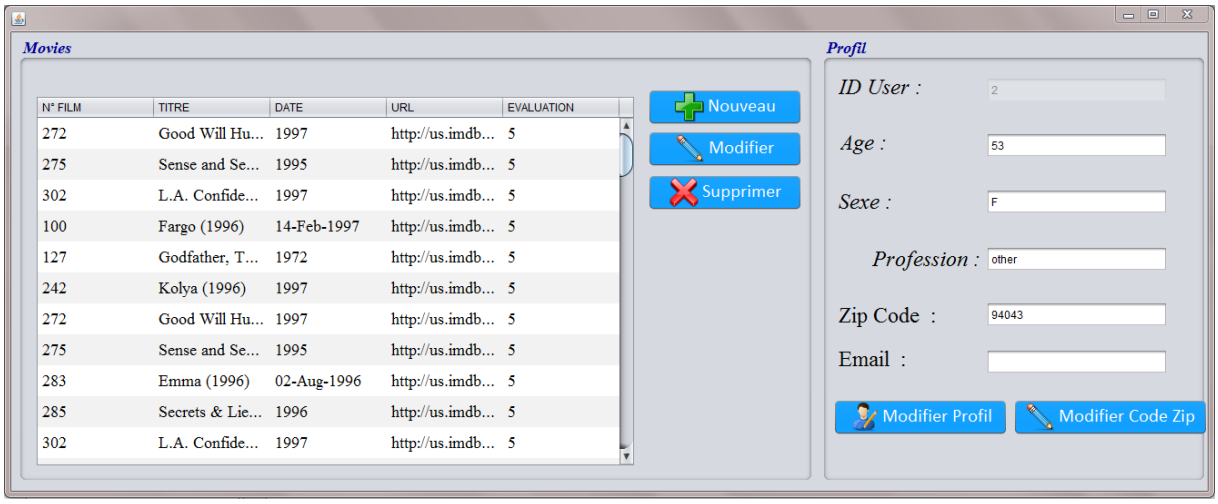

Figure IV. 14: Espace utilisateur.

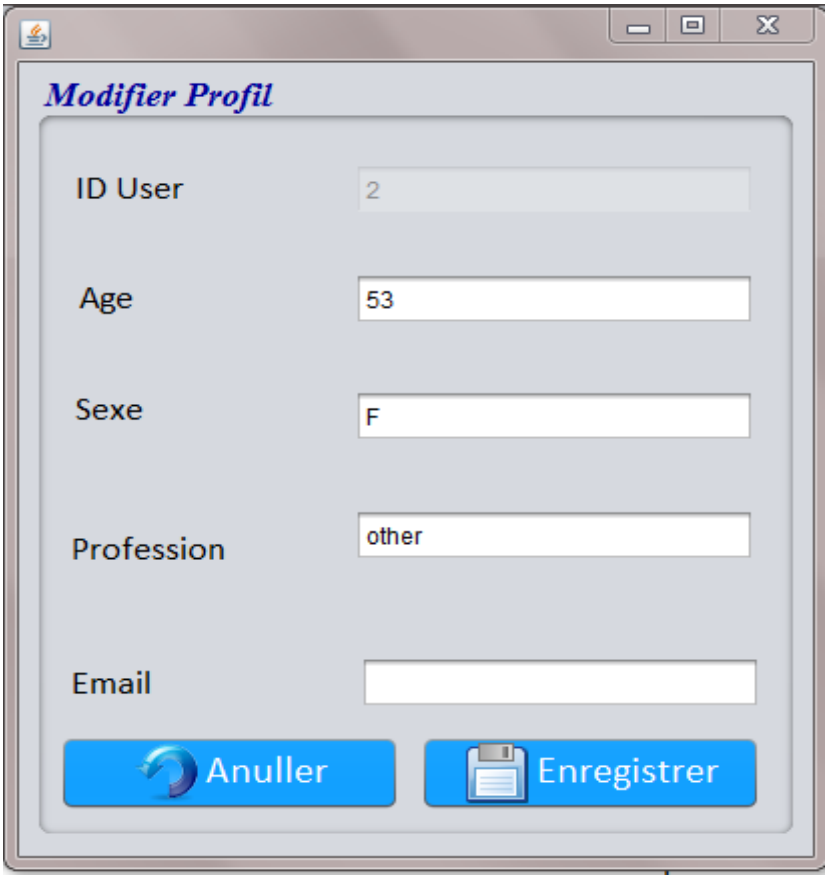

Figure IV. 15: Espace utilisateur « modifier profil ».

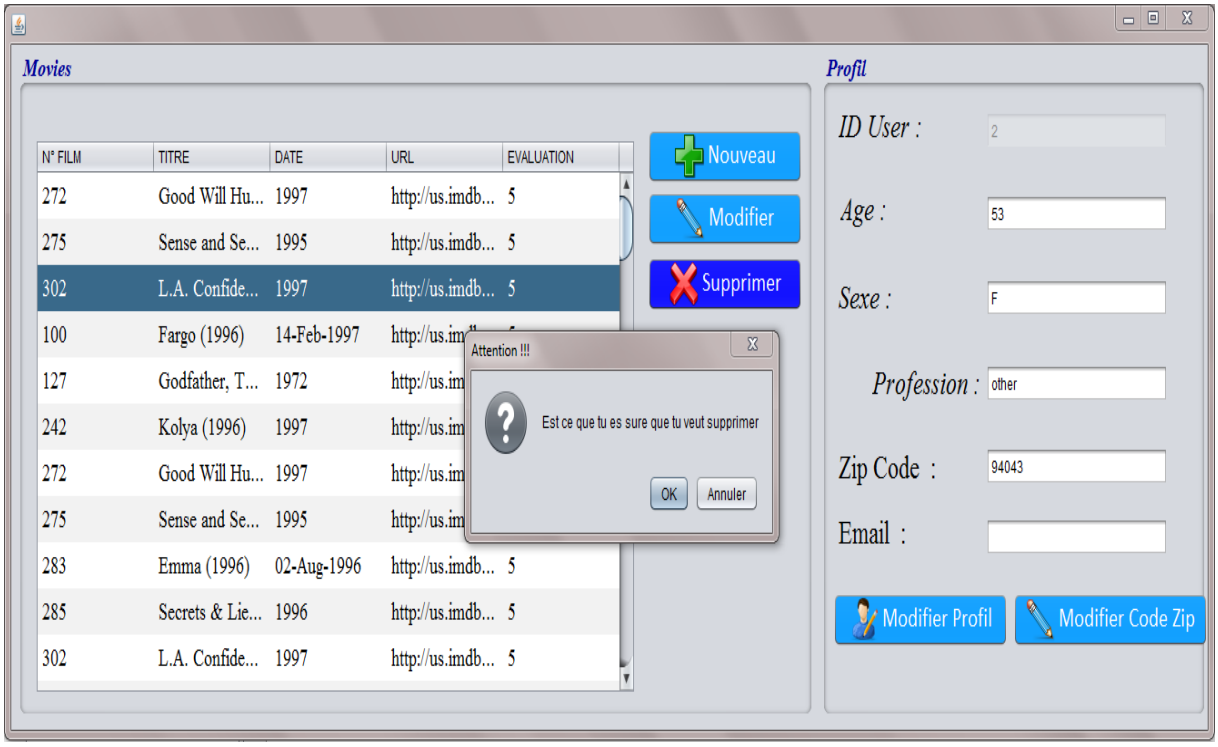

Figure IV. 16: Espace utilisateur « supprimer film ».

Figure IV. 17: Espace utilisateur « modifier évaluation d'un film ».

Conclusion :

Dans ce chapitre, nous avons présenté en premier temps l'étude de cas de notre application qui est réalisé pour la sécurisation d'une base de données d'un système de filtrage ainsi que les acteurs qui l'utilisent. Le diagramme de cas d'utilisation nous a permis de formaliser les besoins de manière synthétique.

Aussi la présentation de notre application, cette dernière contient deux partie, la première est une application réalisée pour gérer les utilisateurs ; nous avons proposé dans cette partie un contrôle d'accès sur le nombre de tentatives erronés d'authentification avant de bloquer l'accès ainsi qu'un mécanisme de chiffrement afin de protéger les informations pertinentes de la base de données. Cette dernière opération est totalement transparente par l'utilisateur car elle est réalisée par le système.

La deuxième partie de notre application est une implémentation afin de gérer les fonctionnalités de l'Administrateur, à ce stade nous avons proposé un contrôle d'accès, ainsi qu'un mécanisme de chiffrement afin d'assurer l'intégrité, la confidentialité et la disponibilité des informations.

Conclusion générale

Le travail réalisé dans le cadre de ce PFE nous a ramené à étudier plusieurs domaines dans le cadre de la sécurité informatique et notamment en système de filtrage d'information.

Notre mémoire est articulé autours de deux parties :

La première est théorique, il s'agit de l'état de l'art et elle contient trois chapitres, le premier traite les systèmes de filtrage d'information en général, le second parle sur les bases de données, et le troisième est une vue globale sur la sécurité informatique et la sécurisation des bases de données.

La seconde partie de notre mémoire est la partie pratique, dans laquelle nous avons implémenté une application pour la gestion des droits d'accès pour les administrateurs et les utilisateurs. Nous avons proposé dans ce contexte un contrôle d'accès pour protéger les informations des utilisateurs mais également une méthode de chiffrement basée sur l'algorithme MD5 pour empêcher toute divulgation d'informations liées aux utilisateurs en cas d'attaques visant leurs données.

Pour la réalisation de notre application, une conception a été nécessaire pour mieux contourner toute les fonctionnalités d'utilisation, dans ce cadre nous avons utilisé UML.

Au cours de l'élaboration de notre application, nous avons acquis plusieurs connaissances qui s'avèrent bénéfiques dans le cadre de notre formation et qui sont un complément essentiel pour la consolidation de plusieurs données théoriques acquises tout au long de notre formation académique.

Nous espérons que ce modeste travail profitera largement à notre université ainsi qu'aux futurs étudiants et ouvrira des perspectives sur des applications regroupant d'autres services.

Nous pensons, qu'il serait intéressant d'implémenter un autre algorithme de chiffrement (RSA ou le code de César), pour assurer l'intégrité, la confidentialité, et disponibilité.

Référence:

[1]: Belkin, N. and Croft, W. (1992). Information retrieval and infor-mation filtering : two sides of the same coin ? Communications of the ACM, 35(12).

[2]: Bouzeghoub, M., Berrut, C., Boughanem, M., Doucet, A., and Rumpler, B. (2004). Action spécifique sur la personnalisation de l'information. In Rapport CNRS-AS98/RTP9.

[3]: R.Abbes. (.edition en 1999). livre Le filtrage des informations.

[4]: B.Samia . « Modélisation hybride du profil utilisateur pour un système de filtrage »,2007.

[5]: B. Amokrane. « L'usage des concepts du web sémantique dans le filtrage d'information collaboratif »,2007.

[6]: Malone, T., Grant, K., Turbak, F., Brobst, S., and Cohen, M. Intelligent information sharing systems. Communications of the ACM, 30(5) :390–402. (1987).

[7]: Oard, D. and Marchionini, G. (1996). A conceptual frame-work for text filtering. Report ee-tr-96-25, Universit´e de Maryland.

[8]: Goldberg, D., Nichols, D., Oki, B., and Terry, D. (1992). Using collaborative filtering to weave an information tapestry. ACM SIGIR Forum, 12(35) :61–70.

[9]: Goldberg, K., Roeder, T., Huptan, D., and Perkins, C. (2000). Ei-gentaste : A constant time collaborative filtering algorithm. In Technical Report M00/41, IEOR and EECS Departments. UC Berkeley.

[10]: Berrut, C. and Denos, N. (2003). Filtrage collaboratif. Assistance intelligente `a la recherche d'informations, Hermes - Lavoisier, chapitre 8

[11]: http://www.memoireonline.com/07/10/3701/m_conception-et-realisation-dune-base-de-donnees-pour-la-gestion-de-facturation--loffice-con.html, 04 mai 2011.

[12]: Jacques le Maitre, « les bases de données relationnelles et leurs systèmes de Gestions », université de Toulon et du Var.

[13]: http://www.telechargercours.com/cours-informatique/acces-aux-bases-de-donnees-odbc-et-jdbc manipulations de données.

[14]: Philippe Rigaux, « Cours de bases de données », 13 juin 2001

[15]: http://www.adproxima.fr

[16]: http://www.journaledunet.com, 05 mai 2011

[17]: http://www.guideinformatique.com/index

[18]: http://www.memoireonline.com/02/11/4278/m_Conception-et-realisation-dune-base-de-donnees-repartie-sous-oracle--cas-de-lhebergement-d2.html

[19] : Donald L.P., « Sécurité des systèmes d'information, protection globale de l'entreprise» CampusPress, 2000. ISBN 2-7440-0948-2.

[20]: ISO-7498-2. *Information processing systems - open systems interconnection- basic reference model - part 2: Security architecture*. Technical report, International Organization for Standardization, Geneva, Switzerland, 1989.

[21]: www.Security Handbook.com « RFC 1244 ».

[22]: Jacques Le Maitre, « Sécurité des bases de données », Université du Sud Toulon-Var.

[23]: Nicolase Anciaux, « sécurité et bases de données », source de transparents : Luc bouganim,philipe pucheral, Fridiric, C uppens.

[24]: http://cyberzoide.developpez.com/securite/privileges-base-de-donnees. visité en mai 2016

[25]: msdn.microsoft.com/fr-fr/library/ms164596.aspx, 23 mai 2011.

[26]: http://www.memoireonline.com/07/08/1225/m_mise-en-place-systeme-information-oracle-architecture-trois-tiers10.html, 22 Avril 2011.

[27] : Mathieu_Blanc, l Sécurité des systèmes d'exploitation répartis : architecture décentralisée de méta-politique pour l'administration du contrôle d'accès obligatoire.», Thèse A L'UNIVERSITÉ D'ORLÉANS, Soutenue le : 19 décembre 2006 version 161 Mar 2010.

[28] : http://www.journaldunet.com/solutions/securite/analyses/07/0917 -9-etapes-securiser-sgbd.shtml, 23 mai 2011.

[29]: Vlastimil Klima, « Tunnels in Hash Functions: MD5 Collisions Within a Minute, Cryptology ePrint Archive Report 2006/105 », 17 avril 2006.

[30]: Lalia benathmane. L'optimisation sémantique des systèmes de recommandation. These de majistaire.2010-2011.

[31] : N. Lumineau . « Un tour d'horizon du filtrage collaboratif, Travail réalisé dans le cadre de l'AS Personnalisation de l'information », Laboratoire d'informatique de Paris 6, 2002.

[32] :C.Berrut ,N.Denos . «Filtrage collaboratif», Assistance intelligente à la recherche d'informations, Hermes - Lavoisier, chapter 8, pp30, 2003.