République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Saad Dahlab, Blida
USDB.

Faculté des sciences.
Département informatique .

Mémoire pour l'obtention
d'un diplôme d'ingénieur d'état en informatique.
Option : Système d'information

Sujet :

Intégration d'une Infrastructure de sécurité dans une plate-forme eGouvernement

Présenté par : BENDAOUI Menouar
HOUAIDJI Hafsa

Promoteur : Mdm.SOUAMI.
Encadreur : Bouhini Chahrazed.
Hamzi Abdelhakim.

Organisme d'accueil : CDTA.

Soutenue le: **, devant le jury composé de :**

Président

Examinateur

Examinateur

- Numéro/2007-

Remerciements

*Pour nous avoir fait l'honneur de nous encadrer, pour l'intérêt constant avec lequel ils ont suivi notre travail, pour leurs disponibilités constante et leurs immenses amabilités, que Melle.**Bouhini Chahrazed**, ingénieur d'etat en informatique, et Mr.**Hamzi Abdelhakim**, magistère en informatique au CDTA, reçoivent le témoignage de notre sincère reconnaissance.*

*Aussi, nous adressons nos vifs remerciements à notre promotrice Mdme.**Souami Feryel** pour sa patience, son attention et sa disponibilité pendant tout notre travail.*

A tous ceux qui nous ont aidés à la réalisation de ce travail.

A tous ceux qui nous ont fait confiance.

Dédicaces

Menouar *:*

*Je dédié ce modeste travail à : ma famille (mes parents, mes sœurs « **zoulikha**, **naziha** » et frères « **hamza**, **youcef** »), mes amies, et **moi même** bien sûr.*

Hafsa :

*Je dédié ce travail à mes très chers **parents***

*Mes deux sœurs **Sanaa** et **Ikram***

*A toute ma famille en particulier ma cousine **Fadhila**,*

Et à tous mes amis.

Résumé:

Internet permet de réaliser de nombreuses opérations : échanger du courrier et des documents en temps réel, effectuer des achats en ligne avec une carte bancaire, passer des ordres de bourse, conclure des contrats à distance, etc. Plusieurs problèmes se posent quant à la fiabilité des opérations d'échange et impression des document Web.

L'eGouvernement est l'adoption par l'état des nouvelles technologies de l'information et de la communication pour assurer la marche régulière des services publiques tant pour son fonctionnement qu'en faveur des citoyens. La sécurité informatique représente un défi majeur pour toute application egouvernement telle que les transactions en terme de validité des documents imprimés ou délivrés aux citoyens et fiabilité des information circulées sur le réseau.

Afin de garantir ces besoins, plusieurs mécanismes de sécurité telle que la signature numérique, l'authentification et la garantie de la validité des documents générés par une application eGouvernement sont donc nécessaires .cela se réfère généralement à une suite d'opérations mathématiques sur un document électronique afin de permettre à une personne d'assurer à la fois l'intégrité et l'authenticité d'un document.

Le projet se résume en l'étude des besoins en terme de sécurité pour la réalisation d'un outil permettant l'intégration d'une infrastructure de sécurité au sein des applications eGouvernement.

Mots clés : eGouvernement, Signature Numérique, Cryptage, Certificats, d'authentification....

Abstract:

Internet makes it possible to carry out many operations: to exchange mail and documents in real time, to carry out purchases on line with a bank card, to sign from the Stock Exchange orders, to conclude from the remote contracts, etc. several problems arise as for the reliability of the operations of exchange and impression of the Web document.

EGouvernement is the adoption by the state of new public communication and information technologies to ensure the uniform running of the services as well for its operation as in favour of the citizens.The computer security egouvernement represents a major challenge for any application such as the transactions in term of validity of the documents printed or delivered to the citizens and reliability of information circulated on the network.

In order to guarantee these needs, several mechanisms of safety such as the numerical signature, the authentification and the guarantee of the validity of the documents generated by an application eGouvernement are thus necessary this generally refers to a succession of mathematical operations on an electronic document in order to make it possible to a person to ensure at the same time the integrity and the authenticity of a document.

The project is summarized in the study of the requirements in term for safety for the realization for a tool allowing integration for an infrastructure for safety within the applications eGouvernement

Key words: eGouvernement, digital Signature, Encoding, Certificates of authentification...

Sommaire

Chapitre I : Présentation de projet

Chapitre II : Introduction à l'eGouvernement

Chapitre III : Outils de sécurité

Chapitre IV : Modélisation et conception

Chapitre V : Test et réalisation

Conclusion Générale

Table des figures

Chapitre III : partie A

Chapitre III : partie B

Chapitre IV

Chapitre V

Chapitre I : Présentation du projet

I. Introduction :

Les vulnérabilités et la maîtrise insuffisante des technologies du numérique leur confèrent un certain niveau d'insécurité. Cet état d'insécurité est largement exploité par les acteurs du monde criminel. De plus, chaque technologie est porteuse de potentialités criminelles et offre des opportunités pour réaliser des infractions. L'Internet n'échappe pas à cette règle et le monde criminel a investi le cyberespace.

On assiste de plus en plus a une prise de conscience croissante du besoin de maîtriser les risques informatiques ceci est du au faite de l'usage extensif des nouvelles technologies.

La conversion d'une société à une société dite société de l'information autorisant des nouvelles technologies dans toutes les activités et infrastructures augmente la dépendance des individus, organisations et des états aux systèmes d'informations et aux réseaux.

« Les pays en développement sont confrontés à la problématique de la nécessité de faire partie de la société de l'information en prenant en considération le risque de leur dépendance vis-à-vis des technologies et des fournisseurs de ces technologies et en pensant que la fracture digitale existante ne doit pas se doubler d'une fracture sécuritaire encore moins d'une dépendance plus forte à des entités qui contrôleraient leurs besoins et moyens de sécurité des technologies de l'information » **[ghe 03]**.

La sécurité est la pierre angulaire de toute activité elle doit être vue comme un service permettant de gérer d'autres services (**e-gouvernement**, **e-santé**, **e-éducation, etc ...**). Au delà Des technologies **[nto 05]**,

Or jusqu 'a présent les outils de communication basiques Ne permettent pas de garantir un niveau minimal de sécurité.

Dans ce modeste travail Nous avons essayé de présenter une solution qui devra nous garantir un certain niveau de sécurité au sein d'une plateforme **eGouvernement**.

II. Présentation de projet :

Le rôle de l'eGouvernement est de mettre les technologies de l'information et les moyens de la télécommunication au service du public et des citoyens. Le progrès amené par de l'eGgouvernement est cependant étroitement lié à la qualité des services qu'il offre

[Ghe 06].

Pour offrir des services, en assurant la sécurité et la confidentialité requises, on doit pouvoir fournir les moyens correspondants de protection, ceci s'opère par la signature électronique des documents lors de leur transmission dans les réseaux.

Cependant notre payé n'est pas encore doté de loi sur la signature numérique comme c'est le cas de nos voisins de la Tunisie qui ont leur propre autorité de certification (ANCE) ou le cas de Egypte (ITIDA), notre travail consiste à proposer une solution l'intégration d'infrastructure de sécurité pour une application eGouvernace, à fin de garantir l'authentification et la validité de tout document produit l'eGouvernement.

III. Problématique :

En effet lors de la génération et de la récupération des différents documents ex : (extrait de naissance, acte de mariage,.. etc) sous format électronique via l'application de l'E-gouvernement ; on se trouve confronté au problème de validité de ces documents légaux ; ainsi :

1. quand un citoyen reçoit un document, il doit être sur de l'identité de celui qui le lui a envoyé (c'est l'authentification de l'expéditeur légal).
2. quand un citoyen reçoit un document, il doit être sur que les données transmises ne sont pas incomplètes (Intégrité des données).
3. quand un citoyen reçoit un document, il devra détenir la preuve que seule l'autorité légale et elle seule le lui a envoyé (c'est la Non répudiation par l'identification de l' APC ou autre expéditeur).

IV. Objectifs :

Nous avons de proposé pour remédier aux divers problématiques citées précédemment une stratégie de sécurité, ainsi l'algorithme de cryptage RSA est utilisé pour la signature numérique .un système centraliser généré et gère la paire de clés nécessaire au cryptage pour chaque autorité légale le système centralisé permet aussi aux utilisateurs (citoyens) de vérifier la validité des documents qu'ils détiennent, le système centralisé que nous proposant permet de garantir :

1. L'authentification de l'autorité légale celui qui a généré le document numérique.
2. Vérifier l'intégrité des données présentées sur le document.
3. Garantir La non répudiation, l'expéditeur du document numérique ne pourra nier l'émission.

Chapitre II. Introduction à l'eGouvernement

I. Introduction :

Le secteur public joue un rôle très important dans le modèle social et économique, offrant au citoyen une qualité de vie élevée, et assurant la cohésion socio-économique et un environnement économique concurrentiel. Il est engagé dans toute une gamme d'activités, telles que l'enseignement, les soins de santé et la sécurité sociale, la protection des consommateurs et la protection de l'environnement. Les atouts économiques, tels qu'une main-d'oeuvre qualifiée et la prédominance dans d'importants secteurs nécessitent le soutien d'un secteur public performant.

Le secteur public est à la croisée des chemins. Il doit faire face à une situation économique et sociale difficile, à des changements institutionnels et aux changements importants imposés par des nouvelles technologies

Au sein du secteur public, les administrations doivent améliorer l'efficacité, la productivité et la qualité de leurs services, avec, cependant, des budgets inchangés, voire en réduction.

Les technologies de l'information et des communications (TIC) peuvent aider les administrations à relever ces nombreux défis. L'accent ne doit cependant pas être mis sur la technologie, mais plutôt sur l'utilisation des TIC dans les administrations publiques, *associées avec* des changements au niveau de l'organisation et de nouvelles aptitudes du personnel, dans le but d'améliorer les services publics et de renforcer les processus démocratiques et de soutien aux politiques publiques. C'est ce que l'on désigne par l'**eGouvernement**. **[Ref01]**

II. Concept de l'eGouvernement :

Afin de répondre aux exigences de la direction administrative efficace, des services administratifs qualifiés, un paradigme nouveau est apparu dans lequel le gouvernement électronique est développé.

Le concept d'eGouvernement se rapproche du service bancaire en ligne qu'une banque fournit à ses clients par l'Internet. De même, l'eGouvernement fournit des services publics en ligne aux citoyens et aux entreprises, qui sont donc ses clients.

L'eGouvernement favorise le projet d'informatisation qui mis en application améliore l'efficacité de l'administration publique et de la qualité des services publics .

III. Rôle de l'eGouvernement

L'eGouvernement permet d'améliorer l'élaboration et la mise en oeuvre des politiques et aide le secteur publique à satisfaire les demandes pour des services plus nembreux et améliorées ,avec à l'inverse des ressources moindres.

La technologie est incapable de transformer de mauvaises procédures en bonnes procédures, mais l'eGouvernement offre au secteur public la possibilité d'effectuer ses tâches autrement ; par exemple, grâce à l'amélioration de l'efficacité et au renforcement de la concurrence dans les marchés publics qui permettront les appels d'offres électroniques, on pourra économiser des centaines de millions, qui pourront être réinvestis dans la fourniture de biens et de services publics, ce qui contribuera à la croissance économique. En même temps, la transparence et la responsabilité dans les marchés publics s'amélioreront.

Certains États ont radicalement réduit les délais nécessaires pour créer une nouvelle entreprise, et permettent un enregistrement gratuit en ligne. Cela permet aux entrepreneurs d'utiliser les ressources limitées dont ils disposent pour des activités strictement économiques. Grâce à la possibilité de faire les déclarations de sécurité sociale en ligne, les entreprises évitent déjà de nombreux frais administratifs, ce qui libère des ressources pour la production ou l'innovation.

En autorisant, dans la mesure où le cadre juridique le permet, la réutilisation des informations du secteur public fournies en ligne, on permet aux entreprises d'élaborer des produits plus attrayants et concurrentiels.**[Ref 00].**

IV. Définitions de l'eGouvernement :

Définition 1 :

L'expression « **e-Gouvernement** » englobe généralement deux notions :

• La dématérialisation des procédures publiques et la numérisation intégrale de l'accès aux services publics de l'Etat et des administrations locales.

• L'utilisation des Technologies de l'Information et de la Communication dans les administrations, établissements publics et collectivités locales. Il s'agit ici de la mise en place de systèmes d'information « métiers » permettant un traitement électronique des procédures publiques, indépendamment de leur mise en ligne (arrière guichets informatisés). [Ref 07].

Définition 2 :

Il s'agirait d'un moyen efficace et efficient pour fournir des services publics de meilleure qualité à savoir : réduire des délais d'attente, augmenter de la productivité, réduire des coûts, améliorer de la transparence et la responsabilité des institutions publiques. D'ailleurs, partout dans le monde, une meilleure transparence contribue énormément à la lutte contre la corruption et la fraude.

"*Implementation of eGovernment is important in making Government more responsive and costeffective*".

Le cybergouvernement consisterait en la transformation des relations internes et externes de la fonction publique par des opérations basées sur Internet, la technologie de l'information et des communications afin d'optimiser la prestation des services gouvernementaux et l'art de gouverner **[Ref 01].**

Définition 3 :

Le gouvernement électronique (eGouvernement) consiste à développer la prestation de services par les pouvoir publics en utilisant au maximum les possibilités offertes par les nouvelles technologies de l'information et de la communication (TIC). Cela implique de repenser en profondeur l'interaction entre les pouvoirs publics et les citoyens.

Un certain nombre de principes sont essentiels das l'élaboration de l'eGouvernement :

- Répondre aux demandes des citoyens et des entreprises.
- Partager et échanger les données entres les services publics.
- Créer un portail d'accès commun aux différents services. **[Ref 01].**

Définition 4 :

Bien qu'électronique le gouvernement peut être différemment exprimé comme : e-gouvernement, gouvernement numérique, gouvernement en ligne et gouvernement de cyber, ils sont tous employés comme concept semblable.

En d'autres termes, le terme « électronique » dans l'e-Gouvernement représente la perspective technique de la technologie numérique, et le terme gouvernement peut être regardé comme système sociotechnique représentant l'aspect social du travail de gouvernement.

« Le e-Gouvernement est l'adoption par le service public des nouvelles technologies de l'information et de la communication (NTIC) dans son rapport avec le souverain et dans sa relation avec les administrés, les collaborateurs et les partenaires ». **[Ref 07].**

V. Volets de l'eGouvernement :

Le e-Gouvernement se décline en deux volets :

- eDemocratie

« eDemocratie est l'institution d'un dispositif destiné à permettre au souverain de participer à l'élaboration et à la détermination des lois au moyen des nouvelles technologies ».

- eAdministration

« eAdministration est l'utilisation par l'Etat des nouvelles technologies pour assurer la marche régulière des services publics tant pour son fonctionnement interne qu'en faveur des usagers » :

✓ Relations externes à l'administration :

Administration → Administrés

Administration → Partenaire

✓ Relations internes à l'administration :

Administration → Administration

Même si la définition du gouvernement en ligne évolue constamment, deux idées émergent de toutes ces définitions :

- L'adoption et l'utilisation des technologies de l'information et de la communication.
- La transformation des relations d'un gouvernement avec l'ensemble des parties prenantes internes et externes: citoyens (particuliers et entreprises), les autres gouvernements, les fournisseurs, les fonctionnaires, les communautés et les régions, les associations et autres acteurs. **[Ref 07]**

VI. Sécurité dans l'eGouvernement :

Il est nécessaire, pour renfoncer la confiance des citoyens et des entreprises dans l'eGouvernement, d'intervenir simultanément et d'une manière suffisante dans les domaines suivants : la sécurité, la protection de la vie privée et bonnes pratiques en matière d'administration électronique.

Il est en effet primordial que les autorités publiques aient une vision globale, à moyen ou long terme, de la politique à mener dans ces matières.

VI.1 Sécurité :

Les aspects juridiques et techniques sont indissociables pour assurer la sécurité. Il est souhaitable que certains processus techniques fassent l'objet de dispositions législatives. Par ailleurs, la sécurité ne peut être efficacement garantie sans l'élaboration de procédures de contrôle des dispositifs mis en place.

Trois points peuvent être distingués : au niveau du front office, la sécurité contre les attaques extérieures et la sécurité dans les échanges entre l'administration et le citoyen ou l'entreprise. Au niveau du back office, nous avons la sécurité dans les échanges entre les administrations.

S'agissant de la sécurité vis-à-vis de l'extérieur, des efforts constants doivent être consentis pour garantir un niveau élevé de sécurité, que ce soit en termes techniques ou organisationnels. A cet égard, les pouvoirs publics peuvent s'inspirer des méthodes employées par certaines entreprises et qui ont fait leurs preuves : des critères doivent être satisfaits dans certains échanges entre l'administration et les citoyens ou les entreprises, telles l'identification de l'émetteur, l'intégrité du message et la confidentialité de la communication **[Ref01]**.

Des techniques existent, parfois réglementées par les autorités publiques, telle que la signature électronique qui permet de rencontrer ces critères, dès lors que diverses conditions sont remplies. Il faut toute fois se garder de l'imposer systématiquement dans tous les échanges.

VI.2 Protection de la vie privée :

La protection de la vie privée constitue un principe fondamental de notre droit, les données à caractère personnel font par ailleurs l'objet d'une réglementation stricte.

Les mesures destinées à protéger la vie privée doivent exister, mais elles ne peuvent pour autant rendre les procédures inutilement compliquées. Une telle conséquence irait à l'encontre des objectifs poursuivis par l'eGouvernement (et notamment la simplification administrative).Les mesures doivent être proportionnelles à l'objectif poursuivi.
Les problèmes en matière de vie privée concernent notamment l'échange des données entre les administrations, l'institution d'un point d'entrée unique ou encore l'attribution d'un identifiant unique. **[Ref 01].**

VII. Conclusion :

Le e-gouvernement est encore dans une phase de décollage. Il ambitionne d'améliorer les relations quotidiennes avec le citoyen et de réduire les coûts engendrés par l'administration.

On ne peut que s'impliquer en faveur d'une telle initiative. Le e-gouvernement propose un modèle d'Internet citoyen qui ne peut que profiter à la démocratie.

Le e-gouvernement est un phénomène international. L'Europe commence à multiplier les initiatives en sa faveur.

La grande ambition est de remettre le citoyen au coeur des affaires de la cité. Pour cela deux moyens sont mis en oeuvre : la diffusion de l'information et la démocratie participative.

Ce dernier point n'est pas au coeur des projets de e-gouvernement, en revanche la problématique du partage de l'information est importante et ouvre des perspectives intéressantes concernant les domaines de la gestion de contenu et des applications web collaboratives.

Chapitre III. Outils de sécurité

Partie A : Mesures de sécurités

I. Historique :

Dans ce chapitre on va tente de présenter quelques outils de la sécurité de l'information D'une manière générale et de se focaliser sur les techniques « algorithmes » implémentés au cours de notre travail en occurrence « **RSA, MD5...** »
Les outils assurent la sécurité de l'information son nombreux et font appel à la « Cryptographie » :

Le mot cryptographie provient du Greg kryptus (caché) et graphein (écrire).La cryptographie ou écriture secrète a été utilisée pour protéger les secrets de l'humanité dès les débuts de l'écriture.

Historiquement, la cryptologie fut employée par quatre catégories de personnes : les militaires, les corps diplomatiques, les journalistes et les amoureux. Si les amoureux eurent le plus souvent recours à des chiffres qui les protégeaient « de leur petite soeur », les militaires jouèrent un rôle prépondérant dans l'évolution et le perfectionnement de la cryptologie.

A l'origine, le secret des communications reposait sur l'utilisation de diverses formes de **sténographies**, un procédé visant à dissimuler l'existence du message, plutôt qu'à masquer le sens du message lui même. Bien que la sténographie fût l'inspiratrice de ruses incroyablement ingénieuses (en Chine on écrivait les messages sur une soie très fine glissée dans une petite boule recouverte de cire, avalée ensuite par le messager ; en Italie on utilisait une encre absorbée par la coquille d'un oeuf dur, ...), ce procédé se révélait d'un grande faiblesse lorsque le message était découvert. [**Leg 07**].c'est pourquoi il fallut trouver une technique destinée à préserver le secret du message, notamment dans le cas où celui-ci était intercepté : la **cryptologie** vit ainsi le jour.

I.1. Chiffrement de César

L'un des plus célèbre ancien pratiquant de la cryptographie fut Jules César. Il a conçu un code, connu sous le nom de chiffrement de César, qui se fonde sur une rotation fixe des lettres de l 'alphabet.

A chaque lettre de l'alphabet, on fait correspondre une autre lettre, correspondant au décalage d'un nombre fixe (la valeur de rotation).Par exemple, avec une rotation de 1, A correspond à B,

B à C, ..., Y à Z et Z à A (il y a une rotation circulaire des lettres de fin et de début).

Avec une rotation de 13, A correspond à N, B à O, ..., Y à L et Z à M.

Key=1 <=> Rotation d'une position

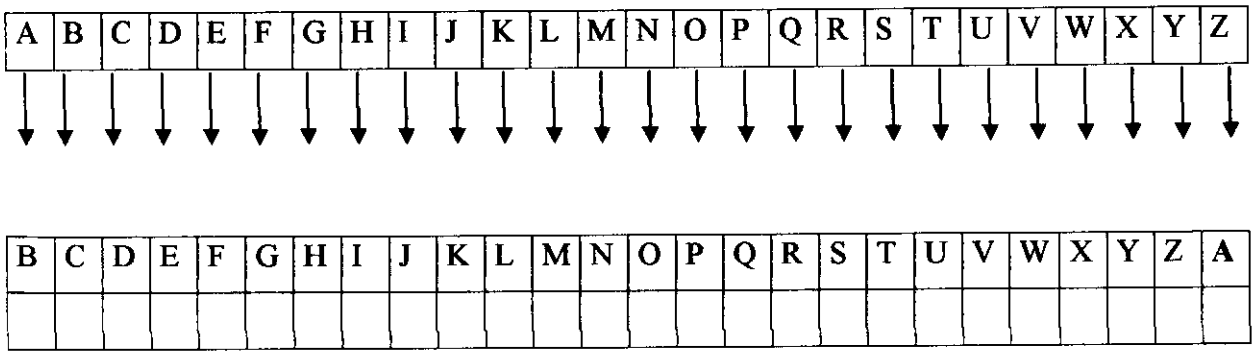


Figure III.1 : Chiffrement de césar

I .2 . Première guerre mondiale

La première Guerre Mondiale fut une suite de véritables batailles sur le plan technique. Dans les deux camps, négligences et atermoiements caractérisèrent la période du début. Bientôt cependant, une activité fébrile marque l'intérêt de ceux qui participèrent à la guerre pour la cryptographie. Cette période est féconde et d'une influence décisive. Elle est à l'origine de la création, dans tous les pays, de services organisés de chiffre et de décryptement.

I.3. La seconde guerre mondiale.

Durant la Seconde Guerre Mondiale, la cryptographie connût un développement considérable notamment avec l'utilisation de la machine ENIGMA.

Dans les faits, la machine pouvait être utilisée comme une machine à écrire standard et cela même en plein milieu de l'encodage d'un texte. ENIGMA A n'a pas connue un très grand succès malgré la publicité faite à cette époque. Par la suite, trois autres modèles apparurent, soit les modèles (B, C et D). Le modèle B est similaire au modèle A à l'exception des rotors qui ont maintenant 26 contacts au lieu de 28 pour le modèle A. Les modèles C et D étaient portables et crypto graphiquement différents des modèles précédents. Ces derniers fonctionnent selon des principes identiques à ceux des machines de Hebern, mais avec néanmoins quelques différences importantes.

II .Les méthodes de cryptographie actuelle

II.1. Cryptographie à clé publique

Une des avancées les plus importantes de la cryptographie du Xxème siècle concerne le développement du cryptage à clé publique.

Les algorithmes à clé publique ou asymétrique, se fondent sur l'utilisation de clé de cryptage (publique) et de décryptage (privée). Les algorithmes à clé publique exigent que le calcul de la clé privée à partir de la clé publique soit quasi impossible. Cette exigence permet de rendre publique la clé de cryptage sans affecter la sécurité de l'algorithme Whitfield Diffie et Martin Hellman furent les premiers à publier l'idée d'une cryptographie à clé publique en 1976. Mais en 1997, le CESG (Communication Electronics Security Group) du Royaume-Uni rendit publiques des informations au par avants secrètes indiquant que la cryptographie à clé publique fut inventée par James Ellis du CESG en 1970.

II.2. Algorithme RSA (Rivest, Shamir, Adleman)

Si la cryptographie à clé publique fut introduite par Diffie et Hellman ; l'algorithme le plus célèbre fut développé par Ronald Rivest, Adi Shamir et Len Adleman en 1977. Il fut

nommé RSA, d'après les initiales de ses inventeurs. Son intérêt réside dans le fait qu'il peut être utilisé pour le cryptage et les signatures numériques.

La sécurité fournit par l'algorithme RSA dépend de la difficulté de la factorisation des grands nombres (Une analyse cryptographique sans factorisation de RSA devrait être impossible, mais personne ne l'a encore trouvée.)**[Ref 02].**

Voici un récapitulatif du fonctionnement de l'algorithme RSA :

1. Deux nombres premiers (de cent chiffres ou plus), p et q sont générés, avec n = pq.
2. Une clé publique entière ''e'' est sélectionnée, de manière qu'elle soit première avec (p-1) (q-1). Deux entiers sont premiers entre eux s'ils n'ont pas d'autre facteur commun que 1.
3. La clé privée d est calculée pour que la formule {ed mod (p-1) (q-1)} soit égale à 1.
4. Le cryptage est effectué sur des nombres du texte en clair m qui sont inférieurs à n en calculant { m^e mod n}.
5. Le décryptage s'effectue sur le texte crypté c en calculant {c^d mod n}.

p et q sont des nombres premiers secrets

n = pq

e est premier (p-1)(q-1)

d est secret et vaut e^{-1} mod ((p-1)(q-1))

m est le texte en clair

c est le texte crypté

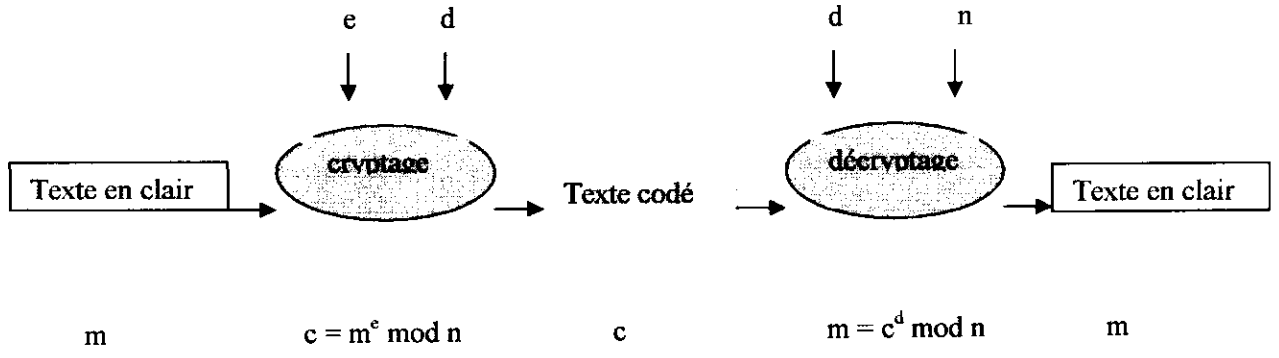


Figure III.2 : Principe de RSA

II.3. Utilisation de RSA

Pour utiliser RSA, une personne ou une organisation X rend publique sa clé publique (e et n). Quand une autre personne Y veut envoyer un message crypter à X, elle décompose le massage en blocs (m_1, m_2...) inférieurs à n et crypte chacun en utilisant la formule { m_{ie} mod n }. Les tailles de blocs de messages sont habituellement des multiples de 64 bits. Un remplissage est utilisé sur les blocs partiels.

Quand X reçoit du message crypté (c_1, c_2...), il les décrypte en calculant pour chacun { c_{ie} mod n }. Comme il est le seul à connaître la valeur de d, personne d'autre ne peut déchiffrer le message envoyé par Y. Si X veut envoyer un message à Y, il doit obtenir sa clé publique et l'utiliser pour coder ses messages.

L'avantage du cryptage à clé publique sur le cryptage à clé secrète est que X et Y n'ont pas à partager de clé. Ils peuvent rendre publique leurs clés publiques et conserver pour eux seules leurs clés privées.

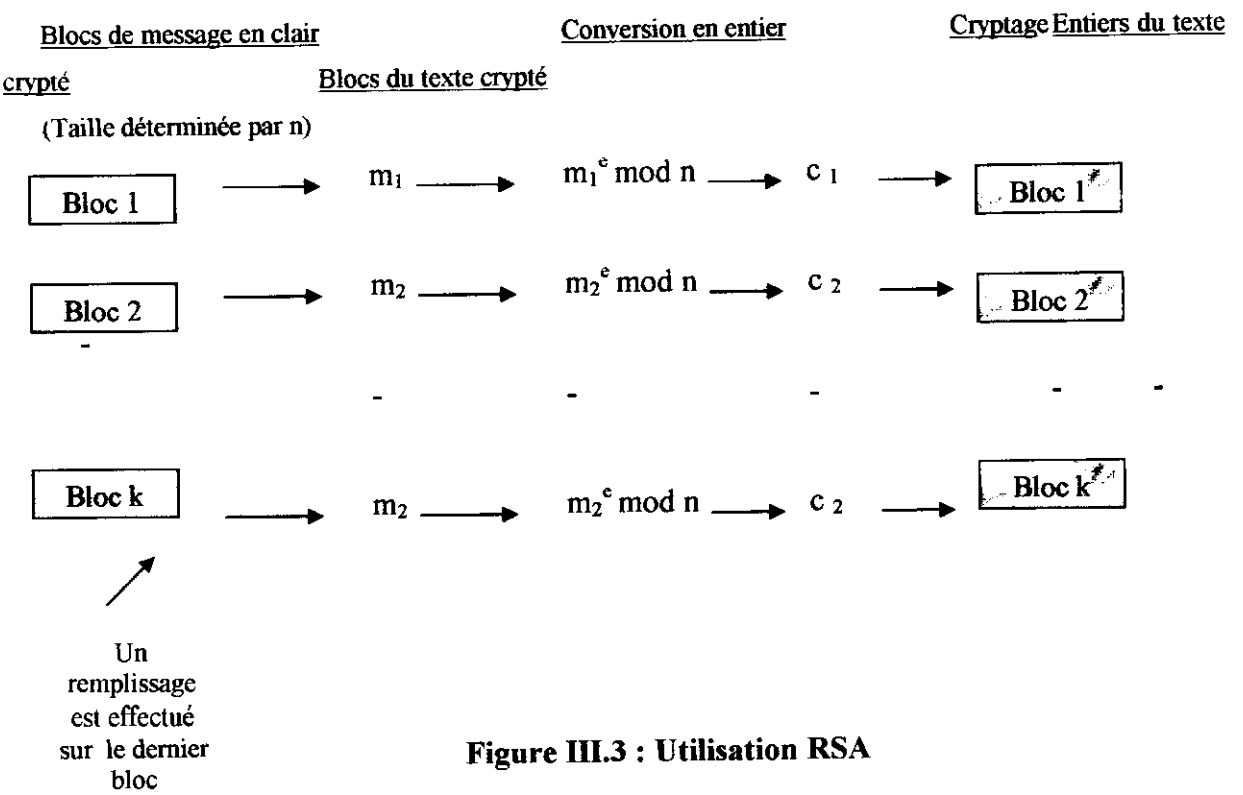


Figure III.3 : Utilisation RSA

III. Digest de message

Les techniciens de cryptographie ne se limitent pas à la préservation du secret des messages, des fichiers ou d'autres objets. Ils ont aussi les fonctions suivantes :

Intégrité. Pour détecter si un message ou un objet a été modifié ou remplacé.

Authentification. Pour vérifier qu'un message ou un autre objet provient bien d'une personne ou d'une organisation d'identité donnée.

Non-Répudation. Pour empêcher que quelqu'un nie avoir envoyer un message ou effectué une opération sur un objet.

Ces tâches sont accomplies en utilisant des digests de message et des signatures numériques.

Un digest de message est un type de fonction particulier, dite unidirectionnelle (ou hachage). Une fonction unidirectionnelle est facile à calculer, mais difficile à inverser. Une bonne analogie est une broyeuse à papier : il est simple d'y mettre un document à broyer, mais il est très difficile de reconstituer ensuite le document d'origine.

Une forme particulière de fonction unidirectionnelle calcule des valeurs, dites digests de messages ou valeurs de hachage, utilisées comme empreintes digitales pour le message. Les fonctions du digest de message efficace ont les propriétés suivantes :

- Etant donné une valeur de digest de message, il est matériellement impossible de calculer un message qui produira cette valeur dans la fonction de digest de message.
- Il est matériellement impossible de trouver deux messages produisant la même valeur de digest de message.

Une fonction de digest de message n'a rien de secret. Elle est disponible publiquement et ne possède pas de clé. Les algorithmes MD5 (Message Digest 5) et SHA-1 (Secure Hash Algorythm 1) en sont les exemples les plus connus.

III.1. MD5

MD5 est une fonction de digest de message développée par Ron Rivest. Les fonctions MD 1 à MD 4 existent mais MD 5 est la plus courante. C'est en fait une extension de MD 4. Elle est dans le domaine public et est documentée dans la RFC 1321. MD 5 fonctionne sur des longueurs de message arbitraire et génère un digest de message de 128 bits (64 octets).

III.2. Algorithme de signature numérique.

DSA (Digital Signature Algorythm) du NIST (National Institute of Standarts and Technology) est un exemple d'algorithme de signature numérique. Comme l'algorithme à clé publique ElGamal, il se fonde sur la difficulté de calcul des logarithmes discrets pour sa sécurité.

IV. Signatures numériques

Les digests de message sont très pratiques pour indiquer qu'un message ou un autre objet a été, accidentellement ou délibérément, altéré, mais ils ne peuvent pas dire s'ils ont effectivement été créés par un individu ou une organisation donnée. C'est la qu'interviennent les signatures numériques.

Une signature numérique est une valeur calculée à partir d'une séquence d'octets en utilisant une clé secrète. Elle indique que la personne possédant cette clé secrète a vérifie que le contenu du message est correct et authentique. Les signatures numériques utilisent souvent les algorithmes de cryptage à clé publique avec une légère modification : une clé privée est utilisée pour le cryptage et une clé publique pour le décryptage. Cette proche est souvent mise en œuvre de la manière suivante :

a) Génération de la signature :

1. Un digest de message est calculé.
2. Le digest de message est crypté en utilisant la clé privée d'une paire de clés publiques/privée, pour générer la signature numérique du message.

b) La vérification de la signature

1. La signature est décryptée en utilisant la clé publique de la paire de clés publique/privée, pour générer la valeur de digest de message.
2. La valeur de digest de message est comparée avec le digest de message calculer à partir du message d'origine.
3. Si les deux digests correspondent, la signature est authentique. Dans le cas contraire la signature, ou le message ont été falsifiés.

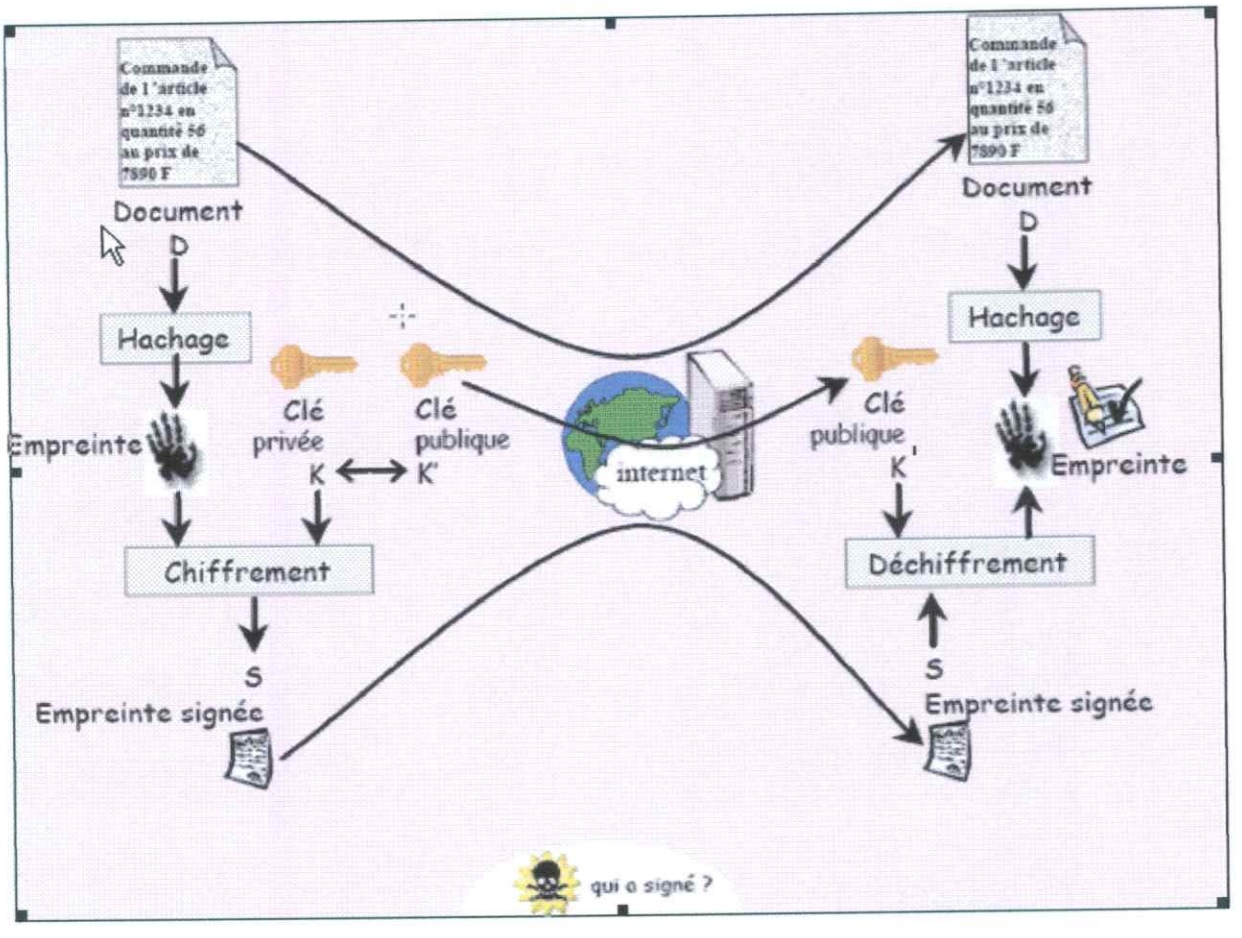


Figure III.4 : Principe de la signature numérique [Lss 00].

V. Certificats numériques

Les signatures numériques sont un excellent moyen d'authentifier l'émetteur, personne ou organisation, d'un message ou d'un objet .Mais il faut un moyen pour déterminer si la clé publique utilisée par le signataire est bien la sienne. C'est l'objet des certificats.

Les certificats numériques sont signés par une autorité de certification (ou CA, Certification Authority) qui certifie l'authenticité de la clé publique d'une entité. Une autorité de certification est une entité approuvée pour vérifier que les autres entités sont bien ce qu'elles prétendent être, et qu'elles utilisent une clé publique donnée avec un algorithme de cryptage à clé publique particulier. Pour obtenir un certificat d'une autorité de certification, il faut en général fournir des documents prouvant votre identité ou celle de votre organisation.

Par exemple, le processus de certification empêche des personnes non autorisées à créer une entreprise sur le web en utilisant l'identité Microsoft ou Bank of America. **[Ref02].**

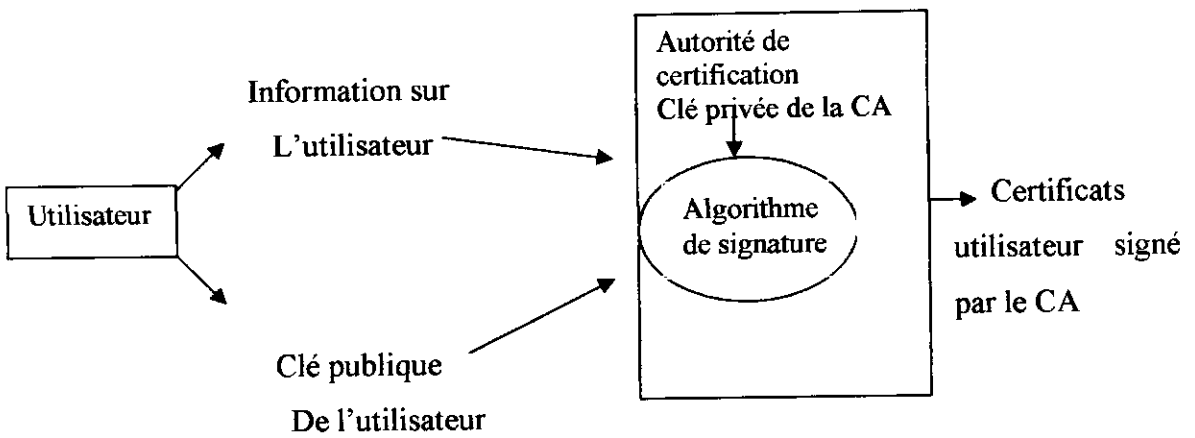


Figure III.5 : Attribution de certificat numérique

Dans un environnement de réseau, plusieurs niveaux d'autorités de certifications peuvent être requis. Par exemple, une autorité supérieur, comme VeriSign, Inc., US Post Office ou la NSA (National Security Agency) peuvent fournir des certificats à des autorités de certification de second niveau **[Bru04]**. Ces dernières peuvent ensuite proposer des certificats aux autres organisations. Une entreprise peut aussi agir comme autorité de certification de ses employés. Une structure hiérarchique de certification en résulte. La certification d'une entité de cette hiérarchie dépend de la certification des entités se trouvant aux niveaux supérieurs. On parle alors de chaîne de certification.

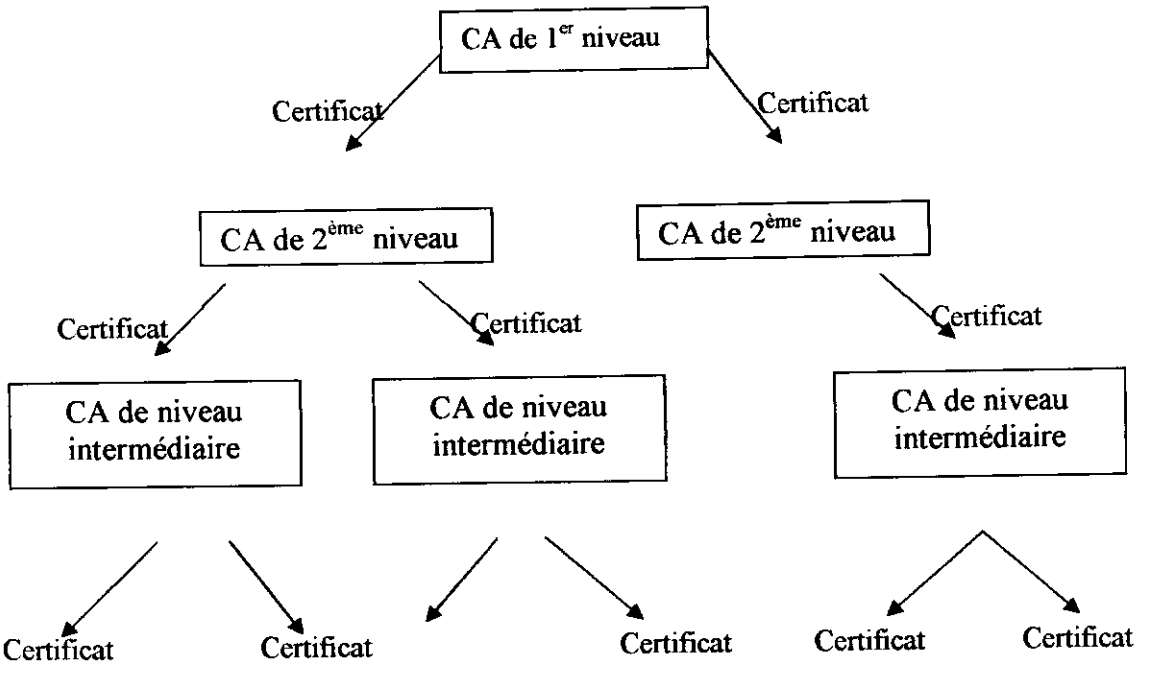


Figure III.6 : Hiérarchie de Certification

Le format de certificats numériques X.509 de l'ISO (Organisation Internationale de Normalisation) est courant. Il identifie un format et un contenu particulier pour les certificats numériques, ce format qui a été popularisé par le protocole de sécurité SSL (Secure Sockets Layer) de Netscape, le format de fichier JAR (Java Archive), PEM (Privacy Enhanced Mail), et d'autres standards émergents de sécurité du réseau Internet. Les certificats X.509 contiennent les informations suivantes : **[Bru 04]**

- La version de X.509 utilisée avec le certificat (1, 2 ou 3) ;
- Le nom de l'entité et sa clé publique ;
- Une plage de dates de validité du certificat ;
- Un numéro de série attribué par l'autorité de certification ;
- Le nom de l'autorité de certification ;
- Une signature numérique créée par l'autorité de certification.

La version actuelle de X.509 est la troisième, mais les certificats de version 1 sont toujours utilisés. La version 3 donne la possibilité d'ajouter des extensions personnalisées aux certificats, comme des adresses de courrier électronique ou IP.

On trouve l'usage des certificats numériques dans diverses applications par exemple, dans Java, le premier usage des certificats numériques est la gestion de l'authentification du code. Les développeurs de code java peuvent signer numériquement leur code en utilisant leurs clés privées. Les utilisateurs du code vérifient la signature du développeur à l'aide de sa clé publique. Les développeurs utilisent les certificats numériques comme une méthode sécurisée pour informer les utilisateurs sur leur clé publique.

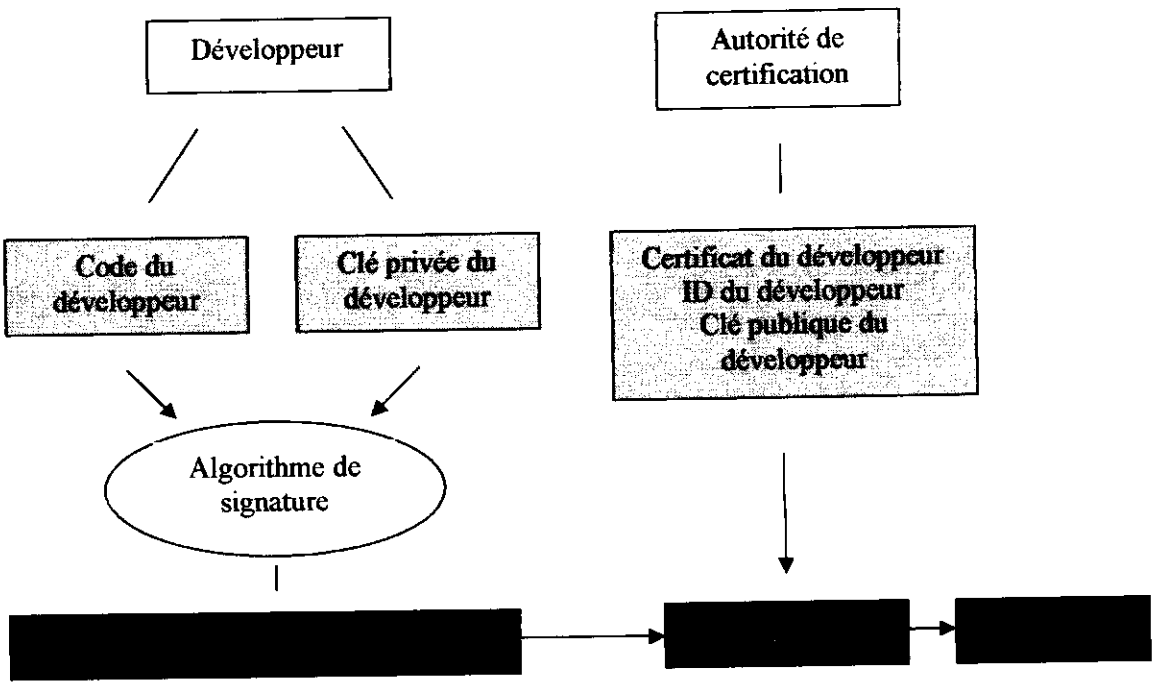


Figure III.7 : Vérification de code de développeur

VI. Gestion des clés

La gestion des clés est un des aspects les plus difficiles de la configuration d'un système cryptographique de sécurité. Pour qu'un tel système fonctionne et soit sécurisé, chacun des utilisateurs doit disposer d'un ensemble de clés secrètes (dans un système à clés secrètes) ou de paire de clés publiques/privés (dans un système à clés publiques). Cela implique de générer les clés et de les distribuer de manière sécurisée aux utilisateurs ou d'offrir à l'utilisateur les moyens de les générer. Il doit aussi pouvoir enregistrer et gérer ses clés publiques et privées de manière sure .dans les systèmes à clés publiques, la gestion des clés comprend la capacité à vérifier et à gérer les clés publiques des autres utilisateurs qui sont signées sous formes de certificats numériques. **[jaw 01]**

VI.1. Les nombres aléatoires sécurisés et la génération des clés

Les nombres aléatoires sont importants en cryptographie. Ils sont habituellement utilisés pour rendre le résultat d'opération comme la génération des clés difficilement prédictibles. Malheureusement, les ordinateurs ne produisent généralement pas de véritables nombres aléatoires. Les algorithmes de génération de nombres aléatoires utilisés (PRNG Pseudo Random Number Generator) produisent en fait des flux de nombres pseudo aléatoires. Certains sont plus aléatoires (et donc moins prévisible) que d'autres. On peut obtenir un meilleur niveau aléatoire en utilisant une semence sélectionnée au hasard et en utilisant un algorithme cryptographique solide, c'est à dire dans lequel il est très difficile par calcul de déterminer des valeurs produites par l'algorithme sans connaître la valeur de la semence.

VI.2. Echange de clés

L'échange de clés, l'accord sur les clés ou la distribution, est le processus dans lequel plusieurs parties communicantes arrivent à s'accorder sur les clés à utiliser pour entreprendre des communications cryptées (ou signées).

Dans les systèmes à clés publiques, l'échange de clé est fortement simplifié. Chaque partie communicante publie sa clé publique ou l'envoie à l'autre. Les clés publiques sont habituellement distribuées en utilisant des certificats numériques, utilisés par le destinataire pour identifier la clé publique reçue ; toutes les communications avec cette partie seront alors cryptées avec cette clé.

L'échange de clés dans les systèmes à clés secrètes est beaucoup plus compliqué. Habituellement, une seule clé secrète est utilisée entre deux parties communicantes, sur une seule session ou sur une période limitée (un jour, une semaine, un mois...). Dans une organisation de 1000 utilisateurs qui en moyenne communiquent avec une vingtaine de personnes extérieures différentes ; il faut environ 1020 clés différentes pour gérer les communications secrètes. De plus ces clés doivent être changées à la fin de la période utile (session, jour, semaine, mois...). **[Bru 04]**

Le problème de la distribution des clés peut être résolue en partageant la même clé pour des groupes d'utilisateurs et en étendant la durée de validité. Cependant, ces deux mesures affaiblissent la sécurité du système. De plus, aucune ne résout le problème fondamental de la distribution matérielle des clés. Dans certaines organisations, les clés sont distribuées par courrier et chargées manuellement dans les unités cryptographiques ou par des appareils de

saisie électronique des clés. Cependant, la meilleure approche dans la plupart des cas consiste à utiliser un protocole d'échange des clés.

Un protocole d'échange de clés utilise un algorithme à clé publique pour autoriser plusieurs parties communicantes à obtenir la même clé secrète. Habituellement la clé secrète sert à activer une seule session de communication secrète (ou une partie donnée de la session). La clé est alors dite « clé de session ».

VI.3. Stockage des clés de cryptage par mot de passe

Le stockage est un autre aspect essentiel de la gestion des clés. Une fois qu'une clé secrète est générée (ou la clé privée d'une paire publique/privée), elle doit être rangée en mémoire ou stockée d'une manière ou d'une autre. Comma la plupart d'entre nous n'arrivent pas à mémoriser une clé (sans parler d'un ensemble de clés), nous avons tendance à la stocker.

Il existe plusieurs façons de stocker des clés : les noter par écrit ou les imprimer, les enregistrer dans un fichier, dans une carte à puce ou dans un dispositif inviolable de clé électronique. Si la plupart des utilisateurs ne se servent pas actuellement de cartes à puce ou de dispositifs électroniques, cela peut changer dans un futur pas si lointain. Pour l'instant le choix consiste à les recopier sur un support papier ou à les enregistrer sur le disque dur.

Le stockage de clés sur un support papier est peu pratique et manque de sûreté. S'il est possible d'imprimer une clé au lieu de la recopier, il faudra généralement la ressaisir. Pour des clés d'une certaine taille, c'est un inconvénient majeur. On peut utiliser un scanner pour lire une clé imprimée, mais il ne sera pas toujours disponible lorsqu'on en aura besoin. En tous cas, personne ne souhaite se promener avec un paquet de papiers listant ses clés : les copies papiers peuvent être facilement dupliquées ou volées.

Dans la plupart des cas, les clés sont enregistrées sur disque dur. Elles sont alors vulnérables à une divulgation à toute personne obtenant un accès au disque ou à l'ordinateur le contenant. Si l'ordinateur est connecté à Internet, un pirate peut avoir accès à l'ordinateur et donc aux clés.

La contre-mesure évidente pour enregistrer les clés sur disque est de les crypter. Mais cette solution se mord la queue : que faire de la clé utilisée pour crypter les clés enregistrer sur l'ordinateur ? La réponse à ce dilemme consiste à utiliser un cryptage par mot de passe (ou PBE : Password Based Encryption). **[jaw 01]**

Le cryptage par mot de passe utilise un mot ou une phrase de passe pour générer une clé de cryptage. Dans beaucoup d'application, le cryptage par mot de passe utilise aussi une valeur aléatoire, dites « sel », afin d'augmenter l'effort nécessaire au décryptage du fichier. Le sel est combiné au mot de passe pour crypter le fichier.

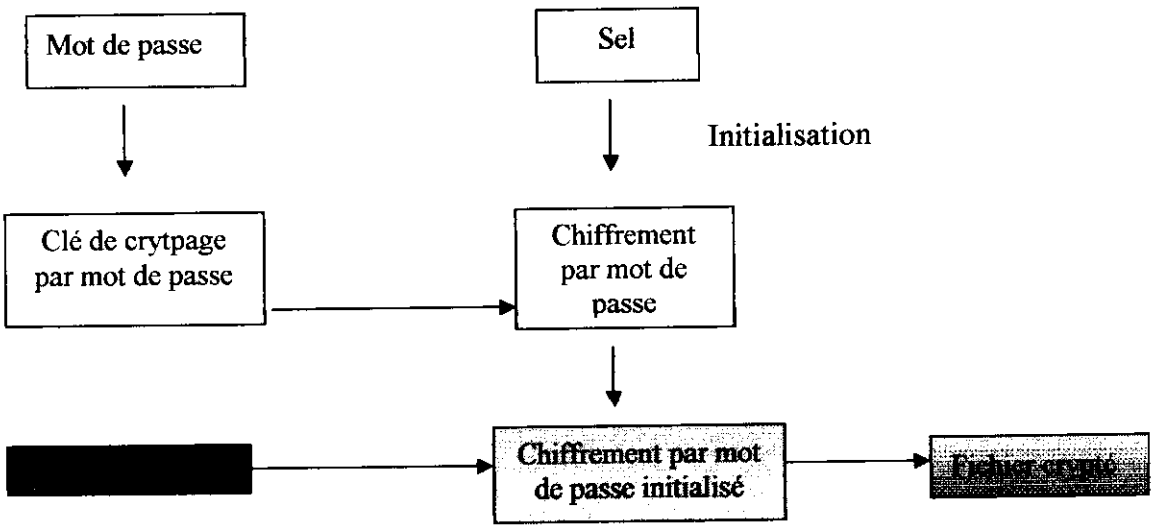
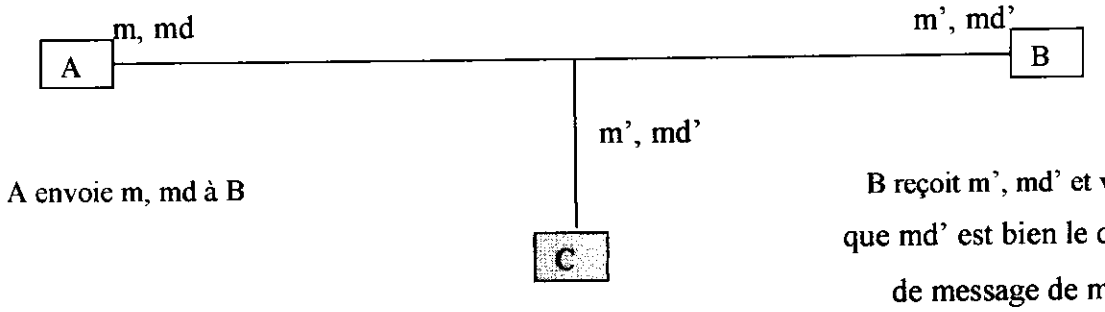


Figure III.8 : Stockage des clés de cryptage par mot de passe

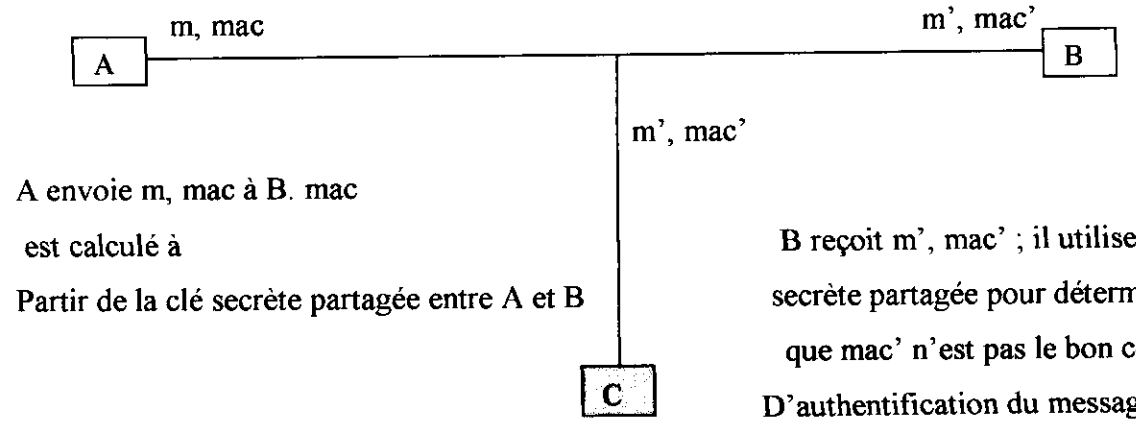
VII. Codes d'authentification de message

Les codes d'authentification de message (ou MAC Message Authentification Code) sont des digests de message calculés par une cryptographie à clé secrète. La clé sert à détecter les modifications éventuelles d'un message envoyé sur un canal de communication non sécurisé. Par exemple : A veut envoyer un message m à B. A calcul md, digest de message de m, et envoie à B les données (m et md) sur un réseau non protégé. Mais C intercepte (m et md) ; il modifie m (qui devient m') et crée md' (le digest de message m'). C envoie envoie alors m' et md' à B. Quand B vérifie md', il constate que c'est bien le digest de message de m'.



C intercepte m et le remplace par m'.
Mais comme il ne connaît pas lé clé de
A et B il ne peut pas créer le bon code mac'.

Comme les codes d'authentification de message utilisent une clé pour calculer le digest de message, les messages qu'il accompagne ne peuvent pas être interceptés, modifiés et retransmis sans que cela se détecte. Par exemple, si C modifie le message de A, il doit posséder la clé secrète de A pour pouvoir calculer le code MAC di message modifié



C intercepte m est le remplace par m'
Mais comme il ne connaît pas la clé de
A et B, il ne peut pas créer le bon code mac'

Dans l'exemple précédent, un protocole sécurisé de distribution/échange de clé est nécessaire pour distribuer la clé partagée entre A et B.

Un code d'authentification de message utilisant une fonction de digest de message ou de hachage est désigné par HMAC. Par exemple, un code HMAC peut être généré par une utilisation avec MD5 ou SHA-1. Un code HMAC utilise une clé secrète avec l'algorithme de digest de message.

VIII. Sécurité de transmission en réseau :

SSL est un protocole de la couche de communication du modèle OSI créée par Netscape Communication Corporation, SSL s'appuie sur la pile de protocole TCP/IP. SSL procure des services sécurisés sur TCP/IP, comme la confidentialité par le cryptage des données, l'intégrité par un algorithme MAC , et une authentification optionnelle des sockets client et serveur. SSL peut être considéré comme une couche intermédiaire fonctionnant entre les couches de communication TCP/IP et les protocoles de couche de plus haut niveau comme http et IIOP, pour fournir une solution de communication sécurisée. **[Chd 03].**

Netscape a introduit SSL v2 dans la version 1 de son navigator. SSL v3 est le standard SSL le plus généralement utilisé. Le protocole TLS (Transport Layer Security) de la couche « data link » élaboré par l'IETF (Internet Engineering Task Force) étend SSL v3 en améliorant les aspects d'authentification de l'algorithme SSL. Le protocole WTLS (Wireless Transport Layer Security) est, comme son nom l'indique, une version de TLS utilisée dans les communications sans fils.

IX. Conclusion :

Les outils de sécurité de se déploiement donc en fonction des besoins exprimés et leurs utilisation.

Dans ce cas dans l'e-gouvernance , il s'agit d'authentifier les échanges des documents et leur sécurisation .c'est ce que nous présentons dans le chapitre suivant.

Partie B : Etat de l'art

I. Gouvernance dans le monde :

Les E-services (E-government, E-commerce, E-banking,...) sont des indicateurs clés pour le développement économique d'un pays c'est pour cette raison qu'il a fallu songer a appliquer une politique de sécurité pour assurer l'authentification ,l'intégrité,la non répudiation et la confidentialité, et cela en employant divers moyens, entre autres l'authentification électronique , la signature numérique, **Carte d'identité électronique** pour l'accès à tous les E-services et **passeport électronique** (USA et Asie 2005, Europe 2006) basés sur la **carte à puce** intégrant des éléments biométriques :**[ghe 03]**

Parmi les Pays disposant d'une loi sur la signature numérique en Afrique on peut citer : Ile Maurice, Tunisie, Cap Vert, Afrique du Sud, Egypte.

Parmi les Pays ayant un projet de loi sur la signature numérique en Afrique on peut citer : Algérie, Burkina Faso, Cameroun, Maroc, Sénégal,...

Il y'a Une nécessité de crypter les documents échangé, et donc de clés de cryptage même au développement de divers ICP (Infrastructure a clé Public) en Afrique parmi elles :

- Afrique du Sud (secteur privé : Thawte, SACA,...),
- Tunisie (ANCE)
- Egypte (ITIDA)
- Ile Maurice (ICT authority CCA)

I.1. Ile Maurice

- Loi sur les transactions électronique, Août 2000 (ETA) et la loi ICT 2001
- ICT authority est le régulateur des Autorité de Certification (AC) : Agrément pour les CA et reconnaissance des autorités étrangères
- Etat actuel: Choix du modèle PKI le plus adapté pour le contexte national

I.2. Afrique du Sud

- Loi sur les Communications et les Transactions électroniques, Août 2002
- Draft de la régulation d'accréditations des AC (Juillet. 2004) : choix d'un contrôleur de CA (CCA). **[Ref 03].**
- ICP du secteur privé (Thawte, SACA, Namitech, etc.

I.3. Egypte

- Loi sur la signature numérique, Avril 2004
- Autorité de régulation de signature électronique : Information Technology Industry Development Agency (ITIDA)
- ITIDA est la AC racine pour l'Egypte, le contrôleur des AC, reconnaissance mutuelle avec les autorités étrangères
- ICP en cours

I.4. Tunisie

- National commission for electronic commerce and electronic exchanges-1997
- Ecommerce and Electronic Exchanges Law, Aug. 2000
- National Digital Certification Agency (root CA) Jan. 2001
- Applications: E-commerce, E-government, E-banking, etc.

Le rôle de L'ANCE :(Figure III.2)

a. Autorité de certification racine (plus haut niveau de confiance)
b. Génération, renouvellement et révocation des certificats électroniques
c. Octroi des licences pour Fournisseurs de Services de Certification Electronique (FSCEs)
d. Etablissement des accords de reconnaissance mutuelle avec les autorités étrangères.
e. Homologation des produits de cryptage
f. Veille technologique (standards et technologies)

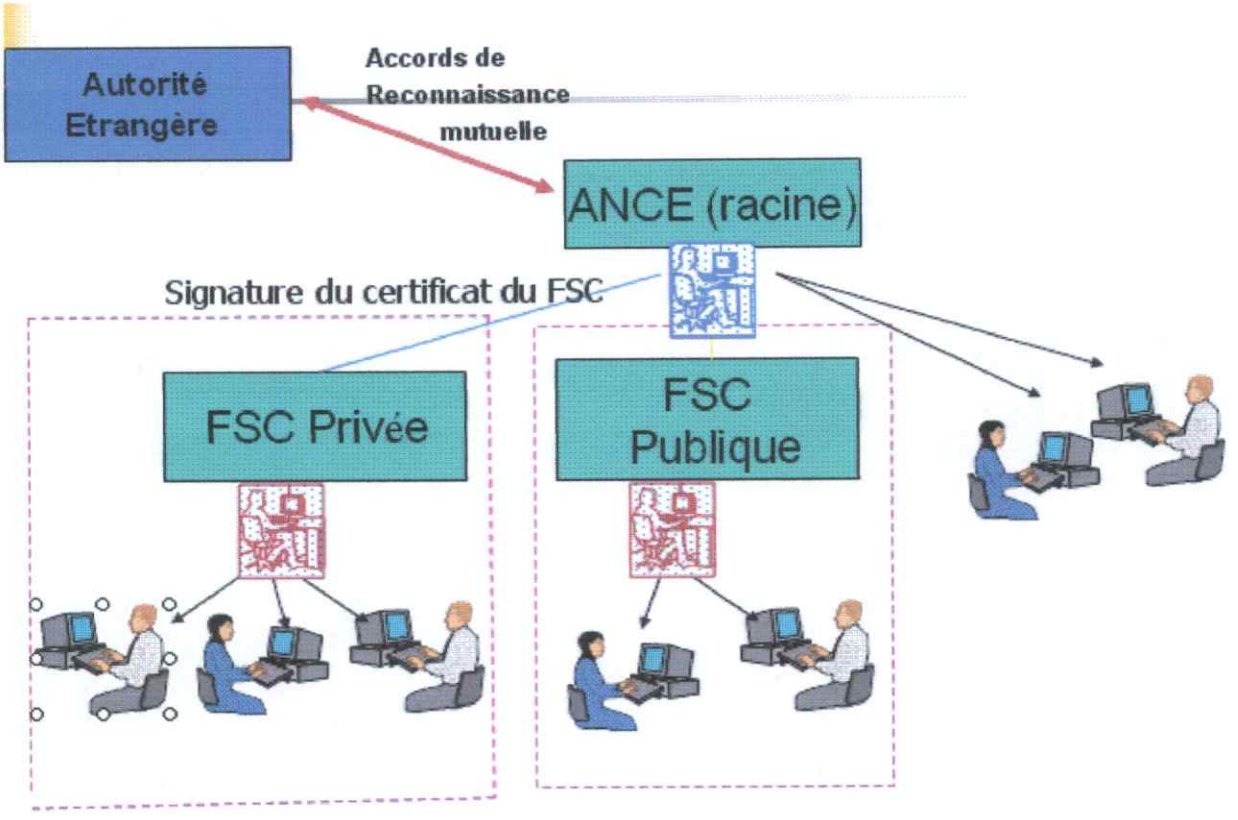


Figure III. 1 : rôle de ANCE. [Ref 04].

II. Conclusion :

Pour développer l'e-gouvernance, il faut, en plus des lois la définissant, créer une infrastructure permettant de déployer le cryptage .cette infrastructure inclut tous les outils de sécurité dont nous faisons une présentation au chapitre suivant.

Chapitre IV. Modélisation et Conception

Plusieurs méthodes sont consacrées pour la conduite d'un tel projet, pour notre part nous avons privilégié une approche objet qui représentent les avantages de la stabilité de la modélisation par rapport aux entités du monde réel, de la construction itérative, de la possibilité de réutiliser des élément d'un développement à un autre et de la simplicité du modèle .nous avons opter pour la modélisation avec UML, avec un processus UP itératif et incrémentale.

Nous présenterons les besoins identifiés de manière formelle avec le diagramme des cas d'utilisation, puis nous donnerons les différents scénarios associés à chaque cas d'utilisation, les diagrammes de séquences, pour montrer la chronologie des interactions et les informations échangées, ainsi que les diagrammes d'activités et les diagrammes d'états-transitions pour mieux présenter notre système .Enfin nous arriverons au diagramme de classe finale qui donne la structure statique de notre système.

I. UML (Unified Modeling Language)

I.1. Introduction

UML (Unified Modeling Language) 1997 est le résultat de la fusion de 3 méthodes : OMT (Object Modeling Technique) de James RUMBAUGH, OOD (Object Oriented Design) de Grady BOOCH et OOSE (Object Oriented Software Engineering) de Ivar JACOBSON. UML permet de représenter un système selon différentes vues complémentaires appelées diagrammes. Un diagramme UML est une représentation graphique, qui s'intéresse à un aspect précis du modèle; c'est une perspective du modèle. Chaque type de diagramme UML possède une structure et véhicule une sémantique précise. Par conséquent, l'ensemble des différents types de diagrammes UML offrent une vue complète des aspects statiques et dynamiques d'un système.

UML est un langage qui permet de représenter un système à travers des modèles, mais il ne définit pas le processus d'élaboration de ces modèles. Qualifier UML de "méthode objet" n'est donc pas approprié.

I.2. Les diagrammes d'UML :

Les diagrammes donnent à l'utilisateur un moyen de visualiser et de manipuler des éléments de modélisation. Ils peuvent montrer tout ou une partie des caractéristiques des

éléments de modélisation, selon le niveau de détail utilisé dans le contexte d'un diagramme donné, c'est une représentation graphique de quelques éléments du modèle. UML définit neuf diagrammes :

I.2.1 Le diagramme de classes :

Le diagramme de classes est le point central dans un développement orienté objet. Il exprime la structure statique d'un système en terme de classes et de relations entre ces classes. Une classe est représentée par un rectangle compartimenté. Le premier compartiment contient le nom de la classe, le deuxième contient les attributs et le troisième les opérations.

I.2.2 Le diagramme d'objets :

Le diagramme d'objets permet de mettre en évidence des liens (instance d'associations) entre les objets (instances de classes). Il utilise les mêmes concepts que le diagramme de classes, il est essentiellement utilisé pour comprendre ou pour illustrer des parties complexes du diagramme de classes.

I.2.3 Le diagramme des cas d'utilisation :

Un cas d'utilisation (use case) est utilisé pour définir le comportement d'un système ou la sémantique de toute autre entité, sans révéler la structure interne de l'entité. Chaque cas d'utilisation spécifie une séquence d'actions, y compris des variantes que l'entité réalise en interagissant avec les acteurs de l'entité. La responsabilité d'un cas d'utilisation est de spécifier un ensemble d'instances du cas d'utilisation que le système réalise et qui fournit un résultat observable par un acteur particulier.

Les cas d'utilisation décrivent sous la forme d'actions et de réactions le comportement d'un système du point de vue d'un utilisateur. Ils permettent de définir les limites du système et les relations entre le système et l'environnement.

Le terme cas d'utilisation est explicite : dans quel cas tel acteur utilise-t-il le système ? Chaque réponse à cette question est donc par définition un cas d'utilisation.

Un cas d'utilisation est constitué d'un ensemble d'interactions entre acteurs et le système. Les cas sont une modélisation du système, mais externe au système. Le formalisme des cas d'utilisation utilisé, est basé sur le langage naturel facilitant l'expression des besoins pour les utilisateurs.

I.2.4 Le diagramme de séquence :

Un diagramme de séquence montre une interaction présentée en séquence dans le temps, il montre les objets qui participent à l'interaction et les messages qu'ils échangent présentés en séquence dans le temps.

Un diagramme de séquence est représenté en deux dimensions. L'axe vertical correspond au temps, tandis que l'axe horizontal recense les objets qui interagissent dans la séquence.

I.2.5 Le diagramme de collaboration :

Un diagramme de collaboration montre une interaction organisée autour des objets de l'interaction et de leurs liens. A l'inverse d'un diagramme de séquence, un digramme de collaboration montre des relations parmi les rôles d'objets. En revanche, un diagramme de collaboration ne montre pas le temps dans une dimension séparée ; ainsi, la séquence des messages et les fils concurrents doivent être déterminés en utilisant les numéros de séquences.

Un diagramme de collaboration montre des interactions entre objets en insistant sur la structure spatiale statique qui permet la mise en collaboration d'un groupe d'objets.

I.2.6 Le diagramme d'états–transitions :

Il a pour objectif de représenter tous les états possibles ainsi que les événements provoquant un changement d'états. Il est associé à une classe pour laquelle on gère différents états.

I.2.7 Le diagramme d'activités :

Un diagramme activités est une variante des diagrammes d'états-transitions, organisés par rapport aux actions et principalement destinés à représenter le comportement interne d'une méthode (la réalisation d'une opération) ou d'un cas d'utilisation.

Le diagramme d'activités, dans sa globalité, est attaché à une classe ou à un cas d'utilisation.

Une activité est représentée par un rectangle arrondi, comme les états, mais plus étiré horizontalement. Les activités sont reliées entre elles par des flèches lorsqu'une activité se termine l'autre démarre.

I.2.8 Le diagramme de composants :

Les diagrammes de composants décrivent les éléments physiques et leurs relations dans l'environnement de réalisation. Les diagrammes de composants montrent les choix de réalisation.

Dans un diagramme de composants, l'accent est mis sur les modules, les processus, les sous-programmes et les sous-systèmes.

I.2.9 Le diagramme de déploiement :

Le diagramme de déploiement montre la disposition physique des différents matériels entrant dans la composition d'un système et la répartition des programmes exécutables sur ces matériels. Les différentes dispositions physiques des matériels sont reliées entre eux par des lignes qui symbolisent un support de communication.

I.3. UP (Unified Process):

Les auteurs d'UML ont défini un processus piloté par les cas d'utilisation, Centré sur l'architecture, Itératif et incrémental; c'est le processus unifié UP (*Unified Process*).

Ce processus n'est pas applicable directement, il définit des principes et une architecture, mais il doit être adapté à l'organisation et au projet visés. C'est un processus générique.

II. Les diagrammes :

II.1. Diagramme de cas d'utilisation :

Nous débutant notre étude par la recherche des acteurs du système, Un acteur représente un rôle joué par une personne ou par un objet qui interagit avec le système. Les acteurs de notre système se répartissent dans les catégories suivantes:

- *Citoyen* : toute personne qui accède au système en consultation. doit être identifié par le système.
- *Officier* : c'est une personne qui s'occupe de la mise à jour du système (Ajout, Suppression et Modification des citoyens).
- *Administrateur Local* : il est responsable de l'administration du système Local : récupération des clés, gestion des officiers (Ajout, Suppression, Modification), et création des sessions citoyen (Donner un mot de passe pour chaque citoyen).
- *Administrateur Central* : il s'occupe de la gestion des clés (Génération et distribution des clés).
- *Autres* : toute personne qui vérifie les documents à travers le système central.

Il est à noter qu'un cas d'utilisation dit « *Authentification* » décrit un comportement commun pour tous les acteurs.

L'étude de cas d'utilisation a pour objectif de déterminer les interactions entre les acteurs et le système. A ce stade de la modélisation, les interactions représentent les principaux événements qui se produisent dans le domaine de l'application. Plus tard, ces événements sont traduits en messages qui déclenchent des opérations.

En étudiant les différents besoins fonctionnels des utilisateurs du système, nous pouvons sortir avec les catégories des acteurs illustrés par le diagramme des cas d'utilisation suivant :

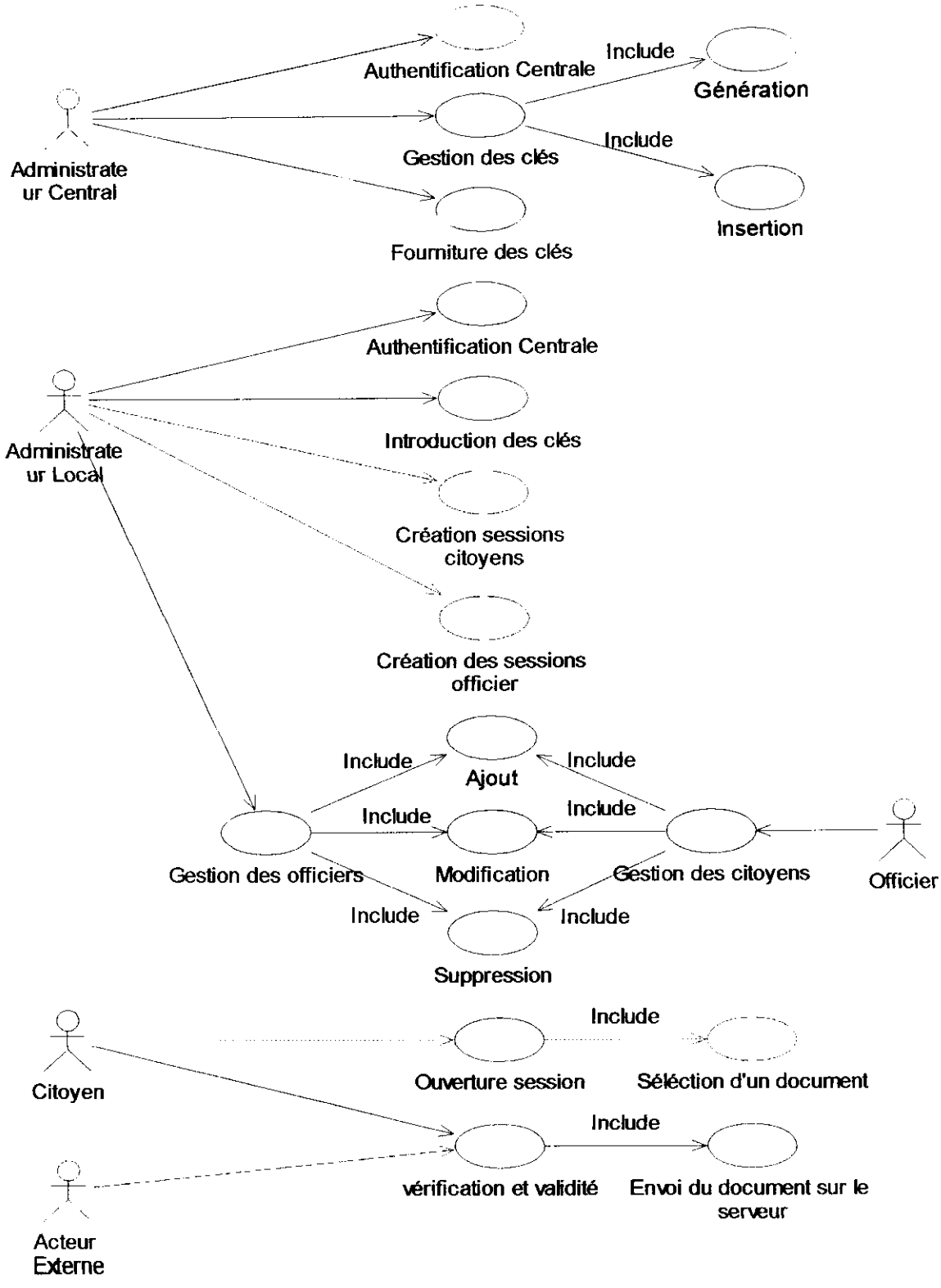


Figure IV.1 : Le diagramme de cas d'utilisation.

Pour illustrer notre démarche, nous allons détailler les cas d'utilisation suivants:

1. Authentification centrale.
2. Gestion des clés.
3. Fourniture des clés.
4. Authentification locale.
5. Introduction des clés.
6. Gestion des officiers.
7. Gestion des citoyens.
8. Création des sessions citoyen.
9. Création des sessions officier.
10. Sélection des documents des documents.
11. Ouverture d'une session.
12. Vérification des clés.

II.2. Les activités et les scénarios:

II.2.1.1 Système Central (Activités de l'administrateur central : Gestion des clés) :

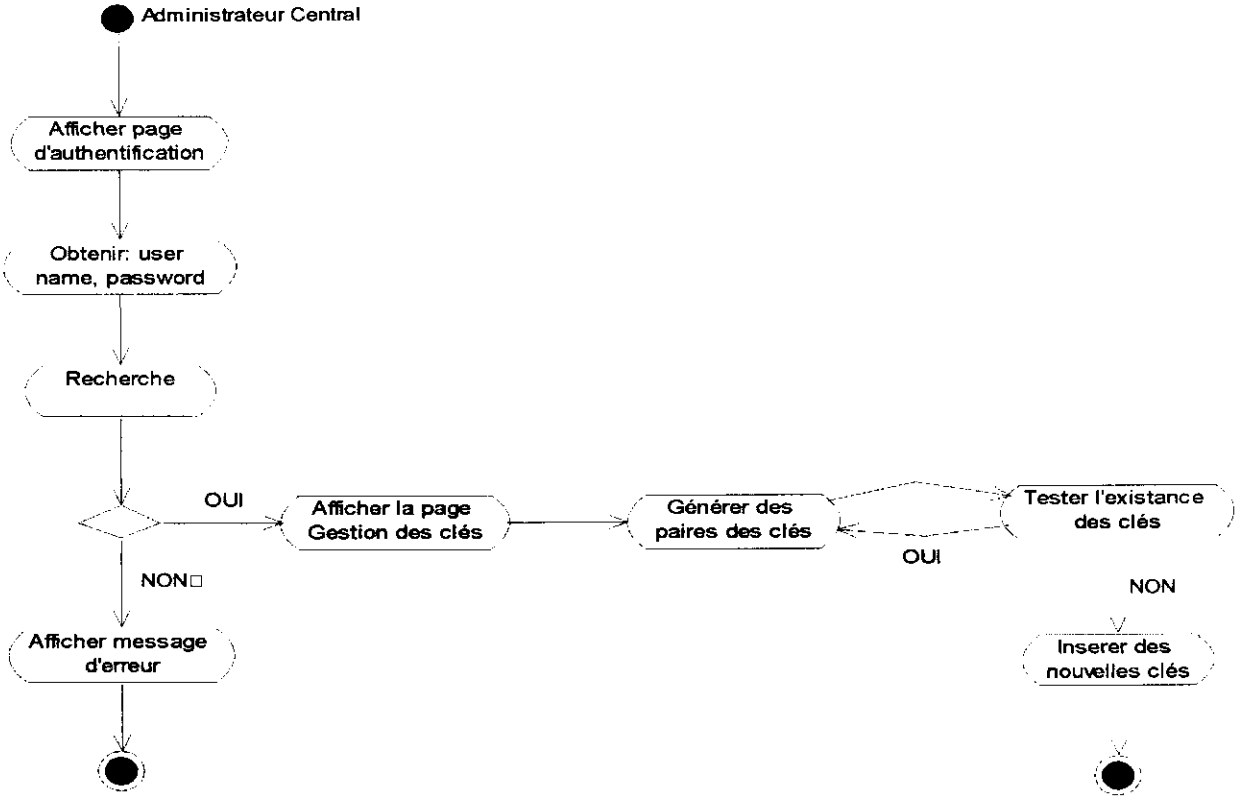


Figure IV.2 : Diagramme d'activité du système central du cas d'utilisation Gestion des clés.

II.2.1.2 Scénario :

1- Le système central affiche la page d'authentification d'administrateur central.

2- Le système demande à l'administrateur central de fournir son nom d'utilisateur et son mot de passe.

3- L'administrateur central fournit les informations demandées.

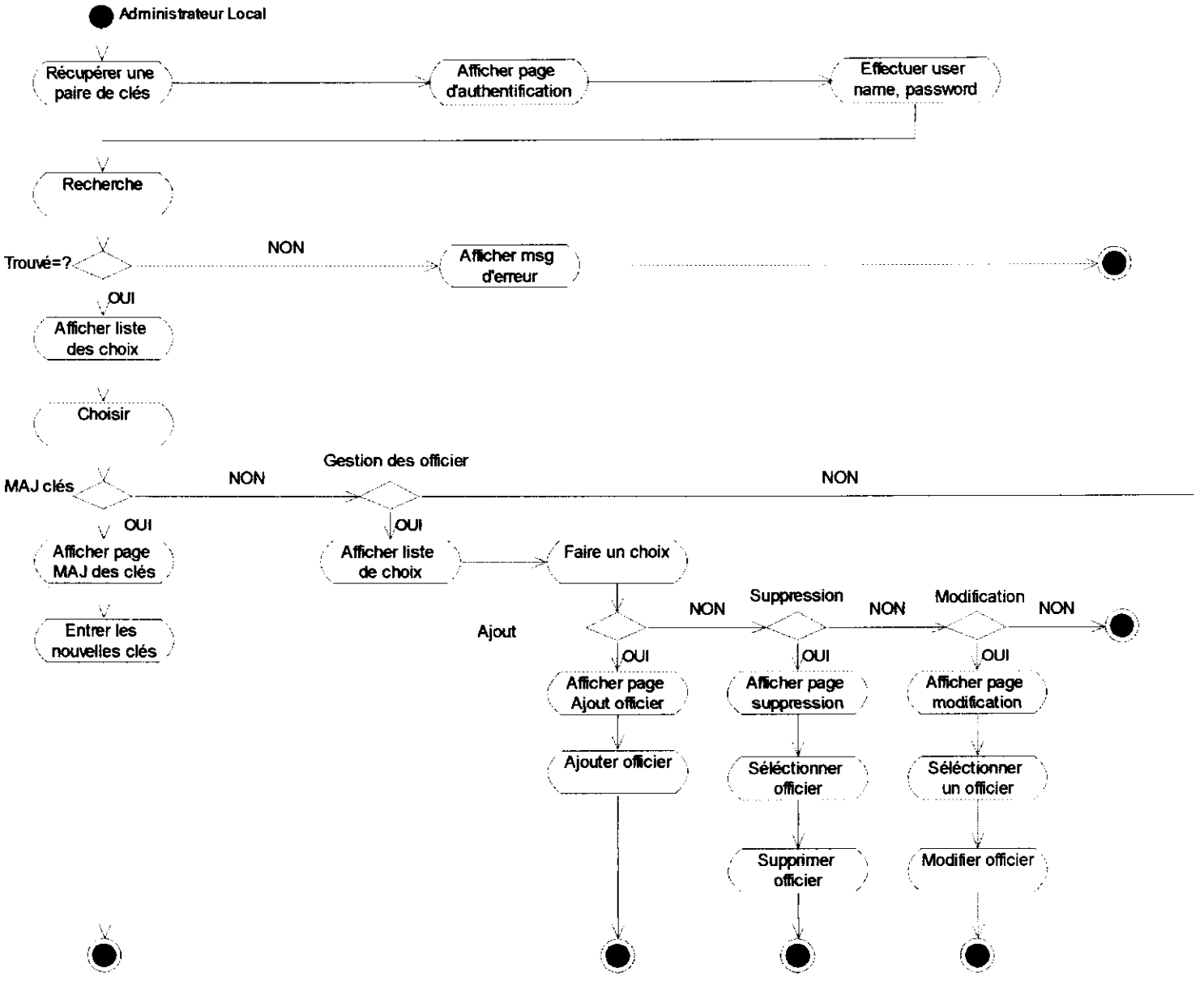
4- Le système central vérifie le nom d'utilisateur et le mot de passe.

5- Si authentification réussite alors:

- Le système affiche la page de gestion des clés.
- L'administrateur central génère une paire de clés (privée et public).
- Le système central teste l'existence des clés.
- Si les clés existent déjà, alors l'administrateur régénère une autre paire de clés.
- Si non, le système central insère les nouvelles clés et les enregistre dans la base de données.

6- Si authentification non trouvée, alors un message d'erreur s'affiche.

II.2.2.1. Système local (Activités de l'administrateur local) :



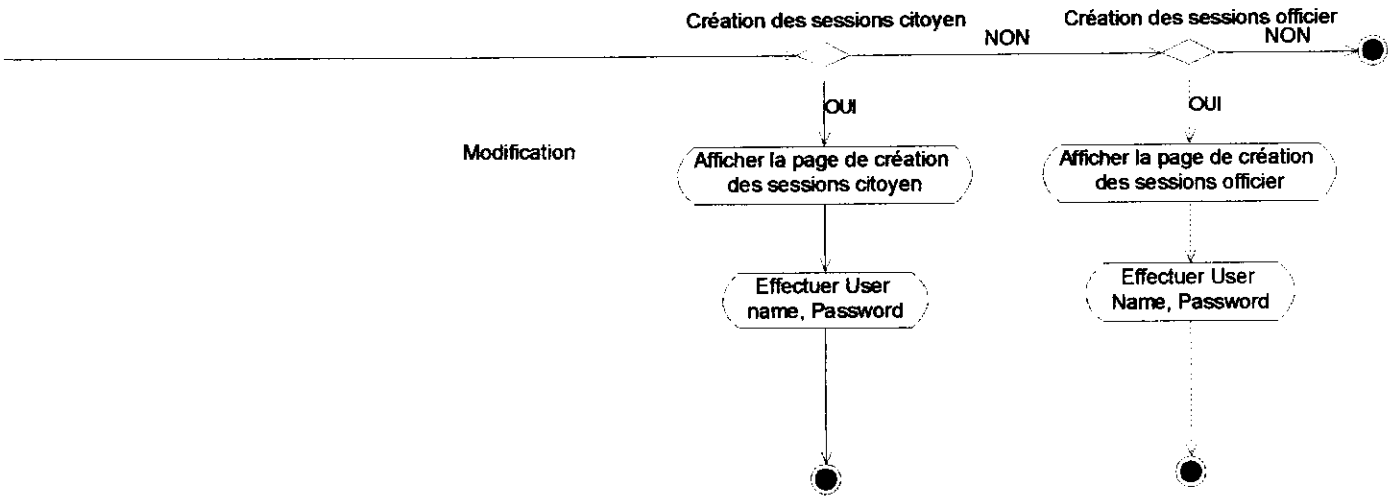


Figure IV.3 : Diagramme d'activités de l'administrateur local.

II.2.2.2. Scénario :

1- L'administrateur local récupère une paire de clés.
2- Le système local affiche la page d'authentification locale.
3- Le système local demande à l'utilisateur de fournir son non d'utilisateur et son mot de passe.
4- Le système local fait une recherche d'authentification.
5- Si utilisateur non authentifié alors :
   5.1- afficher un message d'erreur.
6- Sinon (utilisateur authentifié).
   6.1- Une liste de choix s'affiche.
   6.2- L'administrateur local effectue son choix.
   6.3- Si choix = MAJ des clés alors :
      6.3.1- La page de MAJ des clés s'affiche.
      6.3.2- L'administrateur local introduit les clés.
   6.4- Si choix = Gestion des officiers alors :
      6.4.1- Une liste de choix s'affiche.
      6.4.2- L'administrateur local fait son choix.
      6.4.3- Si le choix = Ajout d'un officier alors :

6.4.3.1- La page d'ajout d'un officier s'affiche.

6.4.3.2- L'administrateur local remplit l'enregistrement d'un officier.

6.4.4- Si le choix = suppression d'un officier alors :

6.4.4.1- Le système affiche la table d'officiers.

6.4.4.2- L'administrateur local fait la suppression.

6.4.4.3- Le système enregistre les modifications effectuées.

6.4.5- Si le choix = modification d'un officier alors :

6.4.5.1- La table des officiers s'affiche.

6.4.5.2- L'administrateur local sélectionne l'officier qu'il va modifier.

6.4.5.3- Le système enregistre ces modifications.

6.5- Si choix= Création des sessions citoyen alors :

6.5.1- La page de création sessions citoyen s'affiche.

6.5.2- L'administrateur local effectue un nom d'utilisateur et un mot de passe pour le citoyen.

6.6- Si choix= Création des sessions officier alors :

6.6.1- La page de création sessions officier s'affiche.

6.6.2- L'administrateur local effectue un nom d'utilisateur et un mot de passe pour l'officier.

II.2.3.1. Activités d'officier :

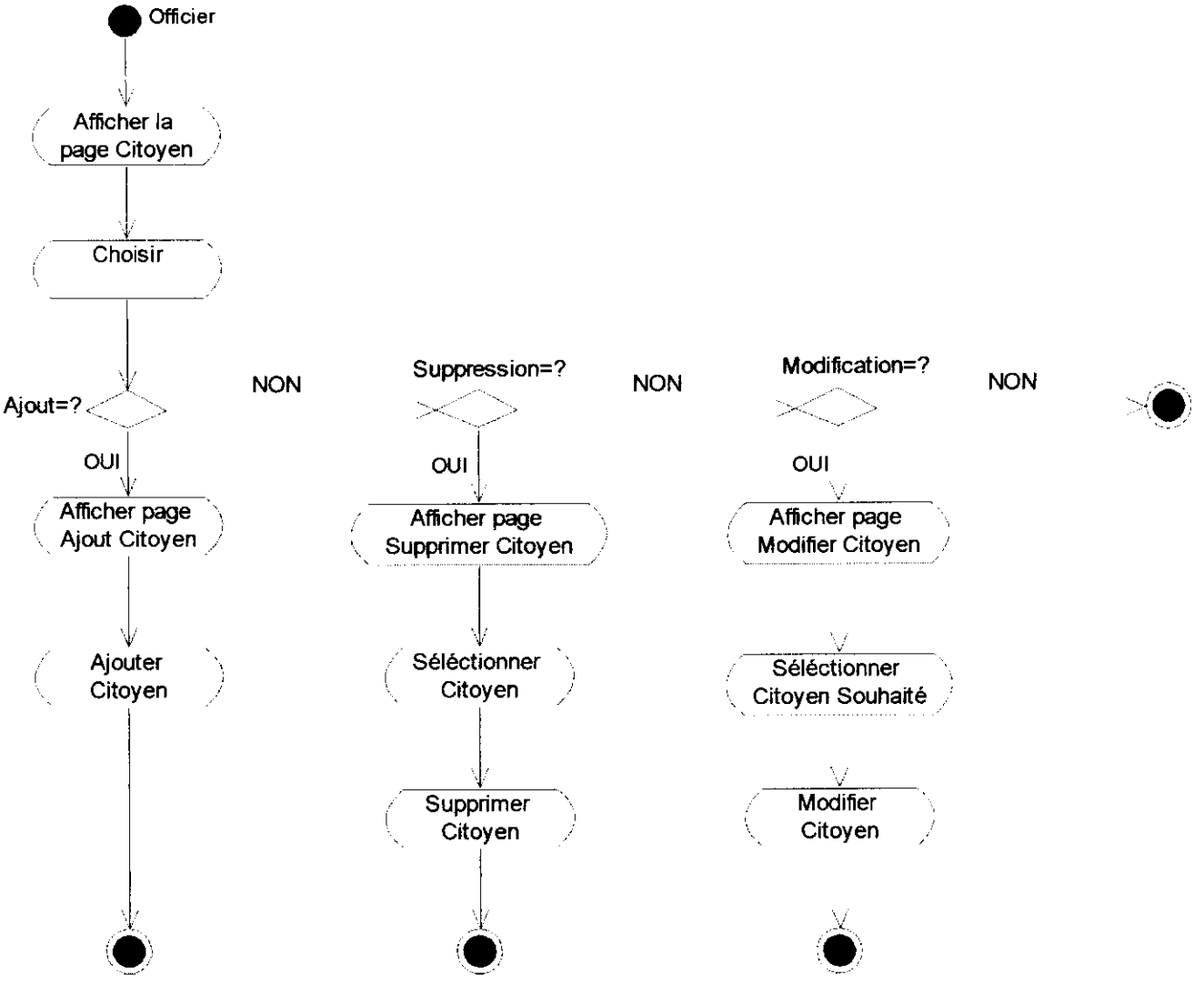


Figure IV.4 : Diagramme d'activités d'officier.

II.2.3.2. Scénario :

1- Le système affiche la page citoyen.

2- L'officier fait un choix.

3- Si choix = ajout d'un citoyen alors :

3.1- La page d'ajout des citoyens s'affiche.

3.2- L'officier ajoute un citoyen.

3.3- Le système enregistre les informations des citoyens dans la base de données.

4- Si choix = suppression d'un citoyen alors :

4.1- La table des citoyens s'affiche.

4.2- L'officier sélectionne le citoyen souhaité.

4.3- L'officier supprime le citoyen sélectionné.

5- Si choix = modification d'un officier alors :

5.1- La table des citoyens s'affiche.

5.2- L'officier sélectionne le citoyen qu'il va modifier.

5.3- L'officier effectue les modifications demandées.

5.4- Le système enregistre les modifications effectuées.

II.2.4.1. Activités Citoyen :

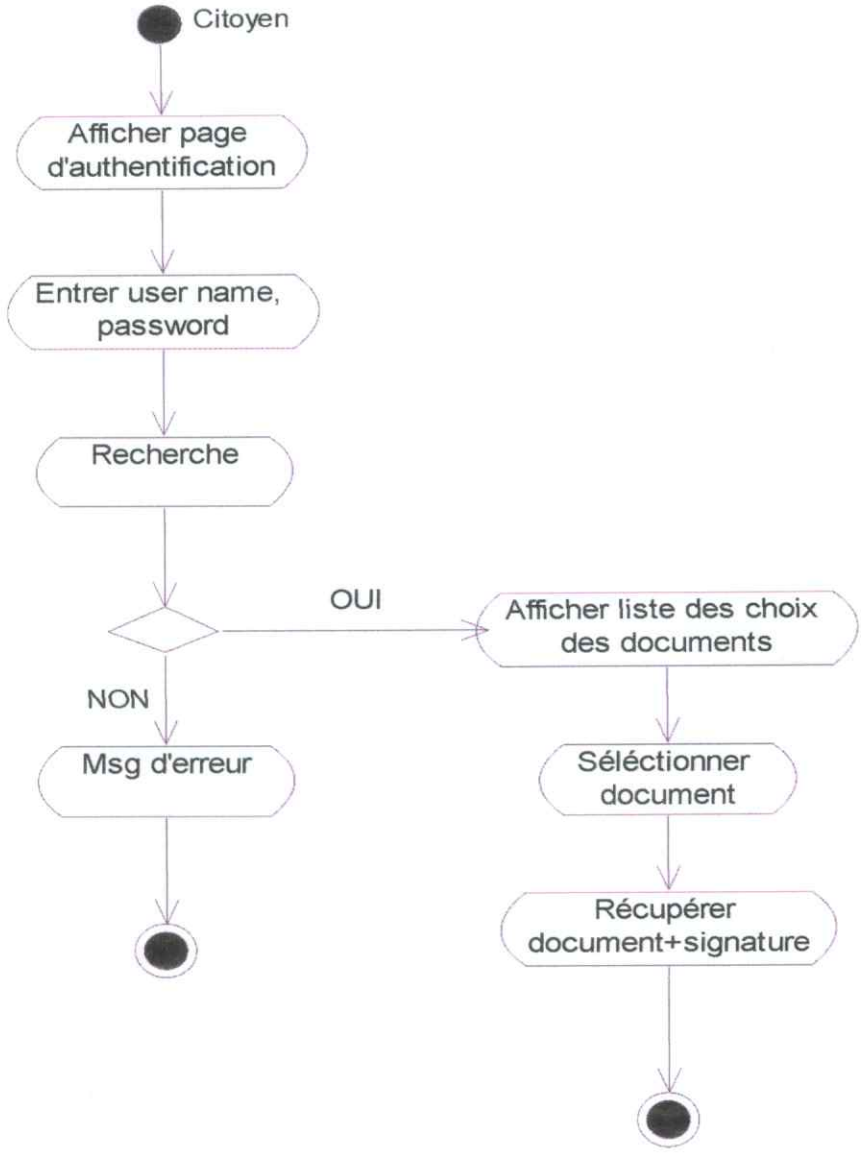


Figure IV.5 : Diagramme d'activités du citoyen.

II.2.4.2. Scénario :

1- Le système affiche la page d'authentification.

2- Le citoyen s'authentifie par son nom d'utilisateur et son mot de passe.

3- Le système recherche l'existence du citoyen.

4- Si citoyen authentifié alors :

4.1- Le système affiche la liste des documents.

4.2- Le citoyen sélectionne le document souhaité.

4.3- Le système génère le document demandé.

4.4- Le citoyen récupère le document avec sa signature.

5- Si citoyen non authentifié alors : un message d'erreur s'affiche.

II.2.5.1. Authentification centrale :

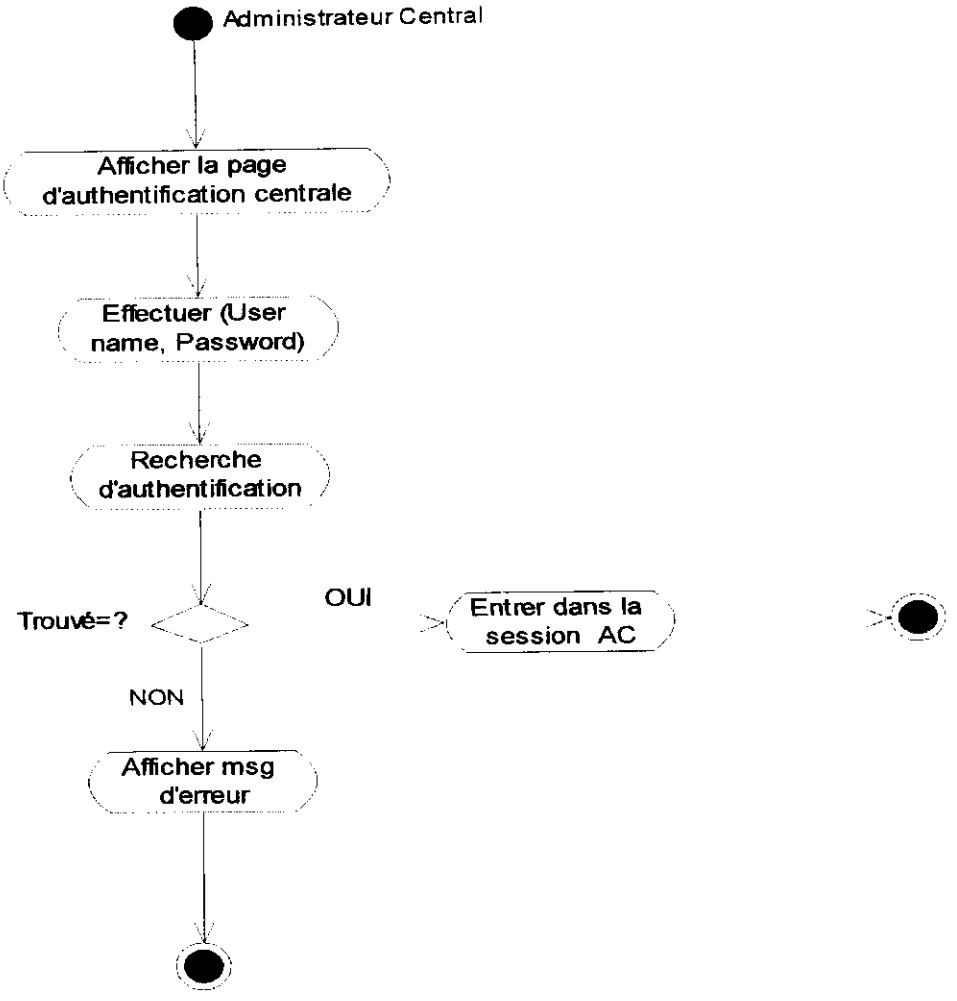


Figure IV.6 : Diagramme d'activité du cas d'utilisation Authentification centrale.

II.2.5.2. Scénario :

1- Le système central affiche la page d'authentification.
2- L'administrateur central fourni les informations demandées (nom d'utilisateur, mot de passe).
3- Le système central vérifie le nom d'utilisateur et le mot de passe de l'administrateur central.
4- Si administrateur central authentifié alors : le système central ouvre la session de ce dernier.
5- Si non, un message d'erreur s'affiche.

II.2.6.1. Authentification locale :

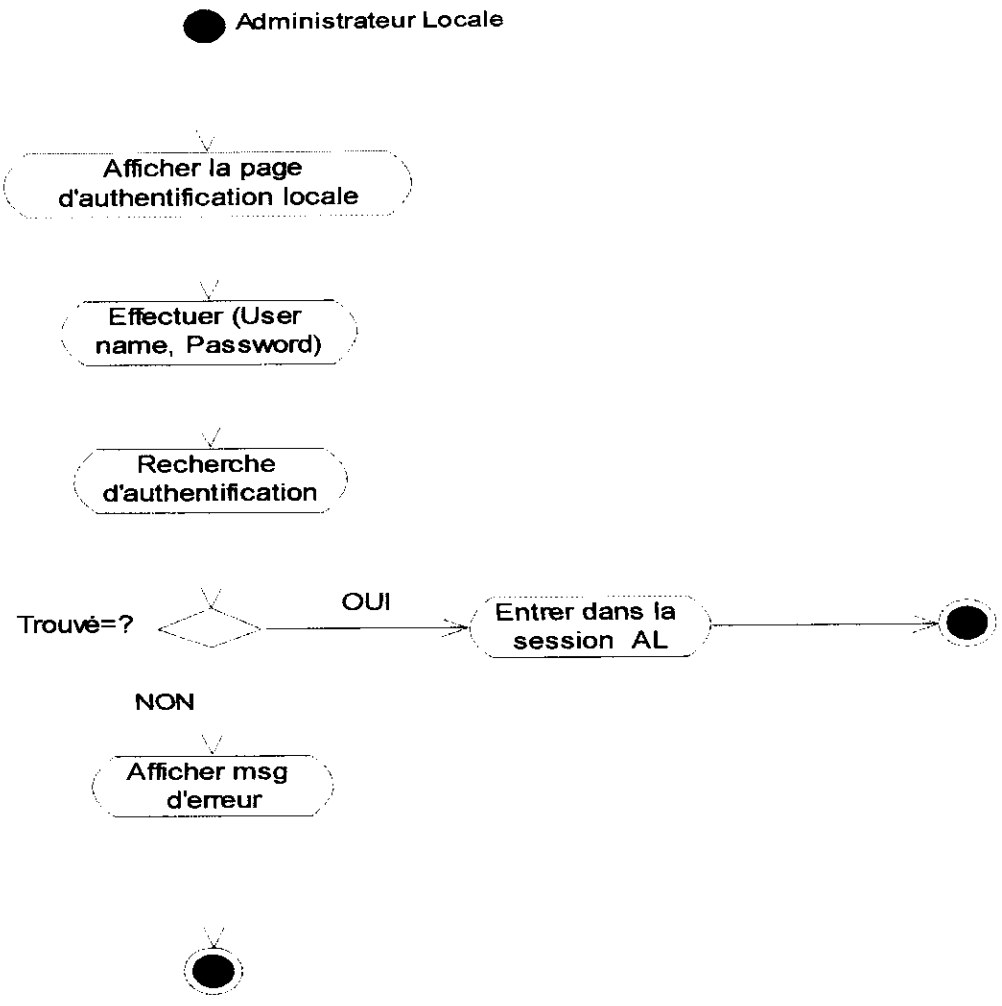


Figure IV.7 : Diagramme d'activité du cas d'utilisation authentification centrale.

II.2.6.2. Scénario :

1- Le système local affiche la page d'authentification.
2- L'administrateur local fourni les informations demandées (nom d'utilisateur, mot de passe).
3- Le système local vérifie le nom d'utilisateur et le mot de passe de l'administrateur local.
4- Si administrateur local authentifié alors : le système local ouvre la session de ce dernier.
5- Si non, un message d'erreur s'affiche.

II.2.7.1. Génération et insertion des clés :

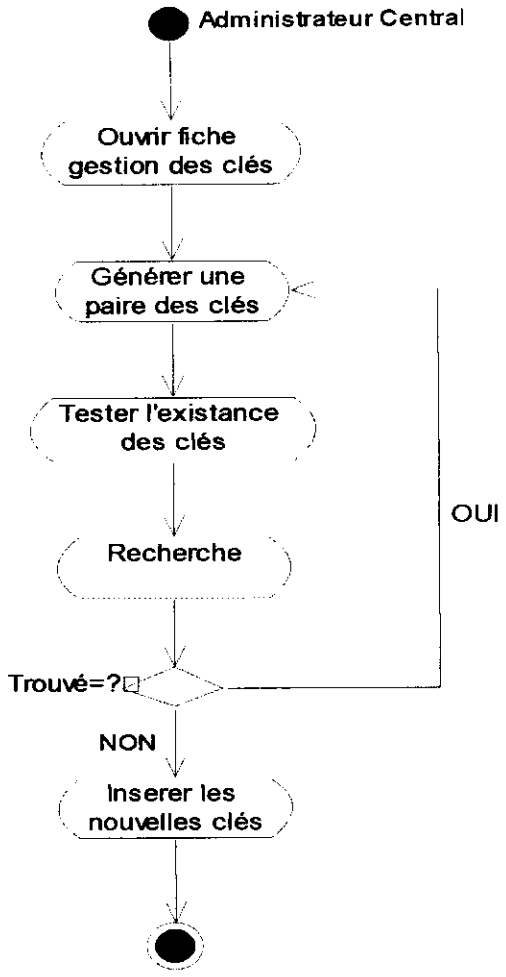


Figure IV.8 : Diagramme d'activité du cas d'utilisation génération et insertion des clés.

II.2.7.2. Scénario :

1- Le système central affiche à l'administrateur central la fiche de la gestion des clés.
2- L'administrateur central génère une paire de clés.
3- Le système central test l'existence des clés.
4- Si les clés existent déjà alors : le système affiche un message d'information et retourne à la phase de génération de clés (phase 2).
5- Si les clés n'existent pas alors : l'administrateur central insère les nouvelles clés dans la base de données centrale.

II.2.8.1. Fourniture des clés :

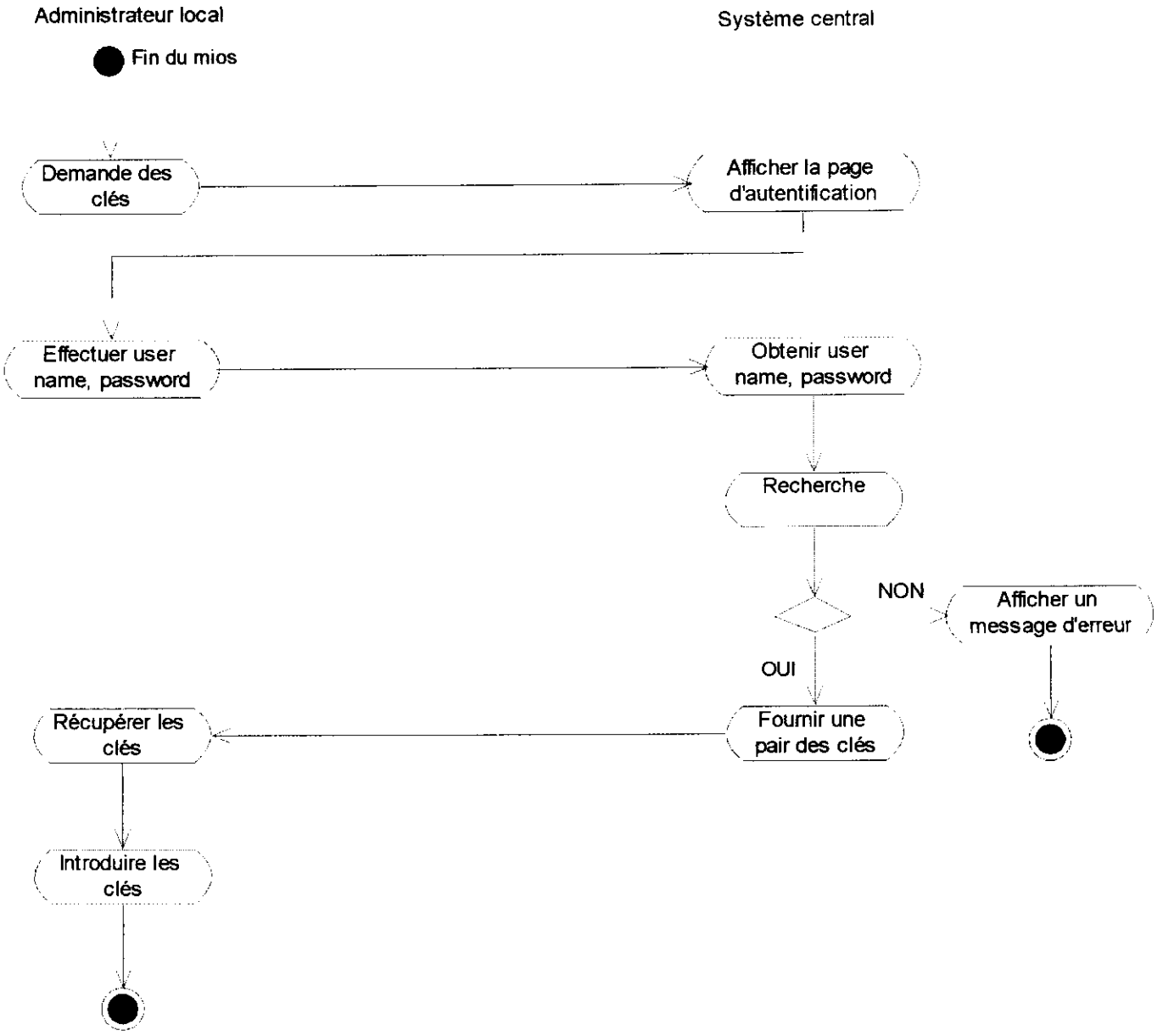


Figure IV.9 : Diagramme d'activité du cas d'utilisation fourniture des clés.

II.2.8.2. Scénario :

1. Chaque fin de mois, l'administrateur local fait une demande de fourniture des clés au système central.
2. Le système central demande à l'administrateur local de saisir son nom d'utilisateur et son mot de passe.
3. L'administrateur local fournit les informations demandées.
4. Le système central vérifie le nom d'utilisateur et le mot de passe de l'administrateur local.
5. Si administrateur local authentifié alors :
   5.1- le système central fournit une paire de clés administrateur local.
   5.2- L'administrateur local récupère les clés et les introduit dans la base de données locale.
6. Si non (authentification échouée), un message d'erreur s'affiche.

II.2.9.1. Ouvrir session Citoyen :

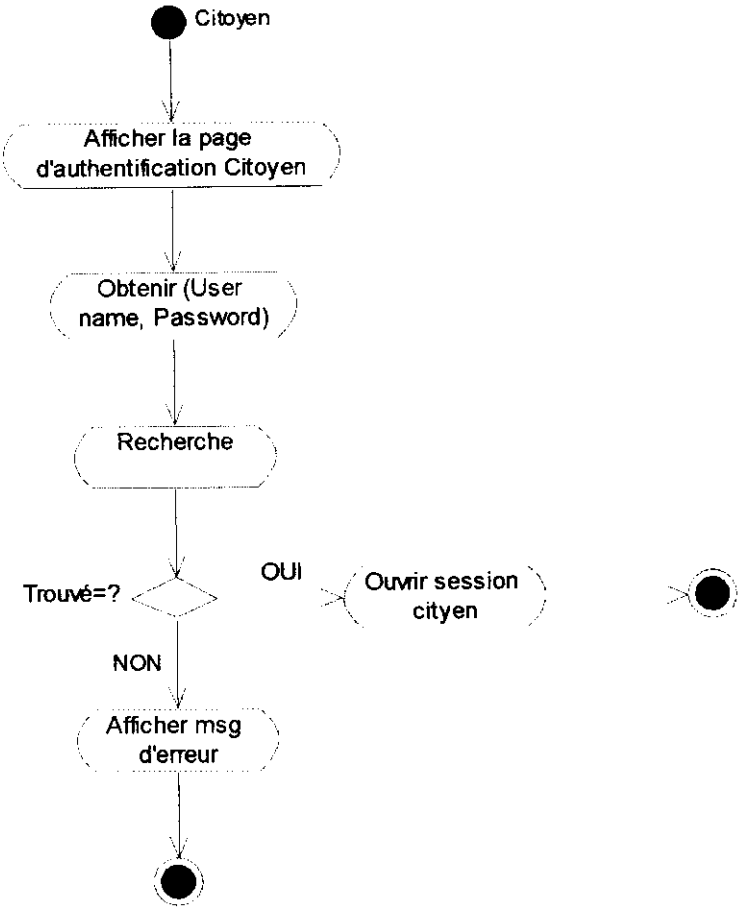
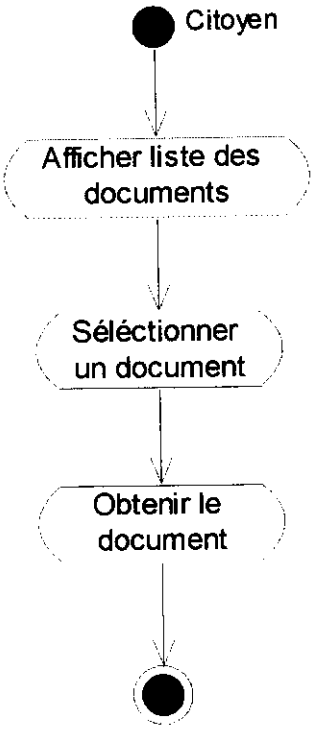


Figure IV.10 : Diagramme d'activité de cas d'utilisation ouvrir session citoyen.

II.2.9.2. Scénario :

1- Le système demande au citoyen de fournir son nom d'utilisateur et son mot de passe.
2- Le citoyen fournit les informations demandées.
3- Le système vérifie le nom d'utilisateur et le mot de passe du Citoyen.
4- Si citoyen authentifié, le système lui ouvre sa session.
5- Sinon, un message d'erreur s'affiche.

II.2.10.1. Sélectionner document :



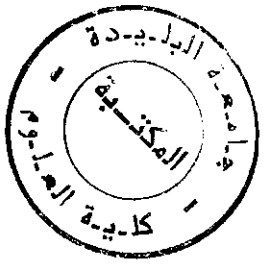


Figure IV.11 : Diagramme d'activité du cas d'utilisation sélection d'un document

II.2.10.2. Scénario :

1- Le système affiche au citoyen la liste des documents.
2- Le citoyen sélectionne un document.
3- Le système lui affiche le document.

II.2.11.1. Vérification et Validation :

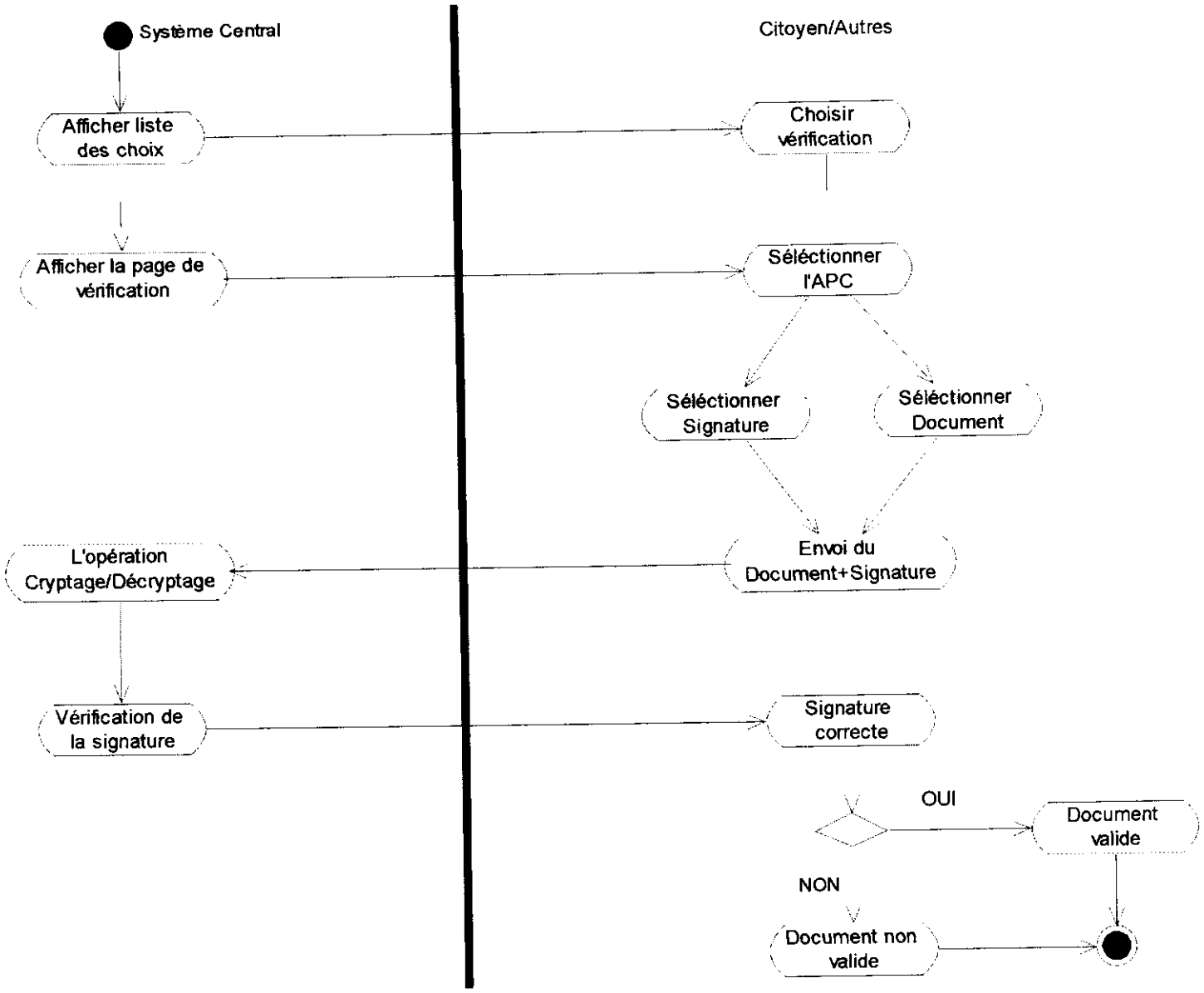


Figure IV.12 : Diagramme d'activité du cas d'utilisation vérification et validation.

II.2.11.2. Scénario :

1- Le système central affiche une liste des choix à l'utilisateur.
2- L'utilisateur choisit "vérification et validation des documents".
3- Le système central lui affiche la page de vérification et validation des documents.
4- L'utilisateur sélectionne l'APC et ensuite le document et sa signature.
5- L'utilisateur envoie le document et sa signature au système central.
6- Le système central exécute les opérations de cryptage/décryptage pour recalculer la signature.
7- Si la signature est correcte (la nouvelle signature n'est pas différente de l'ancienne) alors : le document est valide.
8- Si non, le document n'est pas valide.

II.2.12.1. Cryptage :

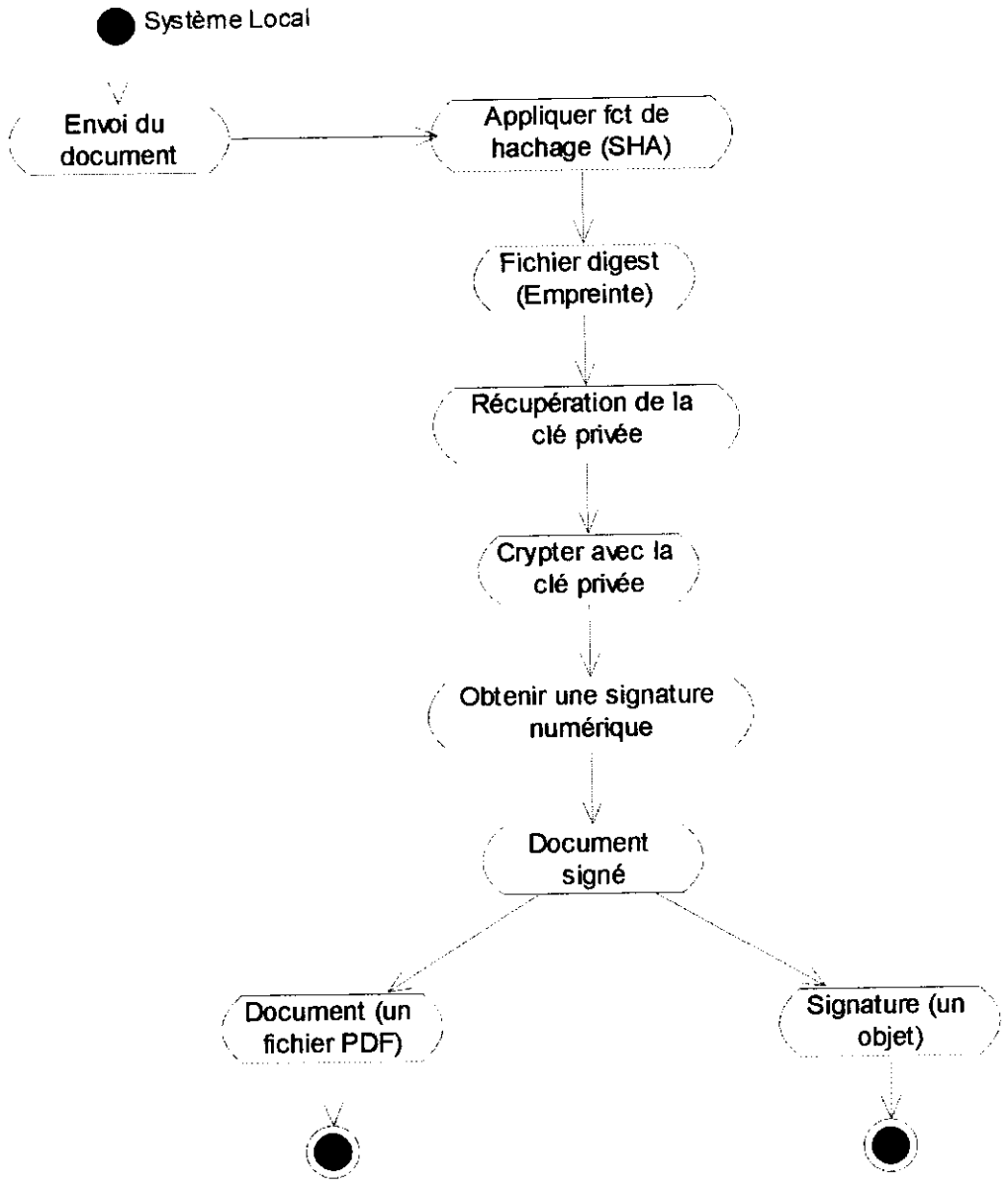


Figure IV.13 : Diagramme d'activité de l'opération de cryptage.

II.2.12.2. Scénario :

1- Le système local applique la fonction de hachage au document (dans notre cas nous avons utilisé le algorithme SHA).
2- Le résultat est un fichier digest (une empreinte).
3- Le système récupère la clé privée.
4- L'empreinte sera cryptée avec la clé privée, on appliquant l'algorithme RSA.
5- Le résultat de cryptage est une signature numérique.
6- Le système central affiche le document dans un fichier PDF, et sa signature numérique dans un objet.

II.2.13.1. Décryptage :

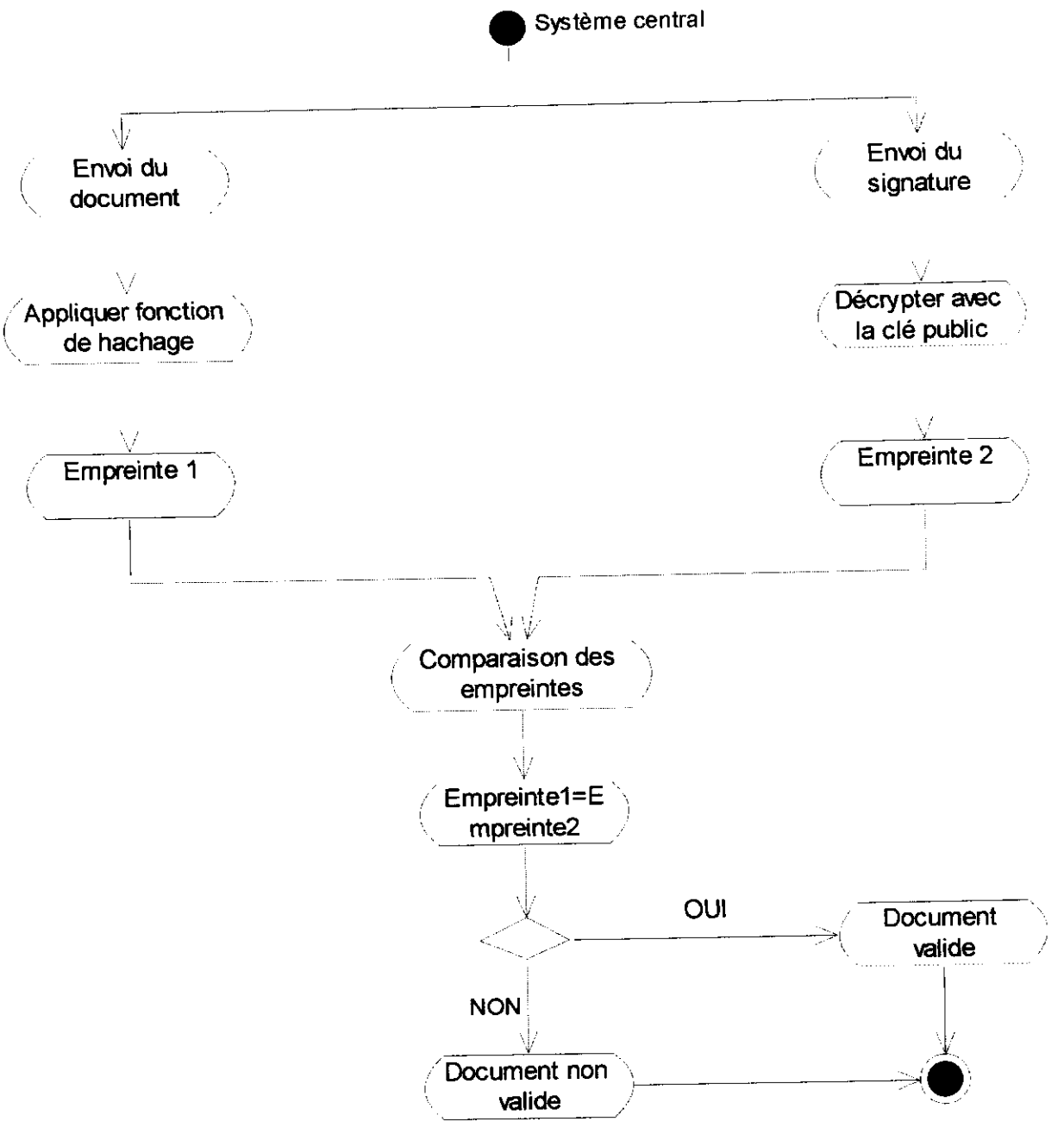


Figure IV.14 : Diagramme d'activité de l'opération de décryptage.

II.2.13.2. Scénario :

1- Dès la réception d'un document et de sa signature, le système central applique une fonction de hachage (SHA) sur le document reçu.
2- Le résultat est une première empreinte.
3- Le système central applique une opération de décryptage, il décrypte à la signature numérique reçue avec la clé publique.
4- Le résultat est une deuxième empreinte.
5- Le système central fait une comparaison entre les deux emprunts obtenus.
6- Si empreinte1 = empreinte2 alors : le document est valide.
7- Si non, le document n'est pas valide.

II.3. Les scénarios des cas d'utilisation :

II.3.1. Système Central (Gestion des clés) :

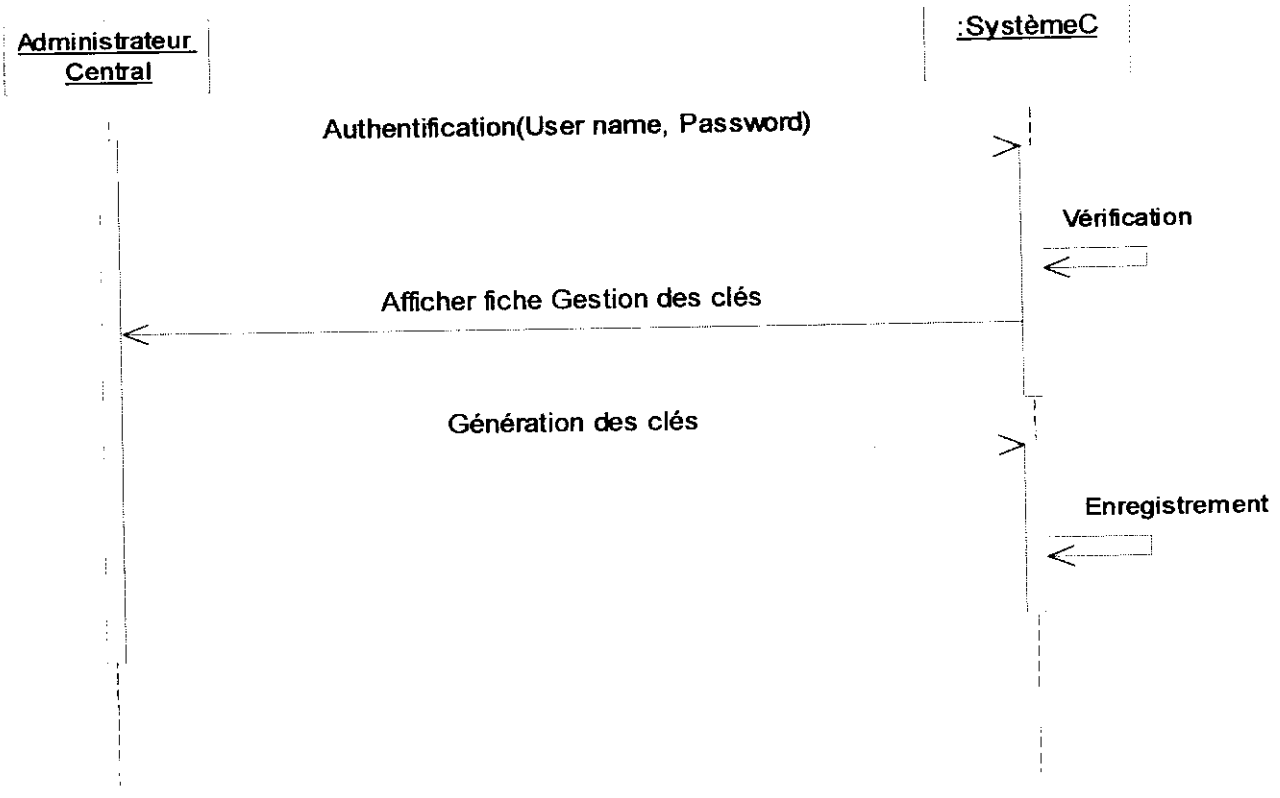


Figure IV.15 : Diagramme de séquence global de la gestion des clés.

II.3.2. Système Central (Génération et validation):

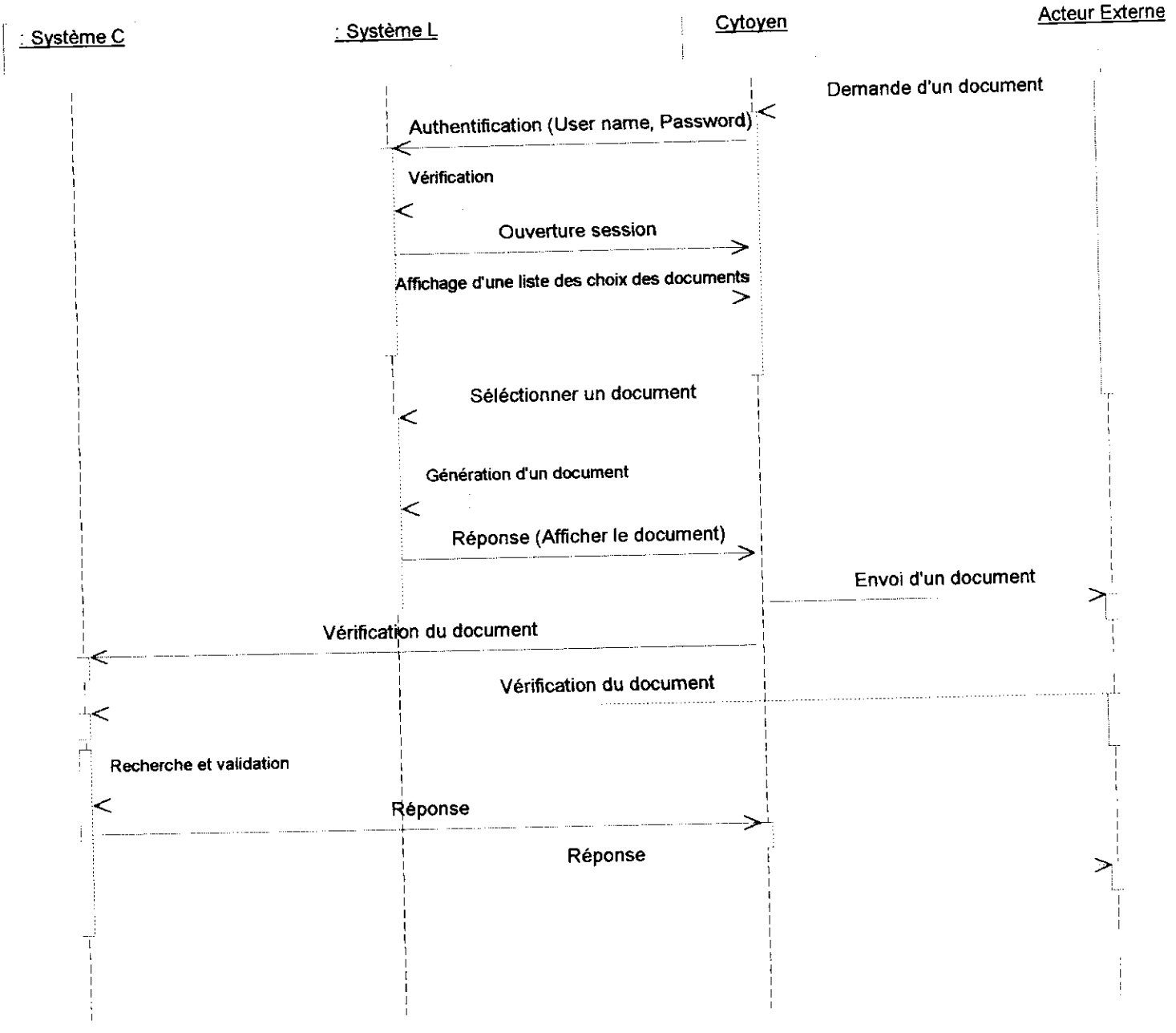


Figure IV.16 : Diagramme de séquence global du génération et validation dans le système central.

II.3.3. Authentification Centrale :

L'administrateur central se connecte au système et donne son nom d'utilisateur ainsi que son mot de passe, le système central vérifie son identité et autorise la connexion.

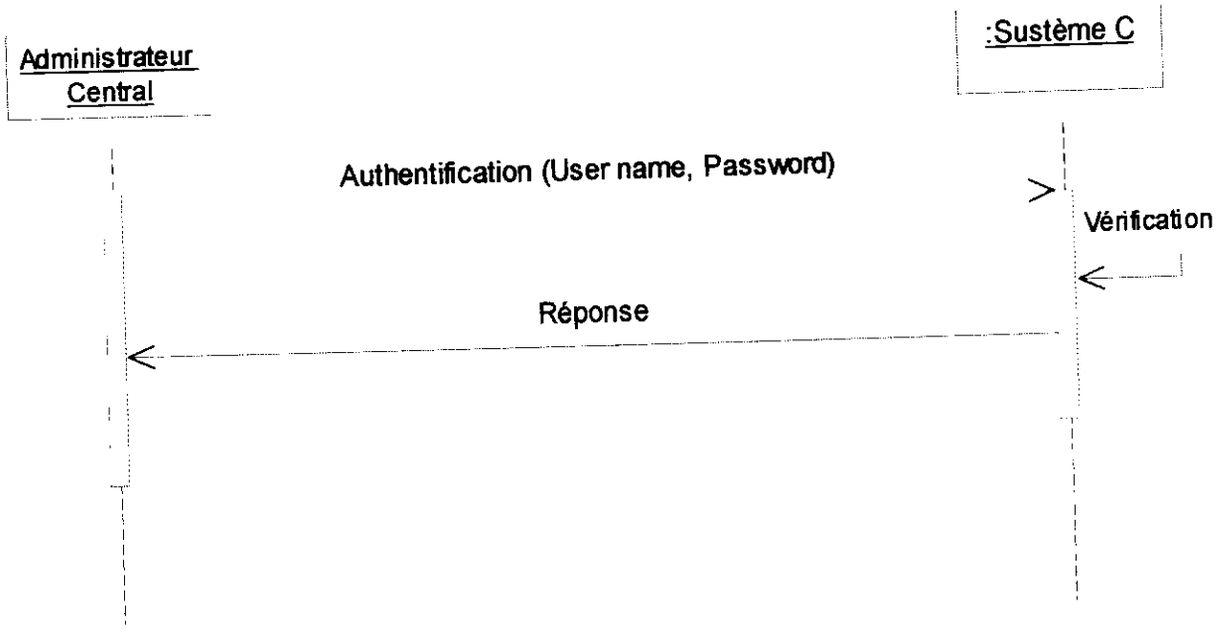


Figure IV.17 : Diagramme de séquence Authentification de l'administrateur central.

II.3.4. Gestion des clés :

Le système central affiche la page de gestion des clés, l'administrateur central choisit l'insertion des clés, le système affiche la page d'insertion, l'administrateur sélectionne l'APC, après une recherche le système affiche l'APC, l'administrateur central insère les nouvelles clés.

Le système enregistre ces informations.

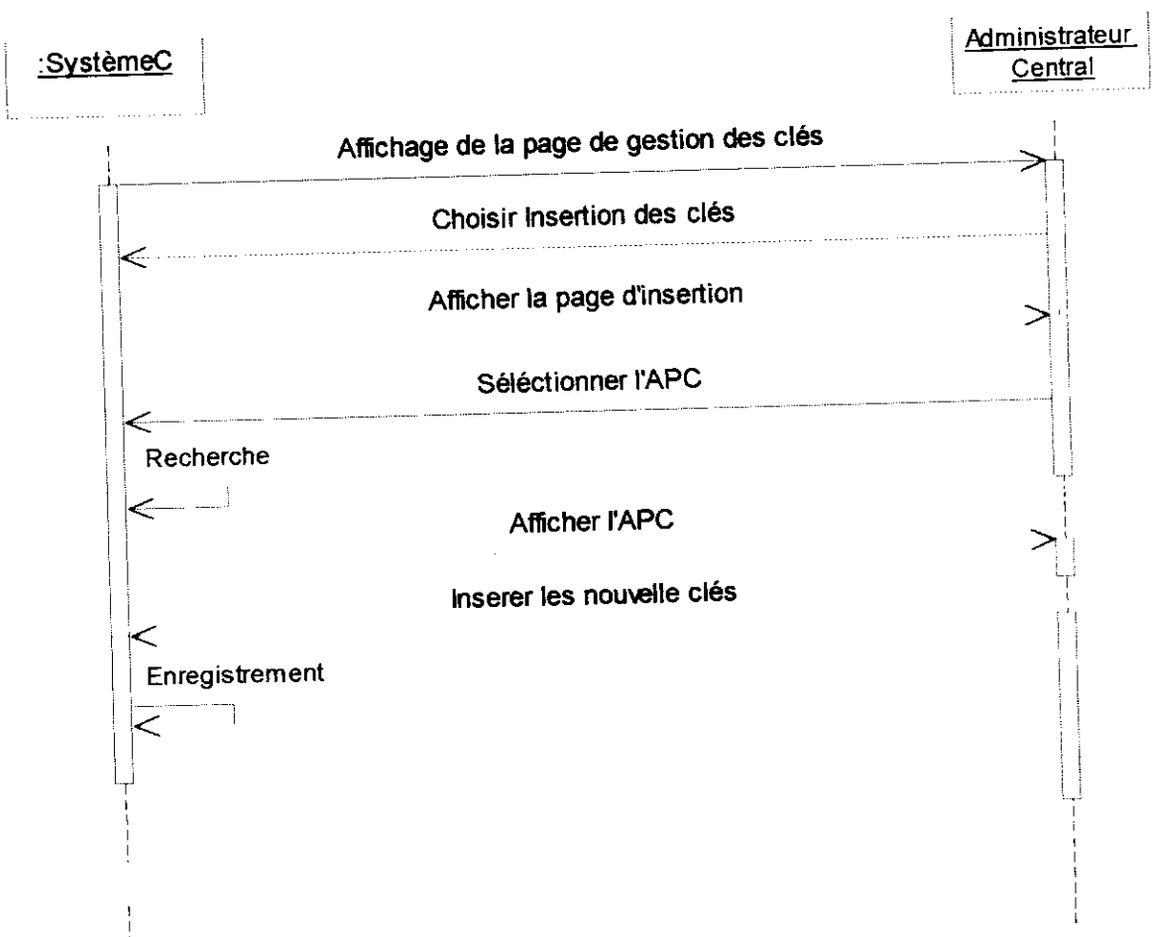


Figure IV.18: Diagramme de séquence Insertion des clés.

II.3.5. Ajout des clés :

L'administrateur local récupère une paire de clés à partir du système central, ce dernier lui affecte une paire de clés, l'administrateur local s'authentifie auprès du système local en entrant son nom d'utilisateur et son mot de passe, après une vérification le système lui autorise pour faire la mis à jour des clés.

Le système enregistre les modifications.

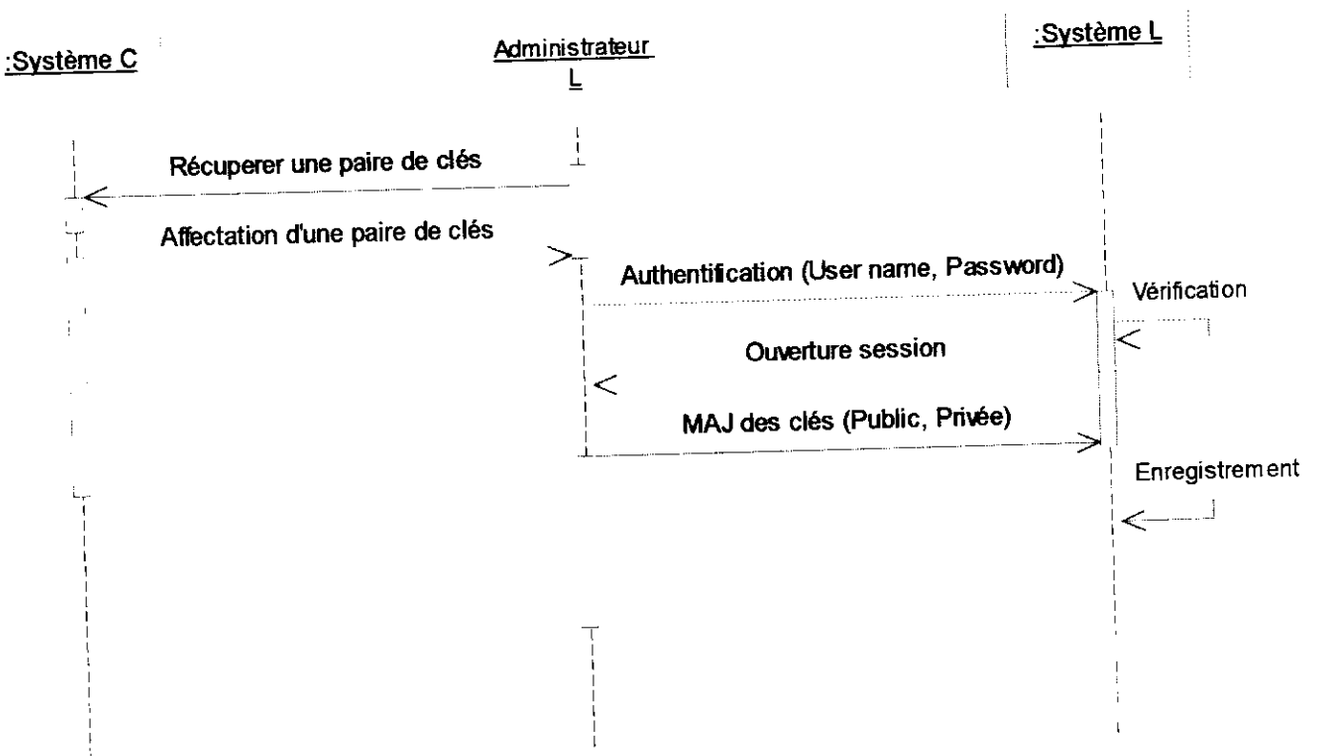


Figure IV.19 : Diagramme de séquence Ajout des clés.

II.3.6. Récupération des clés/MAJ des clés :

Pour protéger les informations qui circulent entre le système central et le système local, nous allons établir un canal sécurisé en utilisant un VPN (Virtual Private Network).

L'administrateur local se connecte au système central en donnant son nom d'utilisateur et son mot de passe, le système central vérifie l'identité de l'utilisateur et autorise la connexion.

L'administrateur local récupère les clés à travers une connexion sécurisée.

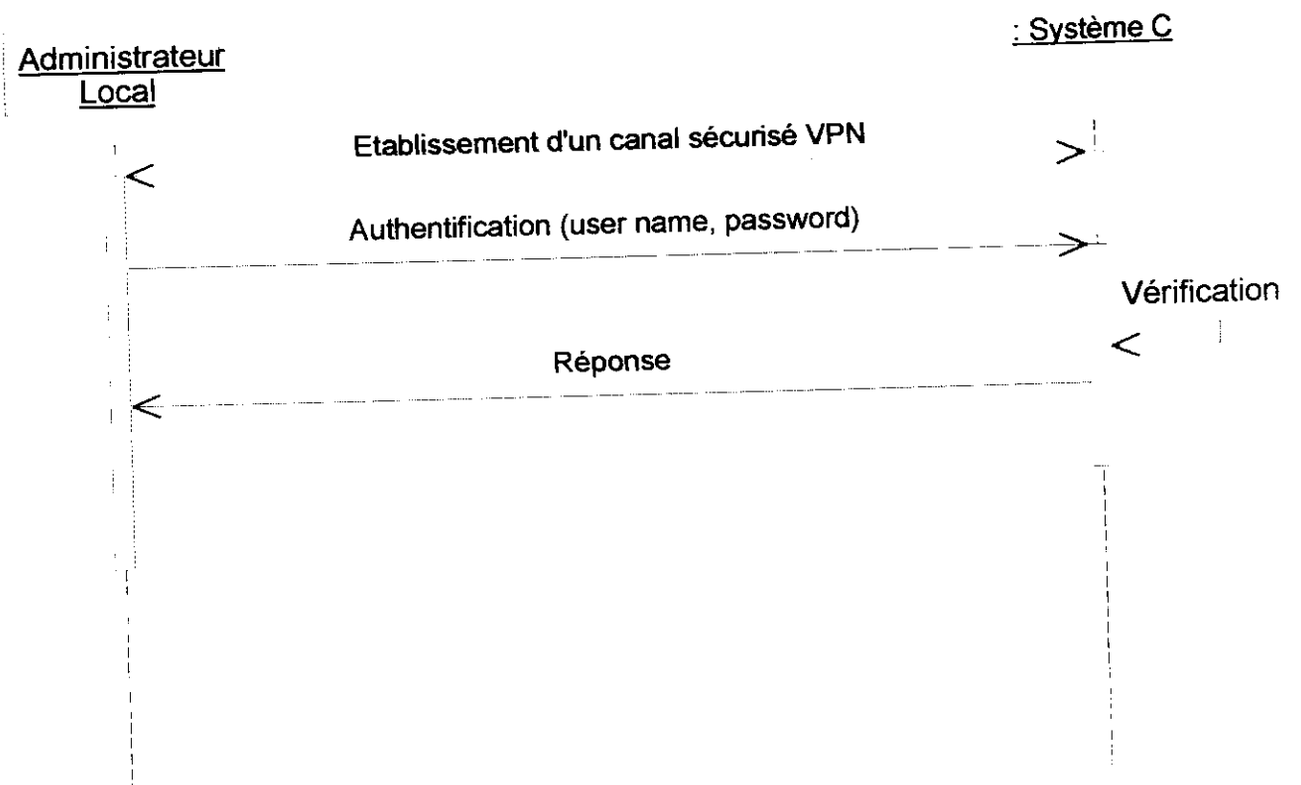


Figure IV.20 : Diagramme de séquence récupération des clés.

II.3.7. Suppression des clés :

Le système central affiche la page de gestion des clés, l'administrateur sélectionne l'APC, après une recherche le système affiche l'APC demandée, l'administrateur central choisit la suppression des clés, le système affiche la page de suppression pour que l'administrateur central effectue la suppression.

Le système central enregistre les modifications.

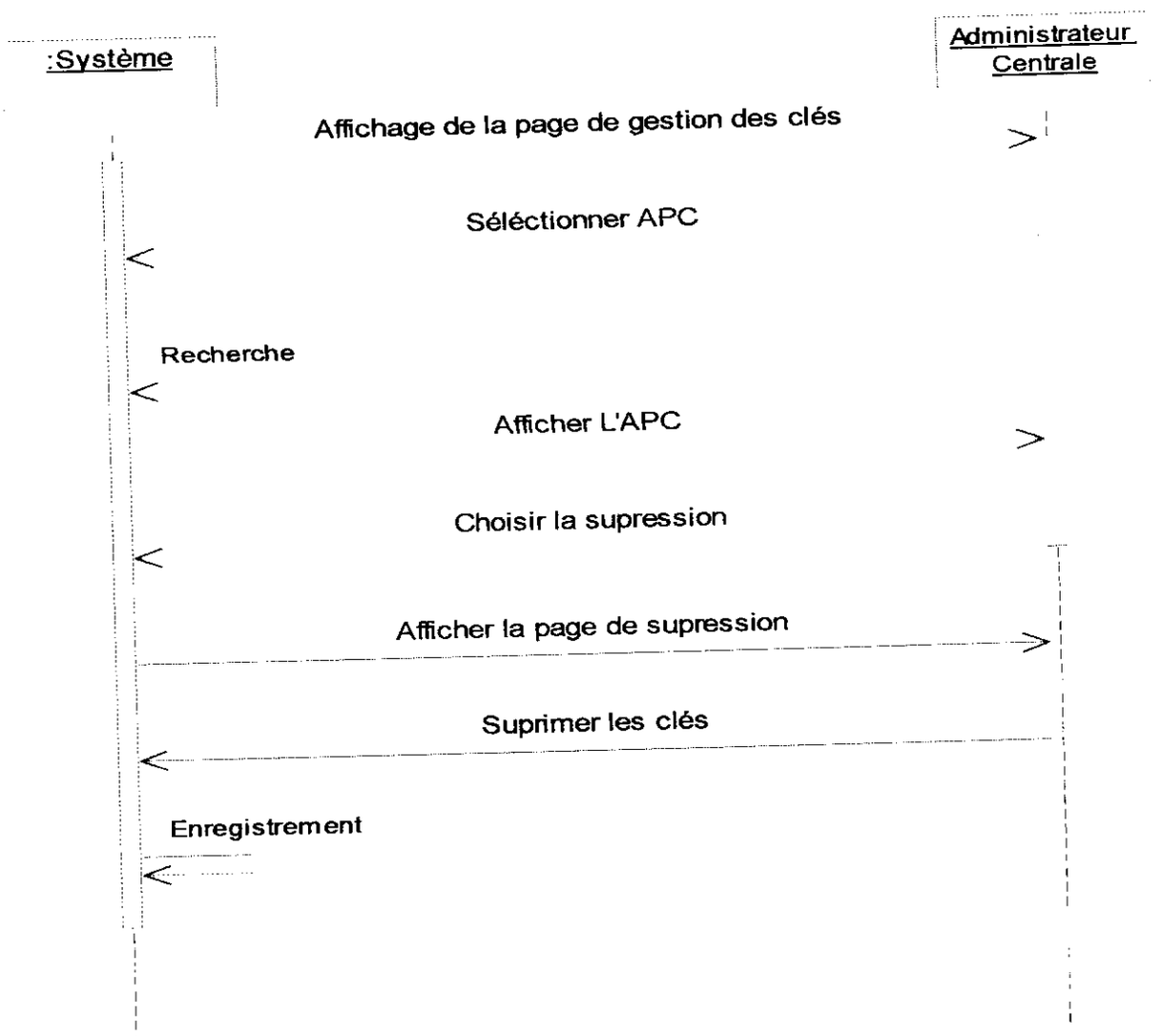


Figure IV.21 : Diagramme de séquence suppression des clés.

II.3.8. Authentification locale :

L'administrateur local se connecte au système et donne son nom d'utilisateur ainsi que son mot de passe, le système local vérifie l'identité de l'administrateur et autorise la connexion.

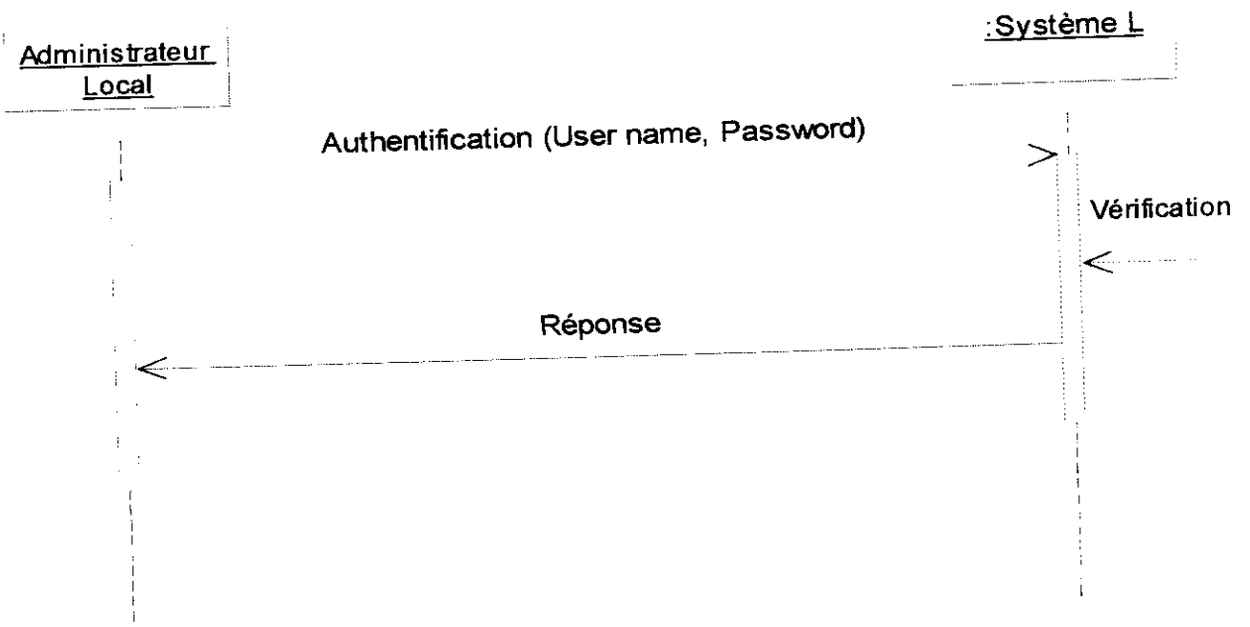


Figure IV.22 : Diagramme de séquence authentification de l'administrateur local.

II.3.9. Ajout d'un officier :

Le système local affiche la page d'officier, l'administrateur local choisi l'ajout d'un officier et remplit toutes la fiche d'officier avec toutes les informations.

Le système enregistre ces informations.

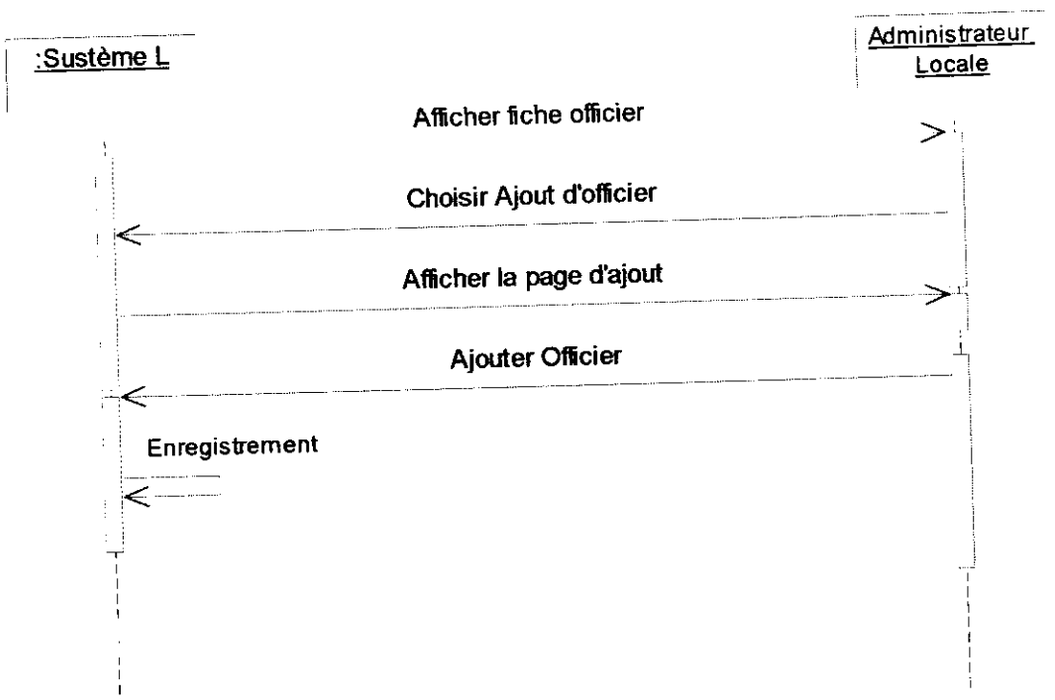


Figure IV.23 : Diagramme de séquence Ajout d'un officier

II.3.10. Suppression d'un officier :

Le système local affiche la fiche officier, l'administrateur local choisit la suppression d'un officier, il sélectionne l'officier et il effectue la suppression.

Le système enregistre les modifications.

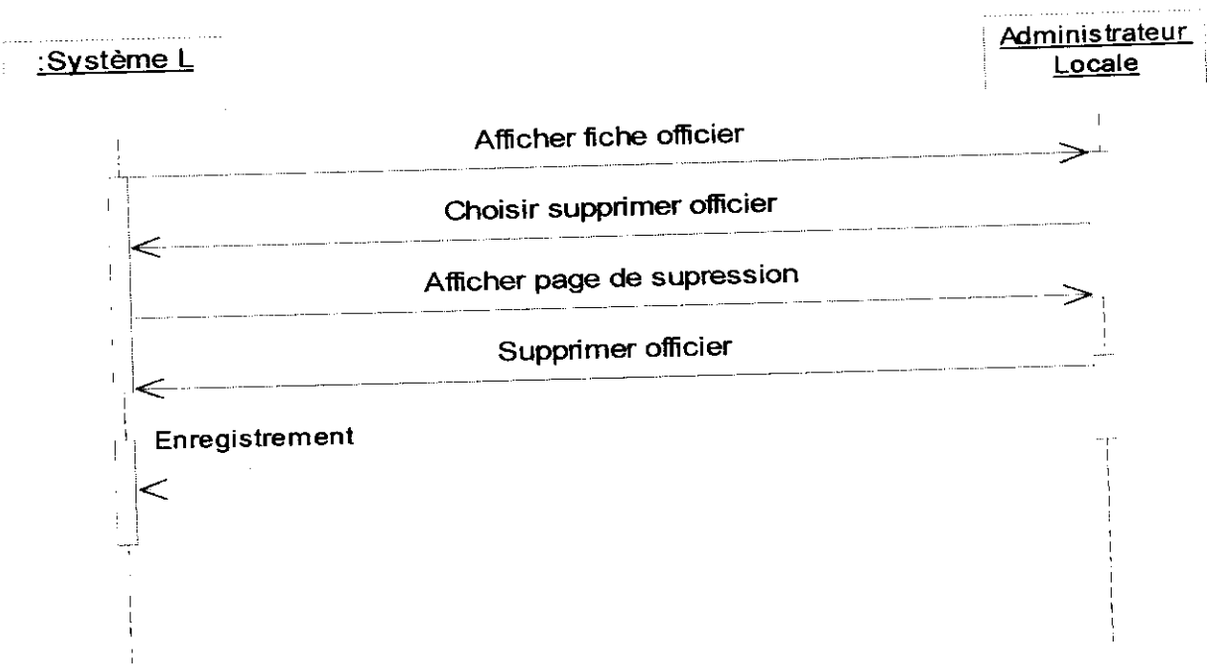


Figure IV.24 : Diagramme de séquence Suppression d'un officier.

II.3.11. Modifier un officier :

Le système local affiche la fiche officier, l'administrateur local choisit la modification d'un officier, il sélectionne l'officier souhaité et il effectue la modification.

Le système local enregistre ces modifications.

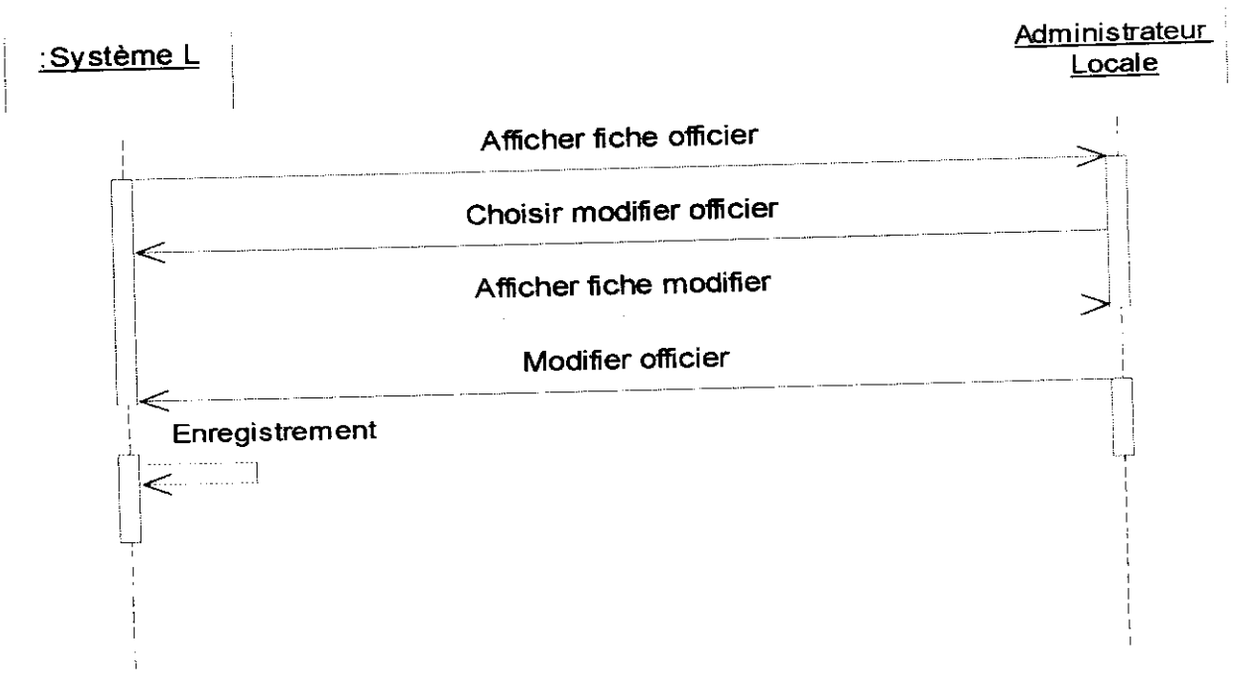


Figure IV.25: Diagramme de séquence modification d'un officier.

II.3.12. Création des sessions citoyen :

Le système affiche la page de création des sessions pour les citoyens, l'administrateur local effectue pour chaque citoyen un nom d'utilisateur et un mot de passe.

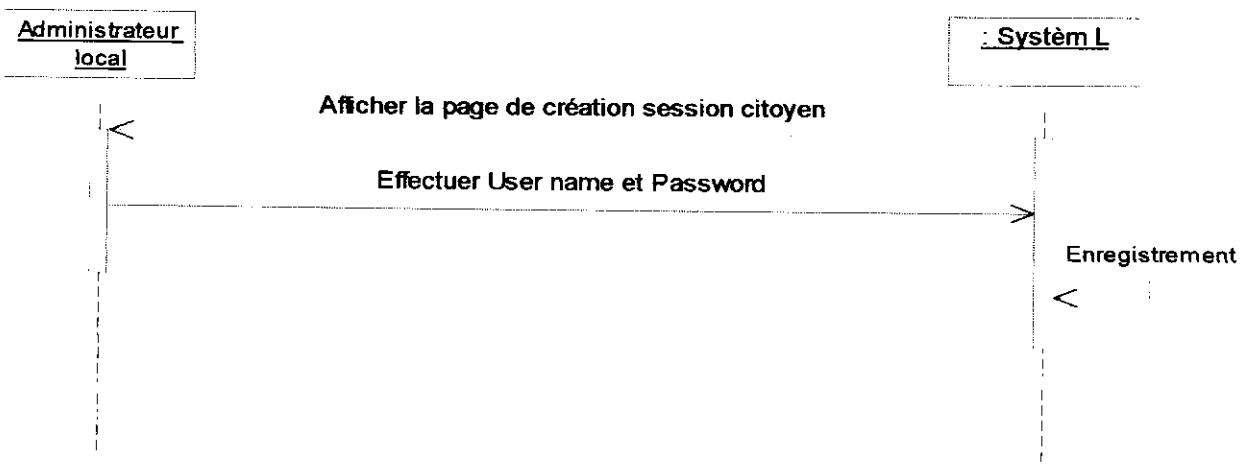


Figure IV.26 : Diagramme de séquence création des session citoyen.

II.3.13. Création des sessions officier :

Le système affiche la page de création des sessions pour les citoyens, l'administrateur local effectue pour chaque citoyen un nom d'utilisateur et un mot de passe.

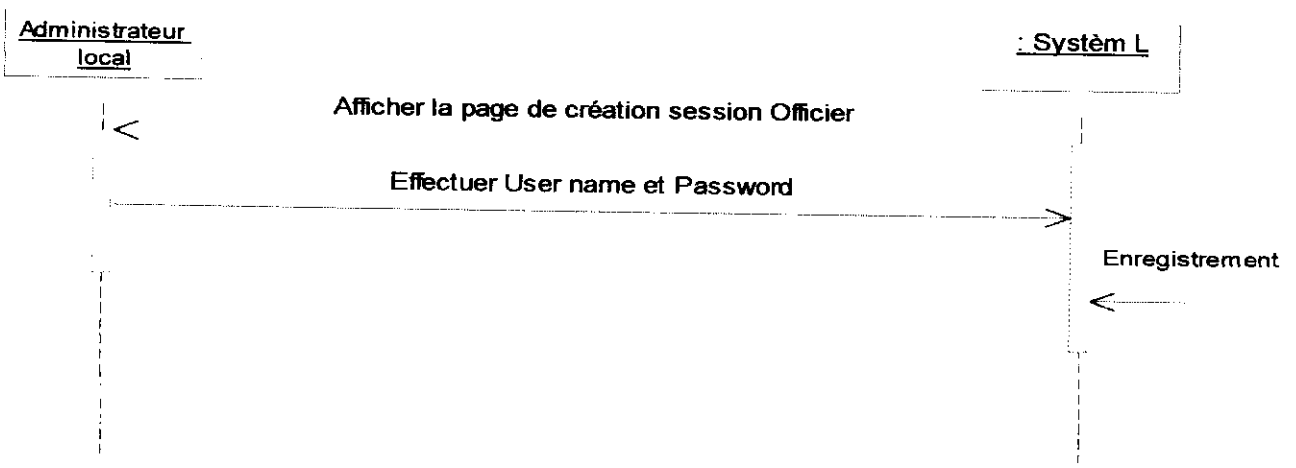


Figure IV.27 : Diagramme de séquence création des session officier.

II.3.14. Authentification officier :

L'officier doit entrer son nom d'utilisateur et son mot de passe, le système local fait une vérification et lui donne l'autorisation.

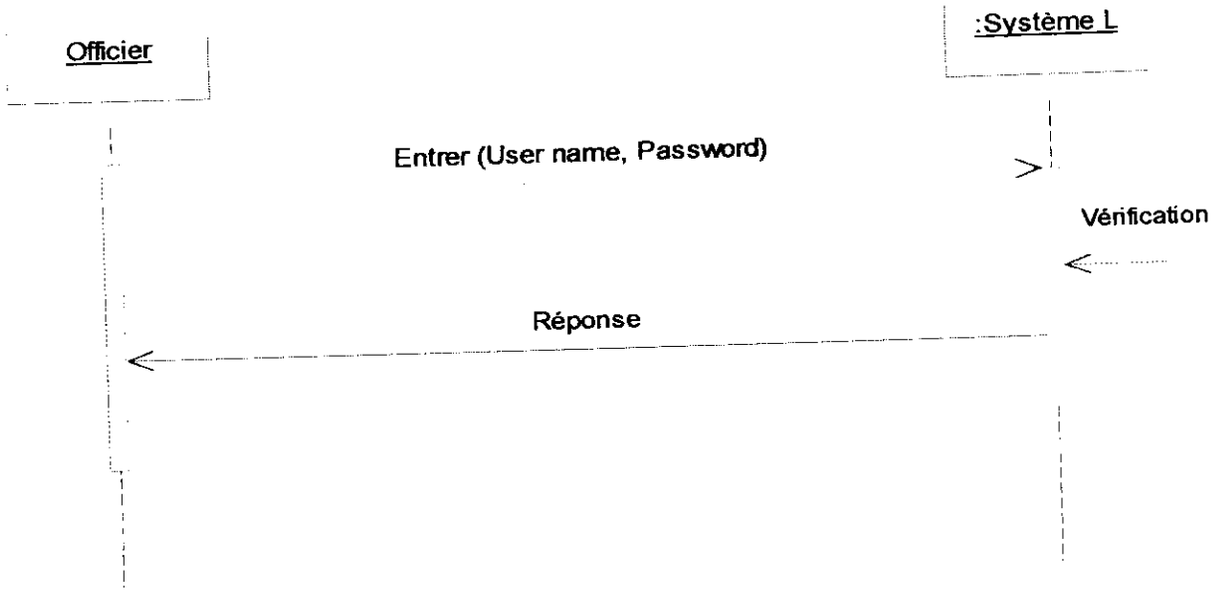


Figure IV.28 : Diagramme de séquence de l'identification d'officier.

II.3.15. Ajout d'un citoyen :

Le système local affiche la fiche citoyen, l'officier choisit l'ajout d'un citoyen, après l'affichage de la page d'ajout, il remplit les informations nécessaires dans la table citoyen.

Le système enregistre ces informations

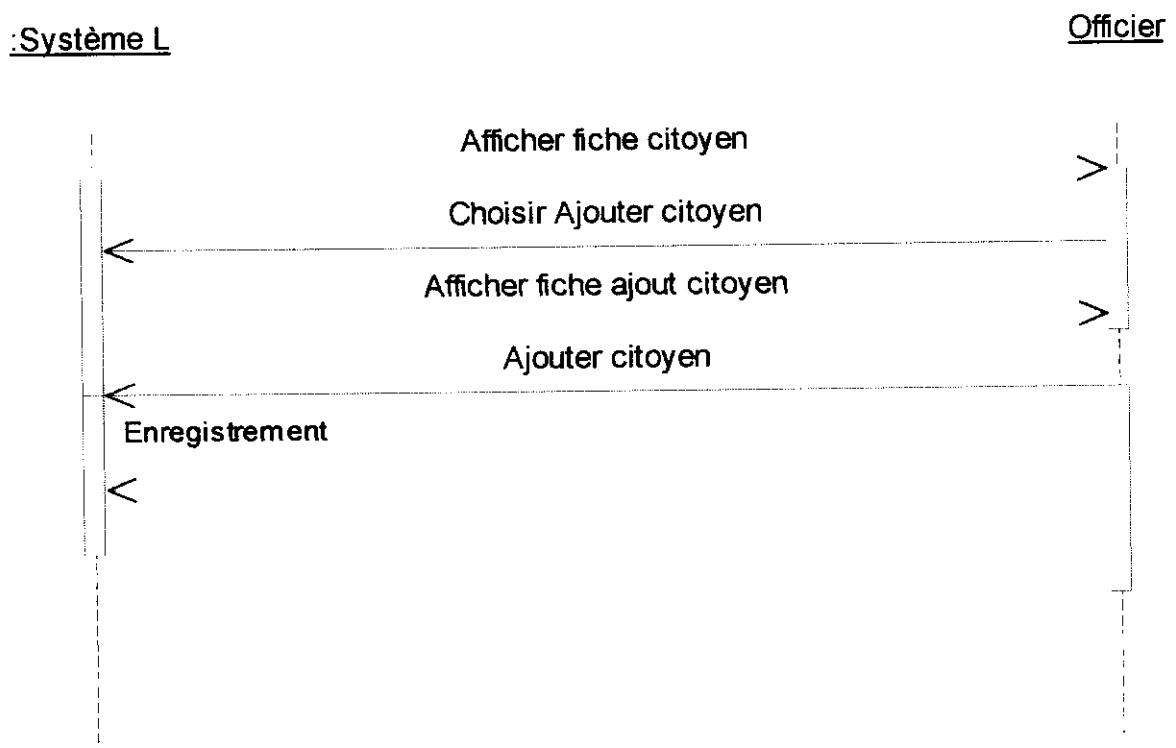


Figure IV.29 : Diagramme de séquence Ajout d'un citoyen.

II.3.16. Suppression d'un citoyen :

Le système local affiche la fiche citoyen, l'administrateur local choisit la suppression, il sélectionne le citoyen, et il effectue la suppression.

Le système enregistre les modifications.

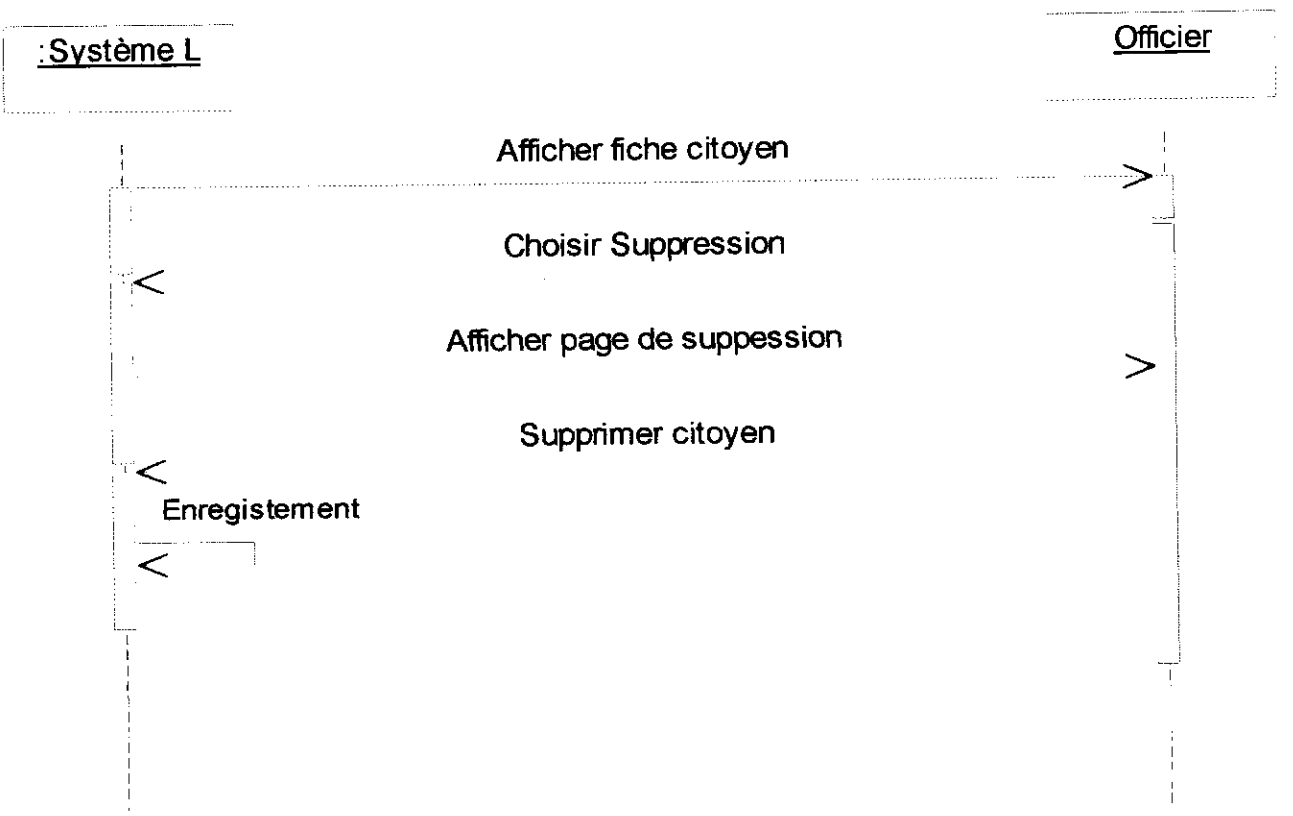


Figure IV.30 : Diagramme de séquence suppression d'un citoyen.

II.3.17. Modifier un citoyen :

Le système local affiche la fiche citoyen, l'administrateur local choisit la modification, il sélectionne le citoyen, et il fait la modification.

Le système enregistre les modifications.

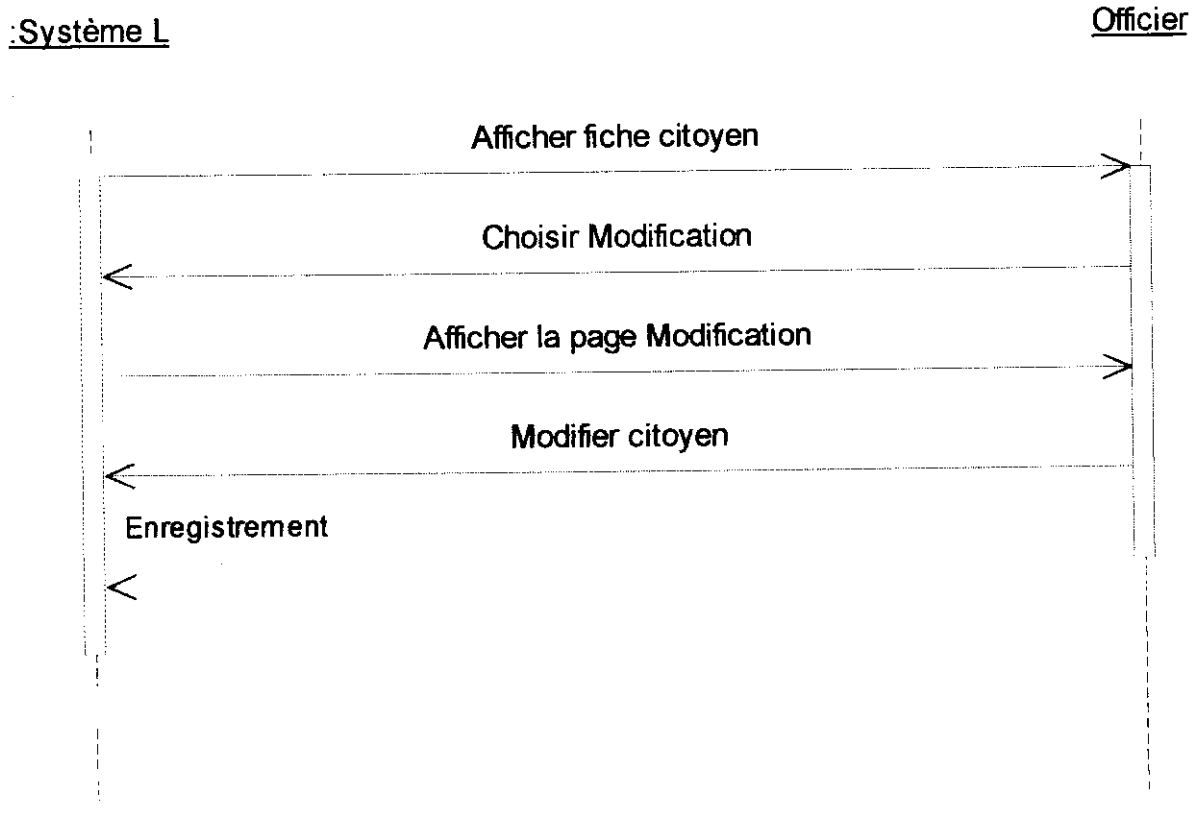


Figure IV.31 : Diagramme de séquence Modifier citoyen.

II.3.18. Ouvrir session citoyen :

Le citoyen se connecte au système et donne son nom d'utilisateur ainsi que son mot de passe, le système local vérifie son identité et autorise la connexion.

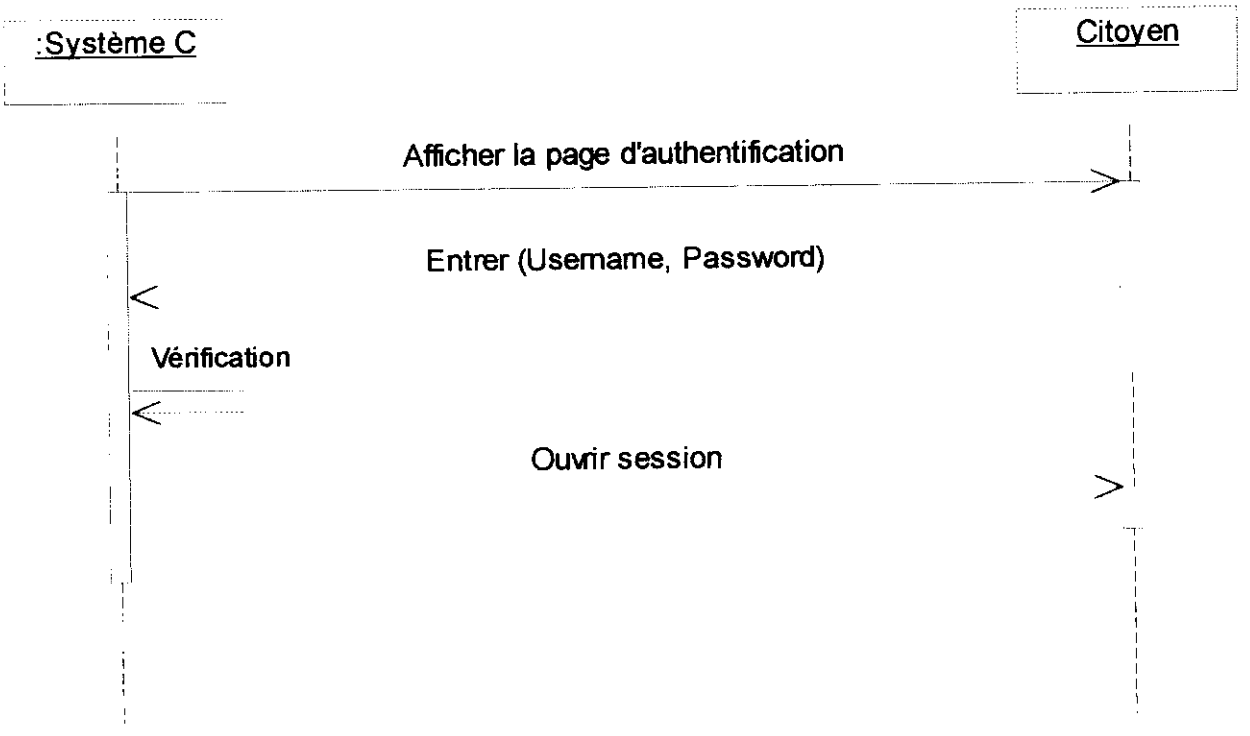


Figure IV.32 : Diagramme de séquence Ouvrir session citoyen.

II.3.19. Sélection / génération d'un document :

Le système central affiche la liste des documents au citoyen, le citoyen sélectionne le document souhaité, le système central fait une recherche, si le document demandé existe déjà sur le serveur FTP il lui donne directement le lien « hypertexte » pour le télécharger sinon le system génère le document en suite il affiche le lien .

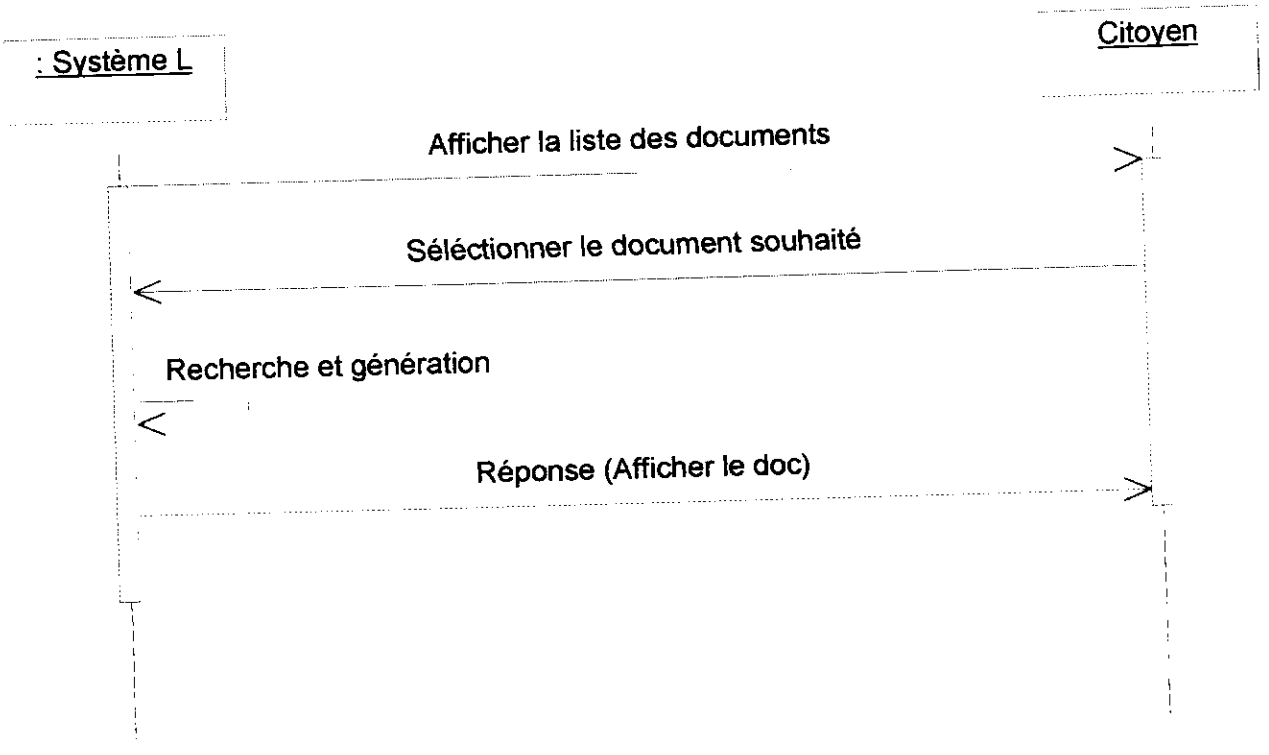


Figure IV.34 : Diagramme de séquence Sélection d'un document.

II.3.20. Vérification et validation :

Le système central affiche la page de la vérification et validation, le citoyen ou bien un autre utilisateur sélectionne l'APC concernée et le document à vérifier, ainsi que la signature du document et les envoie au système central, le système fait une recherche et une vérification et affiche au citoyen le résultat de la vérification du document (valide ou non).

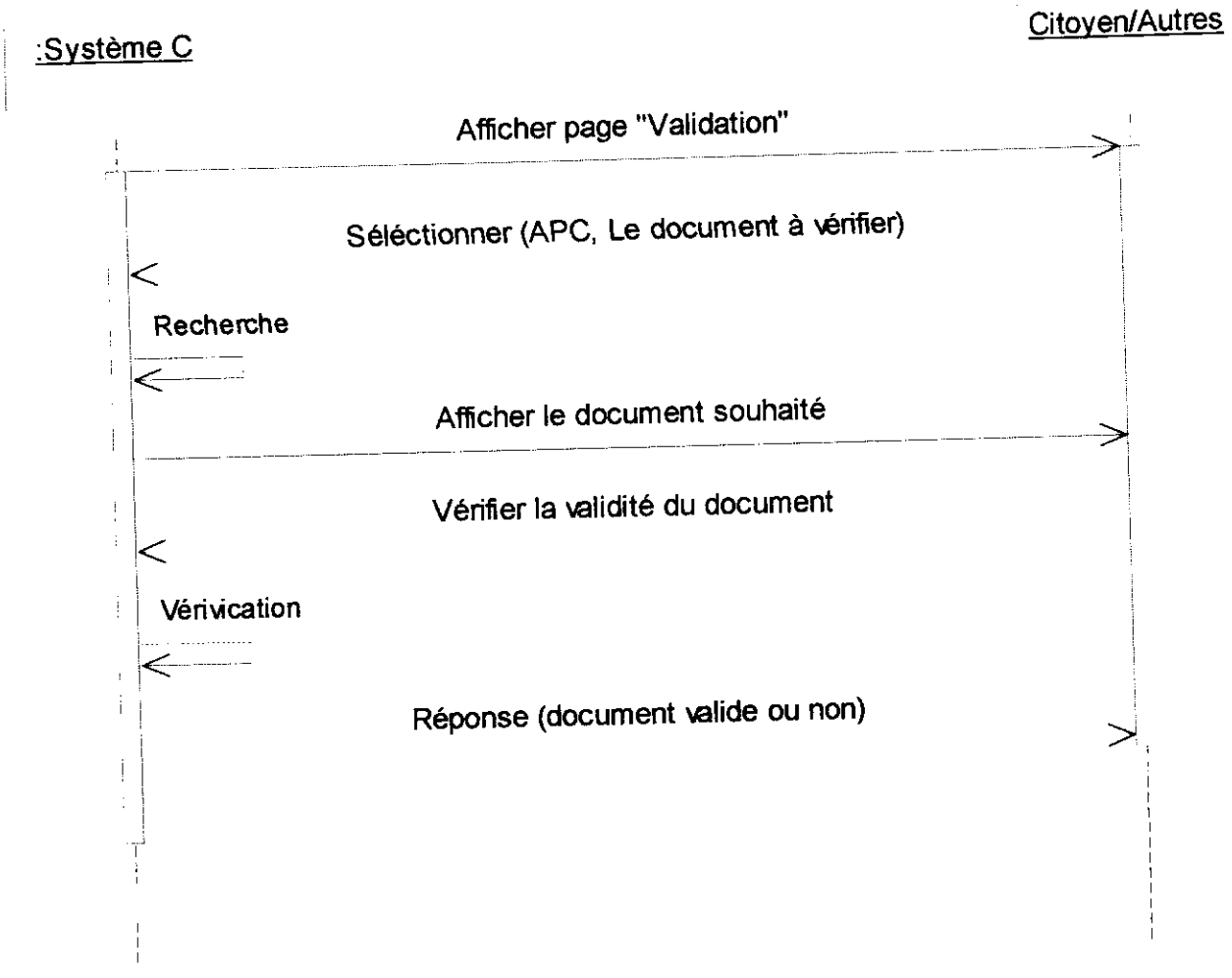


Figure IV.35 : Diagramme de séquence Vérification et Validation.

II.4. Diagrammes d'états-transitions :

II.4.1. Cryptage :

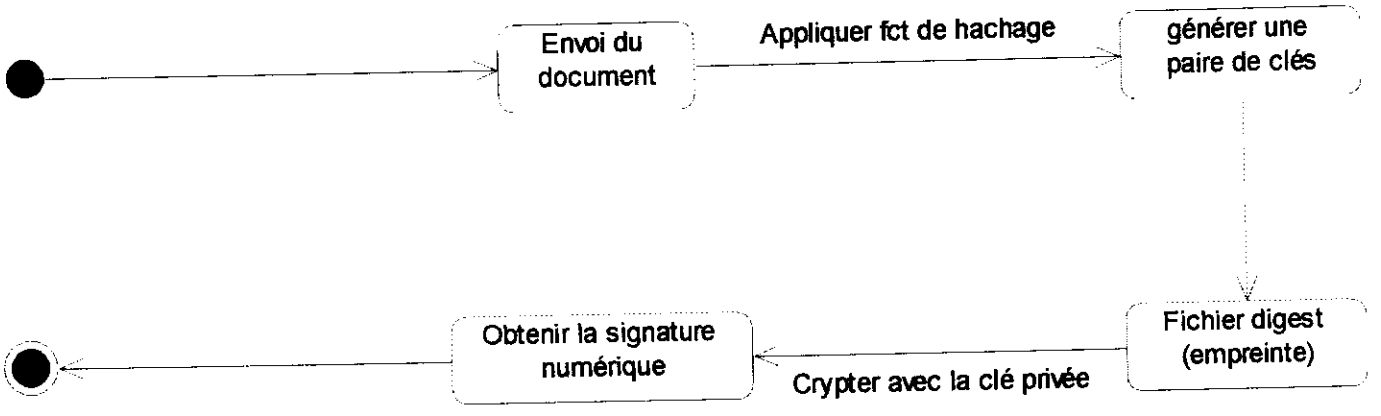


Figure IV.36 : Diagramme d'états-transitions de l'opération de cryptage.

II.4.2. Décryptage :

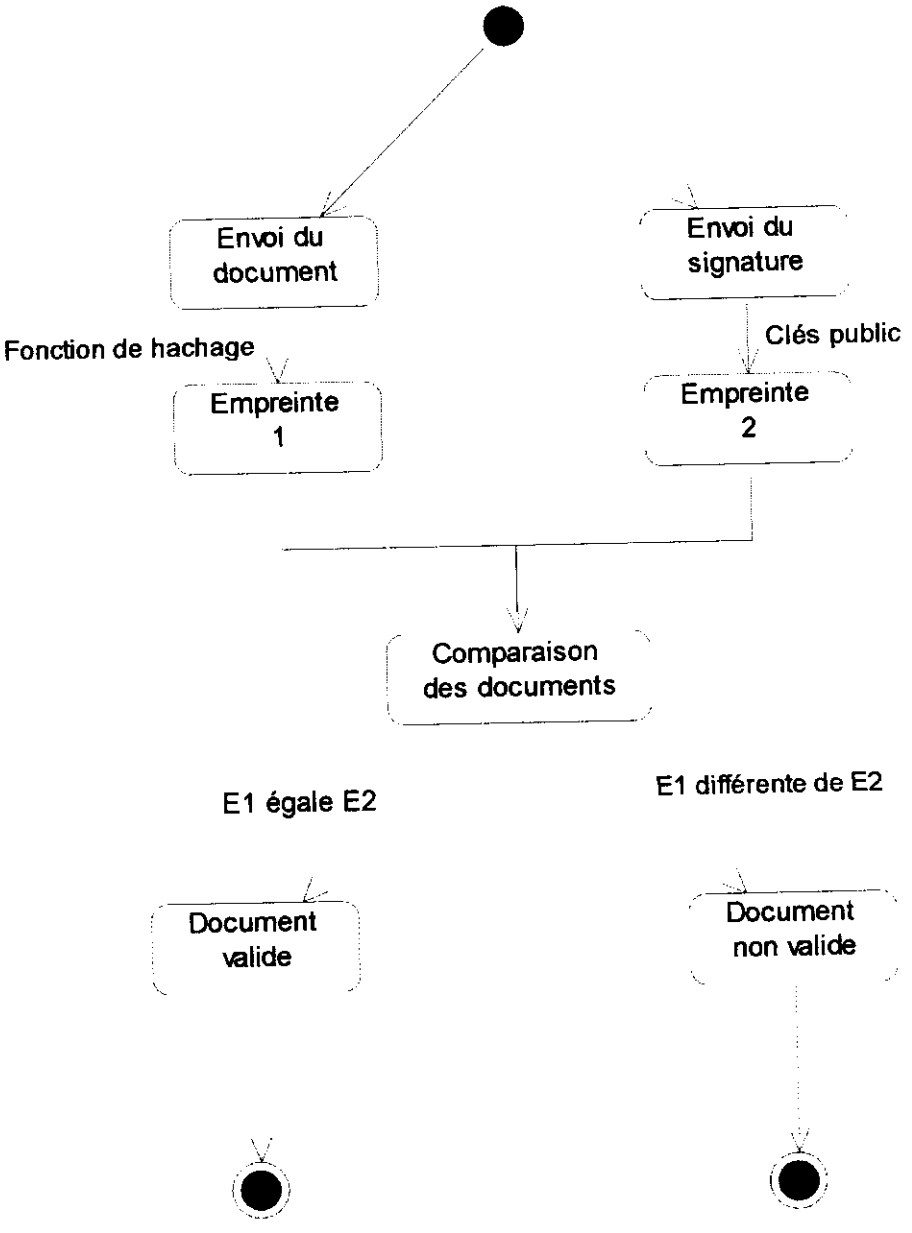


Figure IV.37 : Diagramme d'états/transitions de l'opération de décryptage.

II.5. Diagrammes de classes :

II.5.1.1. Diagramme de classe System central :

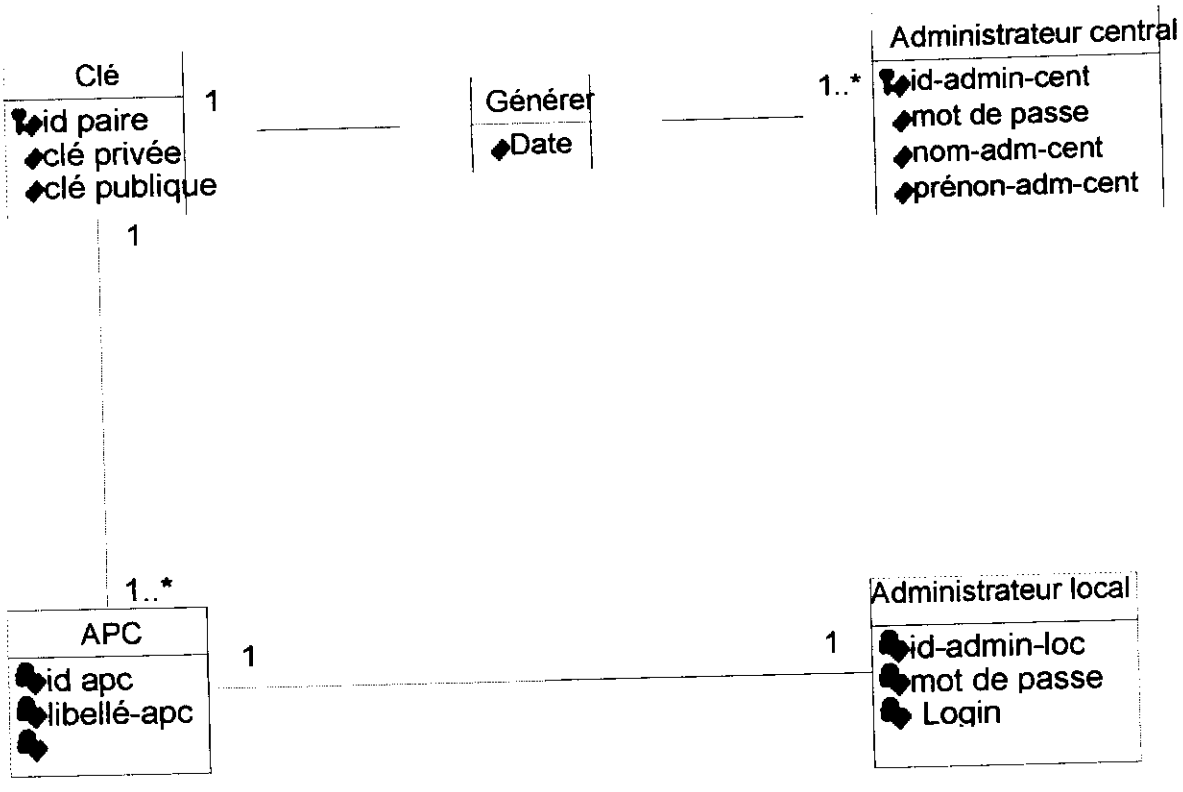


Figure IV.38 : Diagramme de classes du système central.

II.5.1.2. Description des attributs « central »

classes	Codes attribut	attributs
Clés	num_paire	Numéro de paire de clé
	Cle_prive	La clé privé
	Cle_public	La clé Public
Administrateur	Login	login
	Mot de passe	Mot de passe
	Nom	nom
	prénom	Prénom
APC	Libelle_APC	Libellé de l'apc
	Id_admi_local	Identifiant de l'administrateur local
	Login_local	Login de l'administrateur local
	Password_local	Mot de passe de l'administrateur local

II.5.1.3. Modèle relationnel correspondant :

Clé (id_paire, clé privée, clé publique, id_admin_cent, date).

Administrateur central (id_admin, mot de passe, nom_admin_cent, prénom_admin_cent).

APC (id_apc, libellé, mot de passe).

II.5.2.1. Diagramme de classes System Local:

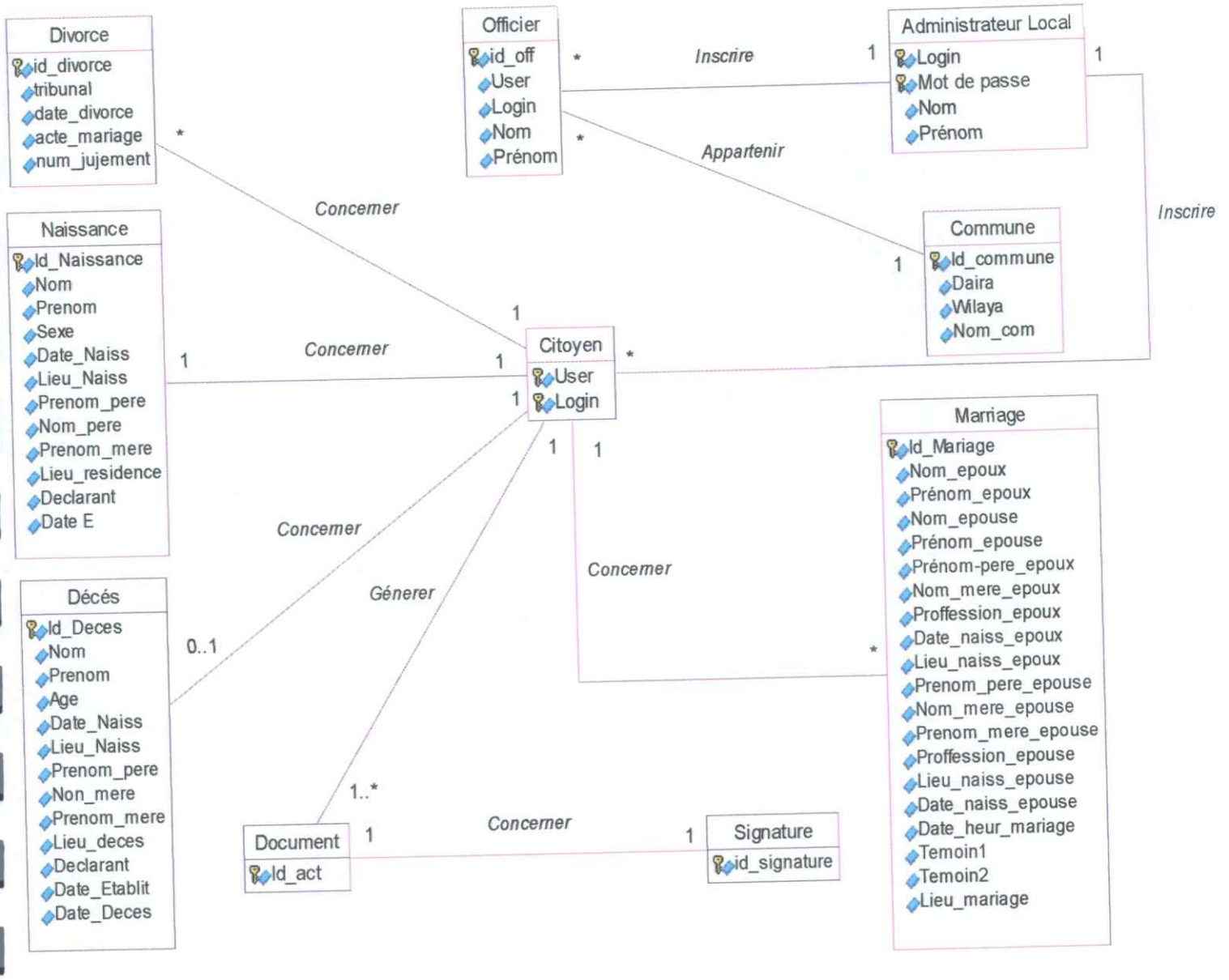


Figure IV.39 : Diagramme de classe du système local.

II.5.2.2. description des attributs « local »

Classes	Codes attribut	Attributs
Décès	Id_Deces Nom Prenom age Date_Naiss Lieu_Naiss Prenom_pere Nom_mere Prenom_mere Lieu_Deces Declarant Date_Etablis Date_Dece	Identifiant de Décès Nom Prénom age Date de Naissance Lieu de Naissance Prénom de père Nom de mère Prénom mère Lieu de Décès Déclarant Date d'établissement Date de Décès
mariage	Id_Mariage Nom_epoux Prenom_epoux Nom_epouse Prenom_epouse Prenom_pere_epouse Nom_mere_epoux Proffesion_epoux Date_naiss_epoux Lieu_naiss_epoux Prenom_pere_epouse Nom_mere_epouse Prenom_mere_epouse Profession_epouse Lieu_naiss_epouse Date_naiss_epouse Date_heur_mariage Temoin1 Temoin2 Lieu_mariage	Identifiant de Mariage Nom de l'époux Prénom de l'époux Nom de l'épouse Prénom de l'épouse Prénom de père de l'épouse Nom de mère de l'époux Profession de l'époux Date de naissance de l'époux Lieu de naissance de l'époux Prénom de père de l'épouse Nom de mère de l'épouse Prénom de mère de l'épouse Profession de l'épouse Lieu de naissance de l'épouse Date de naissance de l'épouse Date et heur de mariage Temoin1 Temoin2 Lieu de mariage
Divorce	Id_Divorce Tribunal Date_Divorce Acte_mariage Num_jujement	Identifiant de Divorce Tribunal Date de Divorce Acte de mariage Numéro de jugement

Classes	Codes attribut	Attributs
Niassance	Id_Naissance Nom Prenom Sex Date_Naiss Lieu_Naiss Prenom_pere Nom_mere Prenom_mere Lieu_residance Declarant Date_E	Identifiant de Naissance Nom Prénom Sexe Date de Naissance Lieu de Naissance Prénom de père Nom mère Prénom de mère Lieu de résidence Déclarant Date d'établissement
Officier	Id_off User Login Nom Prenom	Identifiant de l'officier User Login Nom Prénom
Administrateur local	Login Mot de passe Nom Prénom	Login Mot de passe Nom Prénom
Citoyen	User Login	User Login
Commune	Id_commune daira Wilaya Nom_com	Identifiant de commune daïra Wilaya Commune
Fichier	Id_fich	Identifiant de fishier
Document	Id_Act	Identifiant de l'Acte
Signature	Id_signature	Identifiant de signature

II.5.2.3. Modèle relationnel correspondant :

Citoyen (user, login, id_naissance, id_deces, id_Mariage, id_doc, id divorce).

Naissance (Id_Naissance,Nom,Prenom,Sex,Date_Naiss,Lieu_Naiss,Prenom_pere,Nom_mere, Prenom_mere,Lieu_residance,Declarant,Date_E).

Administrateur local (Login, Mot de passe, NomPrénom, id_citoyen, id_off).

Officier (Id_off, User, Login , Nom, Prenom)

Commune (Id_commune, daira, Wilaya, Nom_com, id_off).

Divorce (Id_Divorce, Tribunal, Date_Divorce, Acte_mariage, Num_jujement).

Décés (Id_Deces, Nom, Prenom, age, Date_Naiss, Lieu_Naiss, Prenom_pere, Nom_mere, Prenom_mere, Lieu_Deces, Declarant, Date_Etablis, Date_Dece).

Document (id_act, id_signature).

Signature (id_signature).

III. Conclusion :

Ayant modélisé notre système, nous avons établi tous les constituants de notre application. Nous présentons dans le chapitre suivant notre implémentation.

Chapitre V. Implémentation

Dans ce chapitre nous allons mettre en œuvre notre stratégie de sécurité que nous avons déjà présenté dans les chapitres précédents et consiste a mettre en place deux systèmes un système « central » pour la gestion des clés de cryptage et la vérification de la validité des documents émis .Et un système que nous appeler « local » pour la génération des différents documents et leurs signer numériquement, en utilisant les clés de cryptage fournies par le système central qui joue de ce faite un rôle semblable d'une autorité de certification.

Nous proposant dans la figure V.1 l'architecture logicielle et matérielle des deux systèmes local et central.

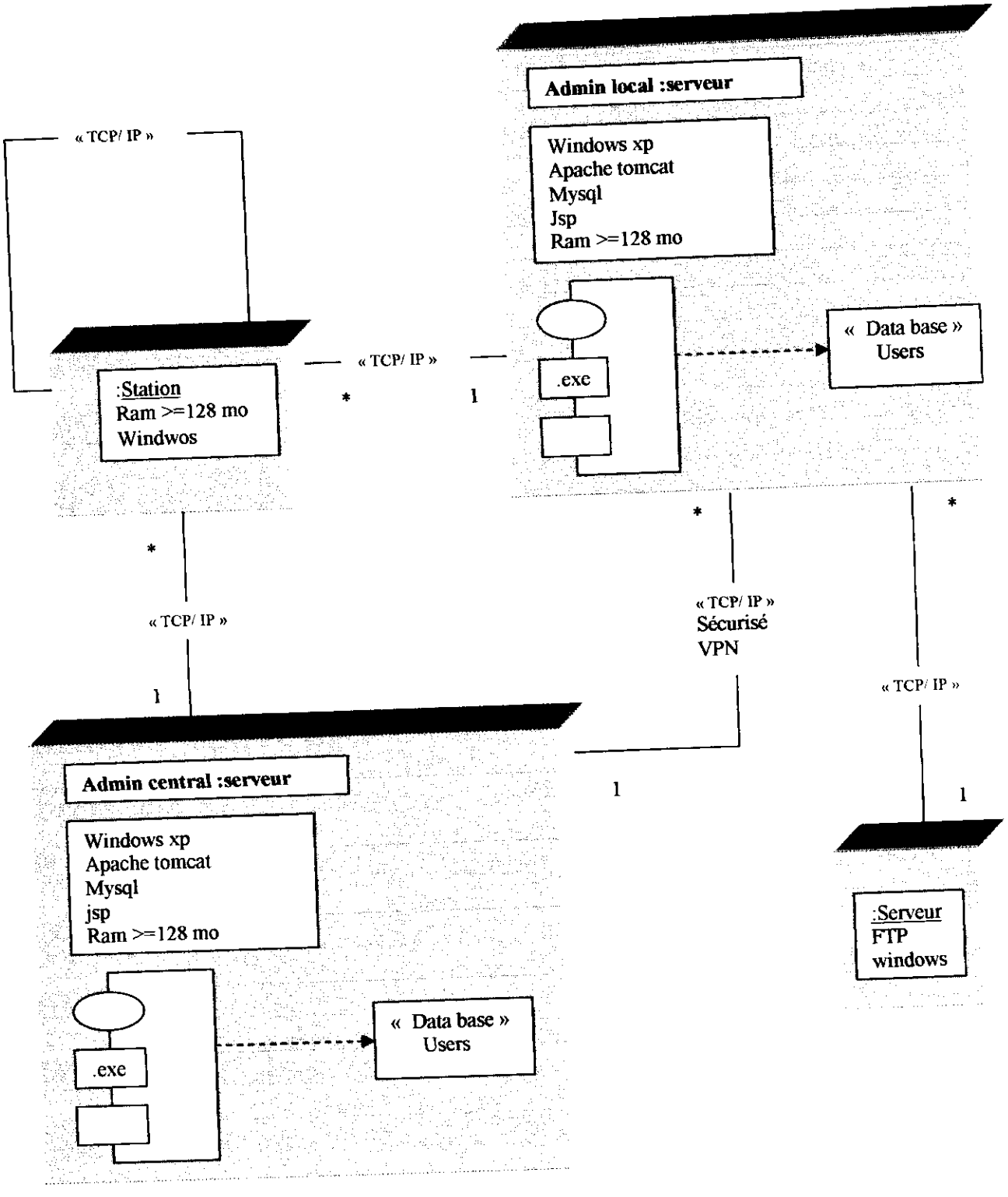


Figure V. 1: Diagramme de déploiement

I. Les outils de développement :

I.1 MySQL:

MySQL est un serveur de base de données relationnelle SQL . **[kof 05]**

MySQL fonctionne sur beaucoup de plates-formes différentes, incluant AIX,HP-UX, Linux , Solaris, SunOS, Windows 95, 98, NT, 2000 et XP. **[Ref 06]**

Les bases de données sont accessibles en utilisant divers les langages de programmation(C, C++, C#, Delphi / Kylix, Java, Perl, PHP) ; une API spécifique est disponible pour chaque langage.

MySQL fait partie du quatuor LAMP: Linux, Apache, MySQL, PHP. Le couple PHP/MySQL étant utilisé sur Internet et proposé par la majorité des hébergeurs.**[Ref05]**

I.2 JavaServer Page :

Java permet de réaliser tous les types d'applications.

- Mais les interfaces utilisateur développées en Java sont relativement peu performantes

- Par contre, Java est très adapté pour réaliser des applications Web :

  - Développer sous Windows, déployer sous UNIX
  - Richesse des programmes et API disponibles
  - Peu d'impacts sur les performances

Le **Java Server Pages** ou **JSP** est une technologie basée sur Java qui permet aux développeurs de générer dynamiquement du code HTML, XML ou tout autre type de page Web. La technologie permet au code Java et à certaines actions prédéfinies d'êtres ajoutés dans un contenu statique.

La syntaxe du JSP ajoute des balises XML, appelées *actions JSP*, qui peuvent être utilisées pour appeler des fonctions. De plus, la technologie permet la création de bibliothèques de balises JSP (*taglib*) qui agissent comme des extensions au HTML ou au XML. Les bibliothèques de balises offrent une méthode indépendante de la plate-forme pour étendre les fonctionnalités d'un serveur HTTP.

Les JSP sont compilées par un compilateur JSP pour devenir des servlets Java. Un compilateur JSP peut générer un servlet Java en code source Java qui peut à son tour être compilé par le compilateur Java, ou peut générer le pseudo code Java interprétable directement. Dans les deux cas, il est bon de comprendre comment le compilateur JSP transforme la page en servlet Java. **[Ref 05]**

I.3 JasperReport :

JasperReports (version 1.1.0) est un outil (librairie) Open Source puissant utilisé Pour la génération d'états. Il permet de créer des rapports à partir de fichiers XML. Le résultat Peut être affiché à l'écran, imprimé ou stocké dans des fichiers au format PDF, HTML, XLS, CSV ou XML.

JasperReports est entièrement développé en Java et peut être intégré dans une gamme très variée d'applications Java (y compris les applications J2EE). Son objectif principal est de fournir un moyen simple et flexible pour la génération de documents.

La librairie JasperReports a été conçue en 2001 par *Teodor Danciu*, qui a également participé nombreux autres projets Open Source (Hibernate, framework Avalon, ...). Il continue régulièrement à proposer de nouvelles évolutions pour JasperReports. Cependant, il n'imaginait pas un tel succès (plus de 300 000 téléchargements et 11 000 nouveaux téléchargements mois).**[Ref08]**

I.4 Tomcat :

Serveur d'application Java permettant d'exécuter des servlets et des pages serveur Java (JSP). Il est développé sous licence open-source par la fondation Apache. Il peut être utilisé ou couplé avec un serveur Web (dont Apache), et porté sur n'importe quel système sur lequel une machine virtuelle Java est installée**[Ref09]**

I.5 FileZilla Serveur FTP :

FileZilla Server permet de mettre en place un serveur FTP (File Transfert Protocol) de manière simple et rapide.

Parmi les fonctionnalités qu'il propose, nous retiendrons :

- La gestion des utilisateurs et des groupes d'utilisateurs
- La sauvegarde sécurisée des identifiants et des mots de passe

- Le filtre contre les attaques extérieures
- La création d'un rapport de fonctionnement
- L'icône présente dans la barre des tâches de Windows qui permet de vérifier l'état de fonctionnement du serveur, de l'activer ou de le désactiver

- Ce logiciel gratuit et sous licence GNU / GPL va de paire avec FileZilla, client du même nom qui permet quant à lui de se connecter à un serveur FTP. Notons au passage qu'il ne fonctionne qu'avec la seule plate-forme Windows. **[Ref10]**

II. Implémentation de Système :

Dans ce qui suit nous allons présenter l'implémentation de notre Système qui est composé de deux partis : système central qui joue un rôle similaire a une autorité de certification en générant des clés de cryptage pour chaque APC, et un système local qui utilise ces clés pour la signature digitale des documents livrés ; chacun de ces deux système possède une base de données qui contient des tables pour stocker les données nécessaires, et un environnement d'exécution des objets du système.

II.1. Système central :

Ce système est basé essentielement sur deux classes , la class generer et la classe decrypter.

II.1.1. La class « generer » : pour générer les clés de cryptage pour chaque APC et les stocker après dans une base de données.

```
class generer {


        RSAPrivateKey Cleprive;
        RSAPublicKey Clepublic;

        // un constructeur pour générer les pairs des clés

        public generer() throws NoSuchAlgorithmException{
        ..................................  }

        public BigInteger getClepulic(){.....}
        public BigInteger getCleprive(){.....}

        public String getClepulicins(){.....}
        public String getCleprivеins(){return getCleprive().toString();
}
```

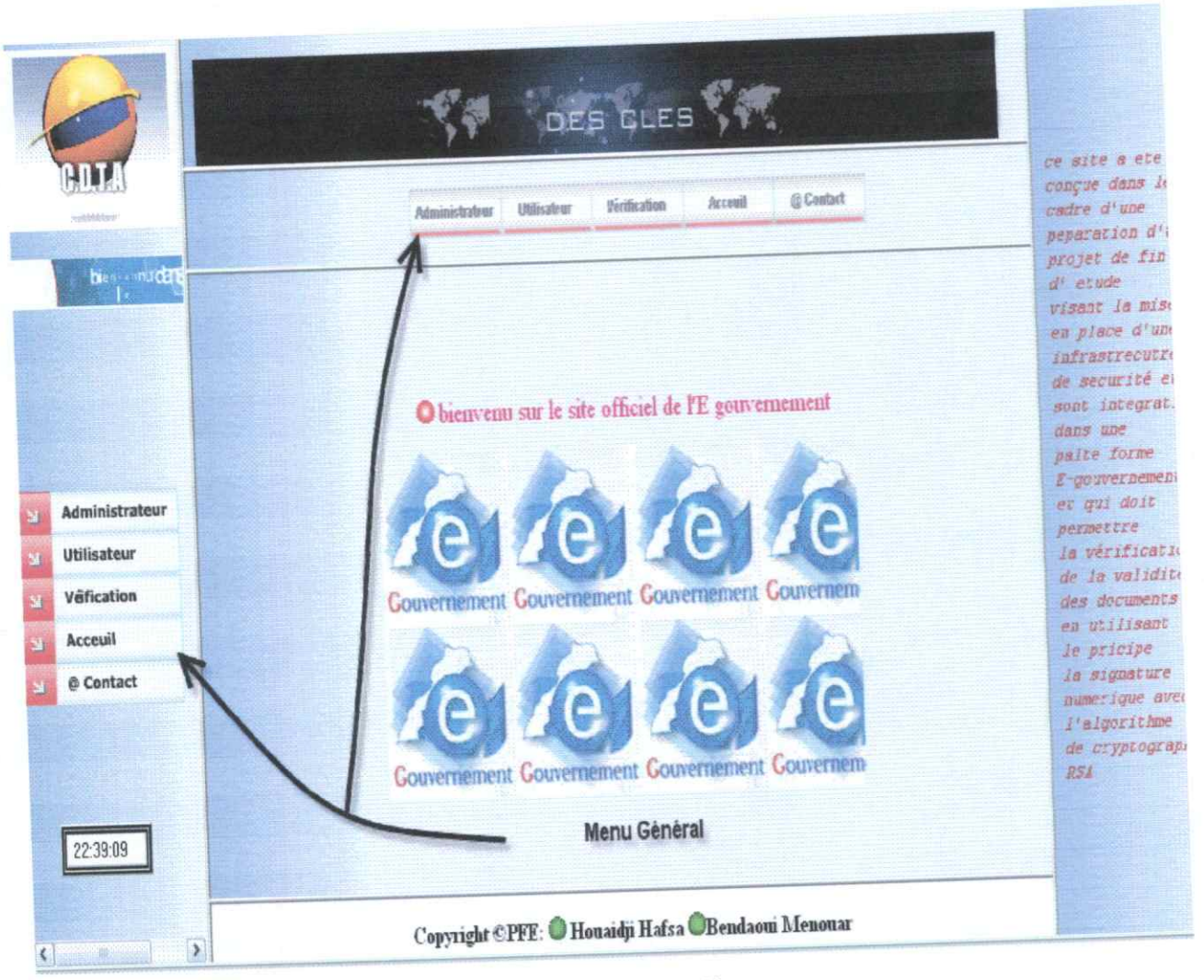
II.1.2. La class « decrypter » : pour la vérification de la validité des documents envoyés sur le serveur de site central, en utilisant le duplicata de la clé publique stocké dans la base de données.

Un message sera affiché d'une manière automatique nous informant sur la validité de document analysé.

```
class decrypter {


public static  RSAPublicKey publicKey=null;
public static  String MessageDeResultat=null;


public decrypter() {.....}
public byte[] getPublicKeyInBytes() {.....}

public void setPublicKey(RSAPublicKey ) { .....}

public void setPublicKey(String ) throws IOException, NoSuchAlgorithmException,
InvalidKeySpecException{..... }

private static byte[] removeOneByte(byte[] ) {  .....};


private static BigInteger decrypt(BigInteger ,RSAPublicKey ) {..... }
//-------------------------

public static    BigInteger RedFilSignature(String) throws IOException{ .....};
//----------------



public          static    byte[] getTableaubyte(String ) throws
IOException{.....}

public static void GenereEmpreinte(String,String )
throws IOException, NoSuchAlgorithmException
{.....}
//-----------
public static  RSAPublicKey getPublicKey() { .....}
public String GetMessage(){.....}


}
```

II.1.3. Réalisation – Système central :

La page d'accueil de site central est représentée ci après, elle est le point d'entrée pour tous les utilisateurs.



La page d'accueil

II.1.3.1 Authentification administrateur :

L'authentification se fait à l'aide d'un formulaire à remplir, ceci nous permet de contrôler l'accès aux tâches d'administrateur. **(Figure V.2)**

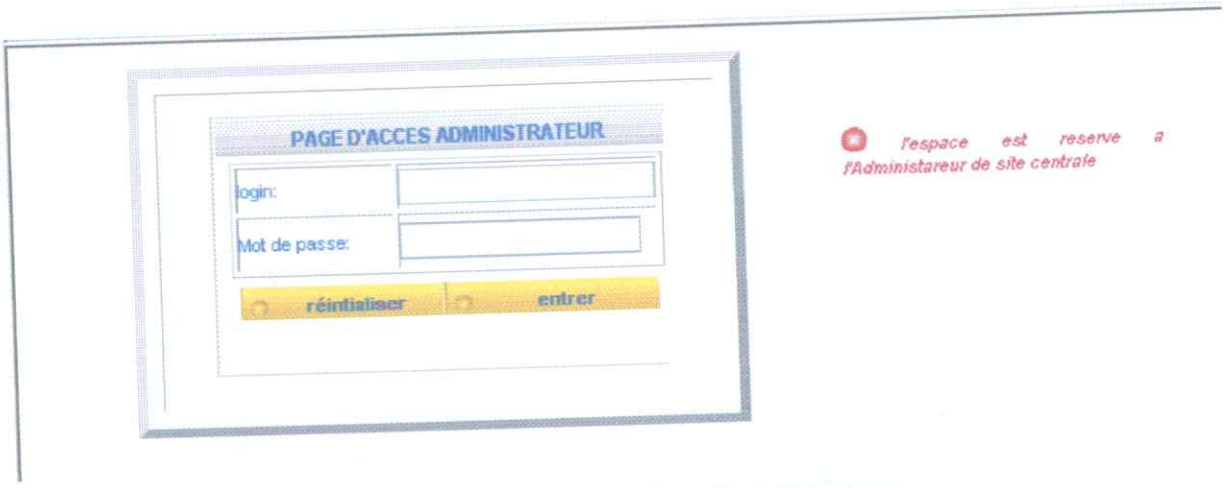


Figure V. 2 : Authentification administrateur

Si l'authentification réussit une page indiquant les différent tâches que peut effectues l'administrateur apparaît :

On doit choisir une APC dans la liste et cliquer sur le bouton « génère une paire », vider la table des clés, ou sur le bouton pour l'ajout ou modification d'une APC.

(FigureV.3)

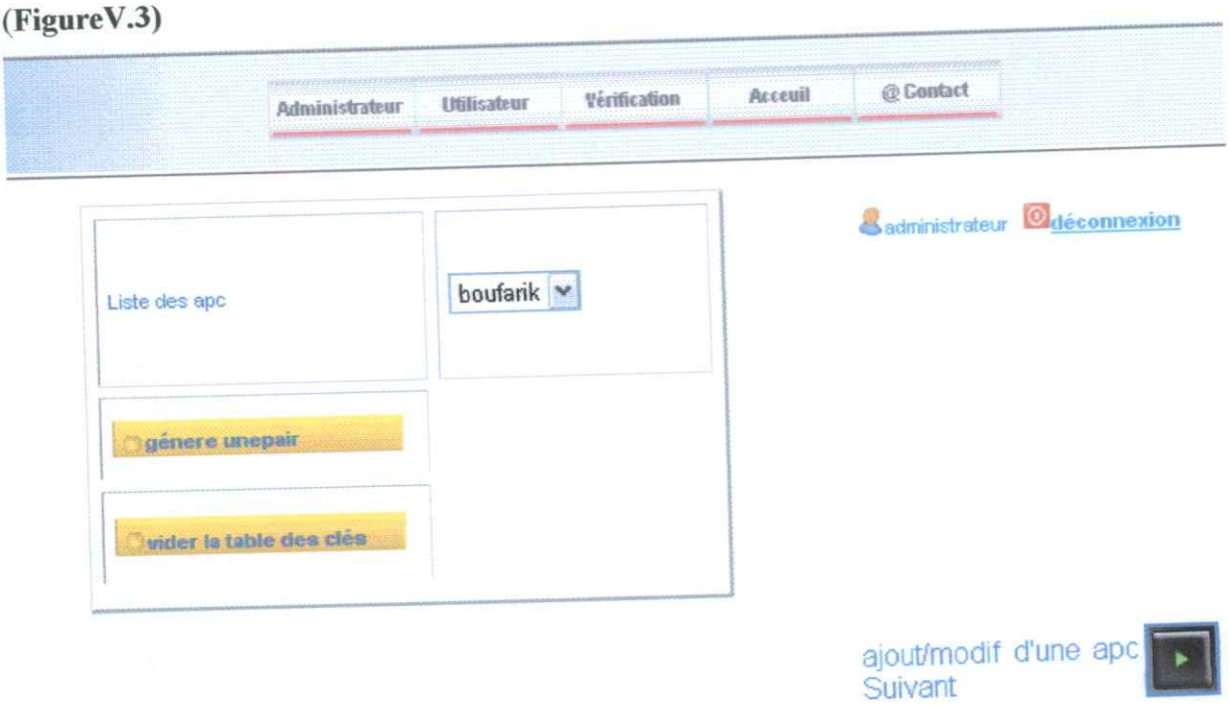


Figure V. 3 : tâches à Réaliser gestion des clés.

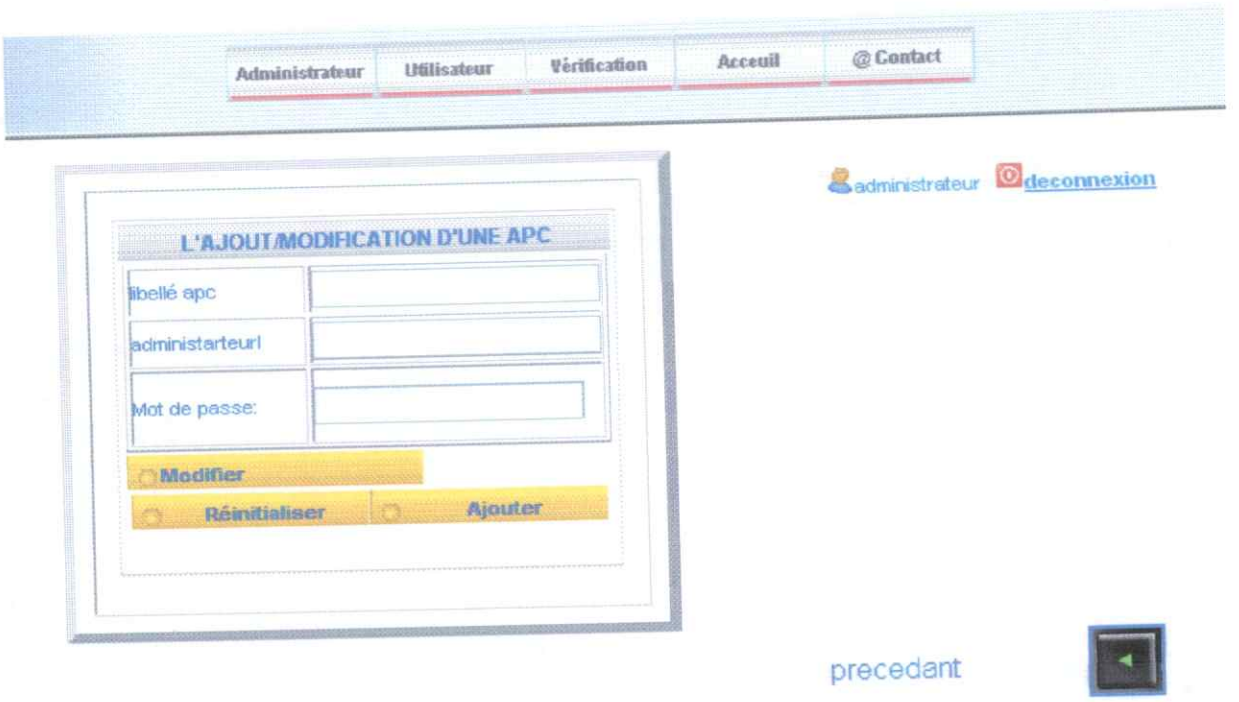


Figure V. 4 : tâches à Réaliser pour une APC.

L'administrateur peut insérer (bouton ajouter) ou modifier (boutant modifier) des APC ainsi que leurs administrateurs locaux et leurs mot de passe

II.1.3.2. Utilisateur APC :

Après avoir cliquer sur utilisateur dans le menu générale, les administrateurs locaux peuvent ouvrir leurs sessions en tapant leurs login et leurs mot de passe fournis par l'administrateur centrale, et en sélectionnant l'APC correspondante, puis valider sur « enter » **(Figure V.5).**

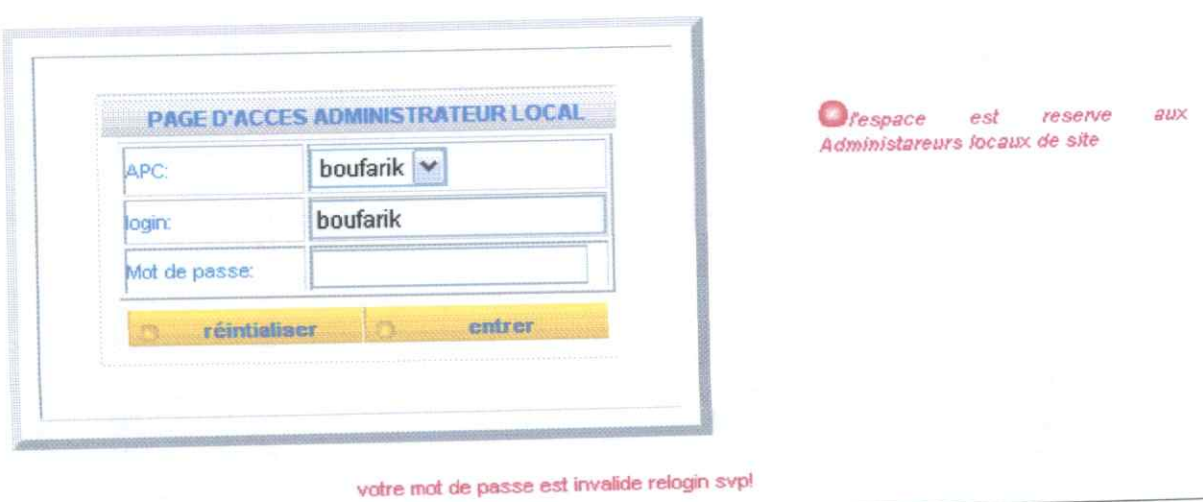


Figure V. 5 : Authentification administrateur local

Après l'ouverture de la session chaque administrateur local peu télécharger les clés de cryptage générer par l'administrateur central pour l'APC correspondante et ceci en cliquant sur l'image pour chaque clé (privé et public) ; le transfert des clés est protégé par un réseau virtuel privé (VPN) . **(FigureV.6).**

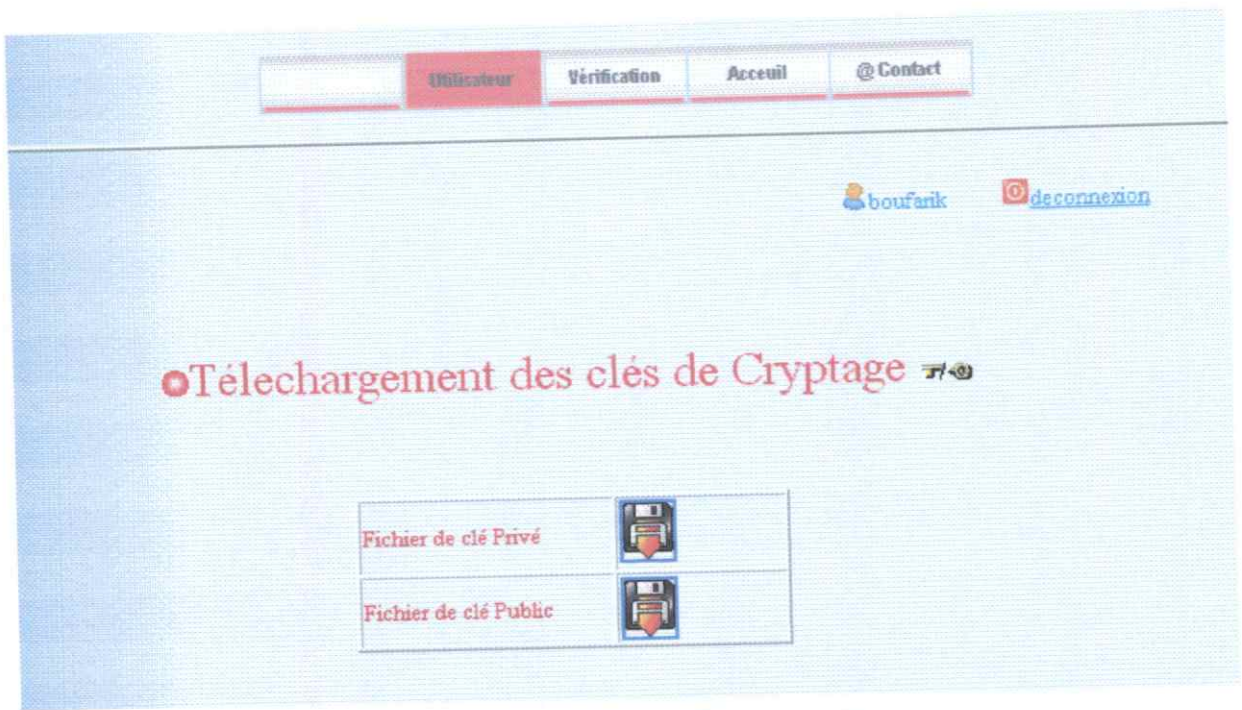


Figure V. 6 : Téléchargement des clés

II.1.3.3. Vérification :

Tout personne possédant un document électronique livré par une APC, et qui veut s'assurer de la validité de ce dernier, peut le faire en sélectionnant « vérification » dans le menu générale .Une page apparaît **(Figure V.7)** où il peut sélectionner l'EAPC et spécifier le document et sa signature, puis envoyer les fichiers sur le serveur en appuyant sur « envoyer vos fichiers ».le document est **(Figure V.8)** ou non **(Figure V.9)**.

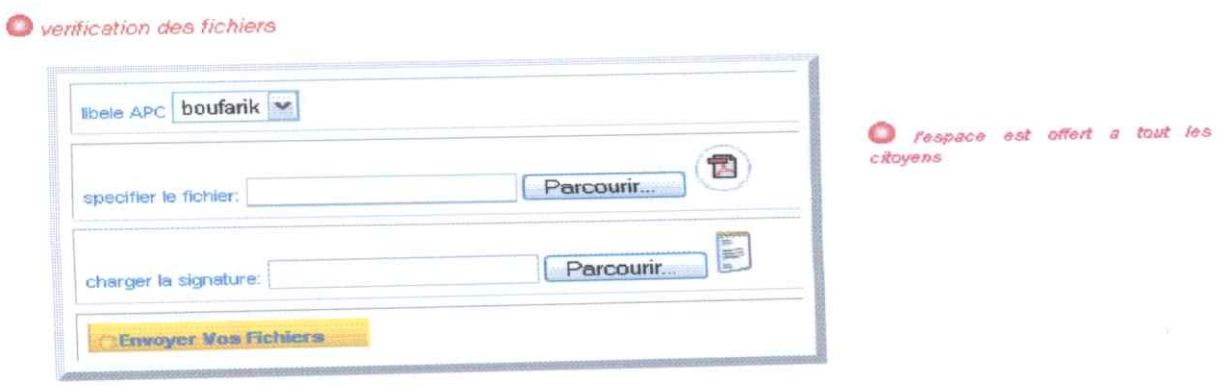


Figure V. 7 :L'envoie des documents au serveur

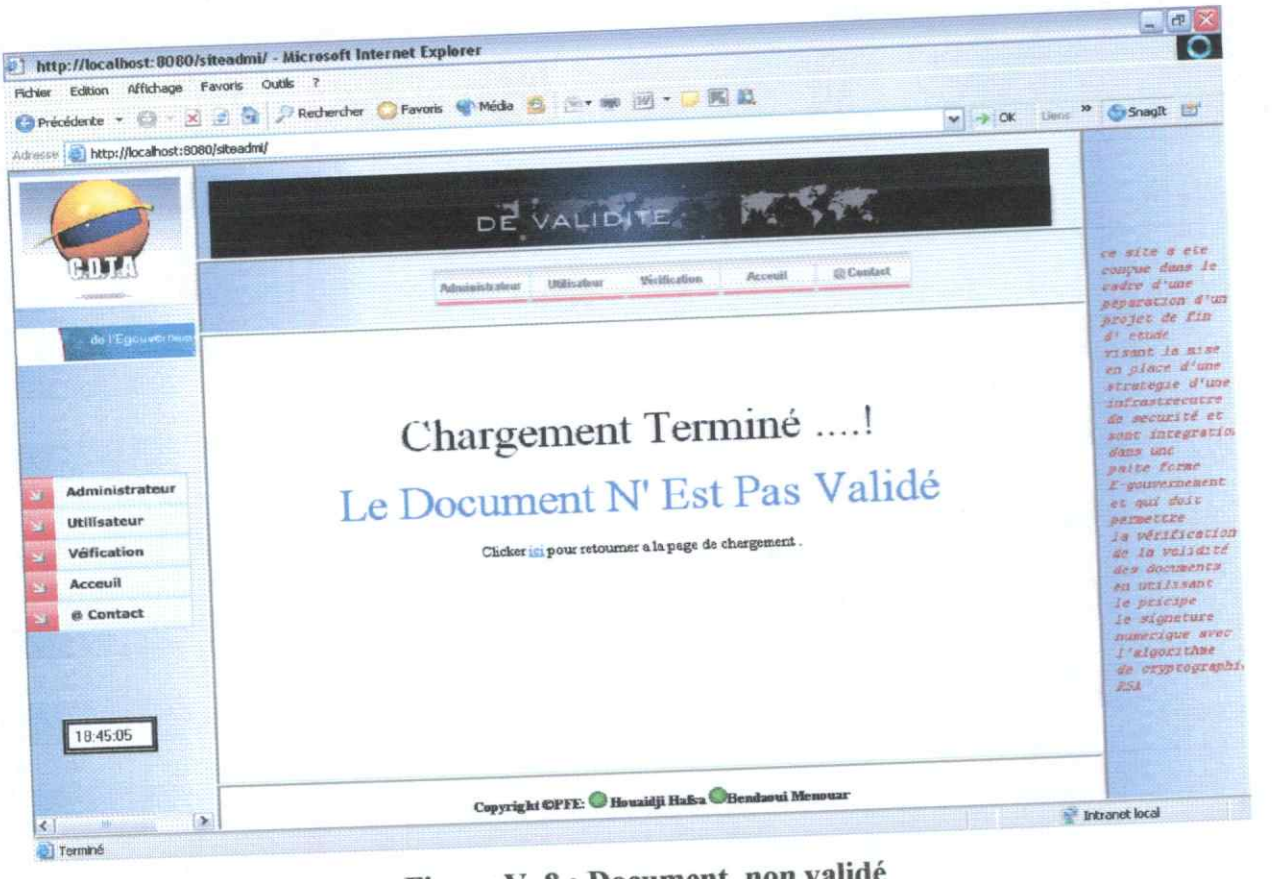


Figure V. 8 : Document non validé

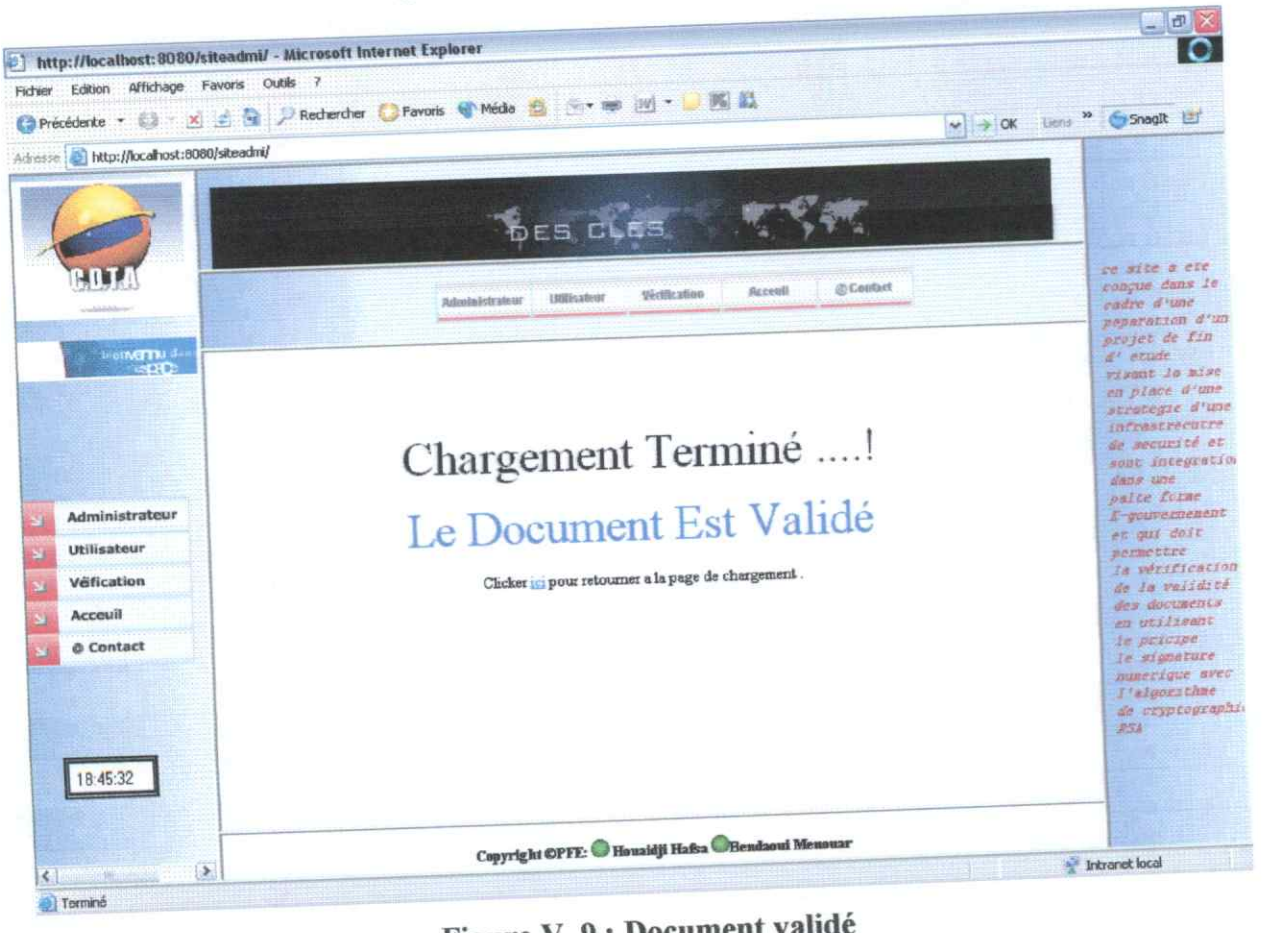


Figure V. 9 : Document validé

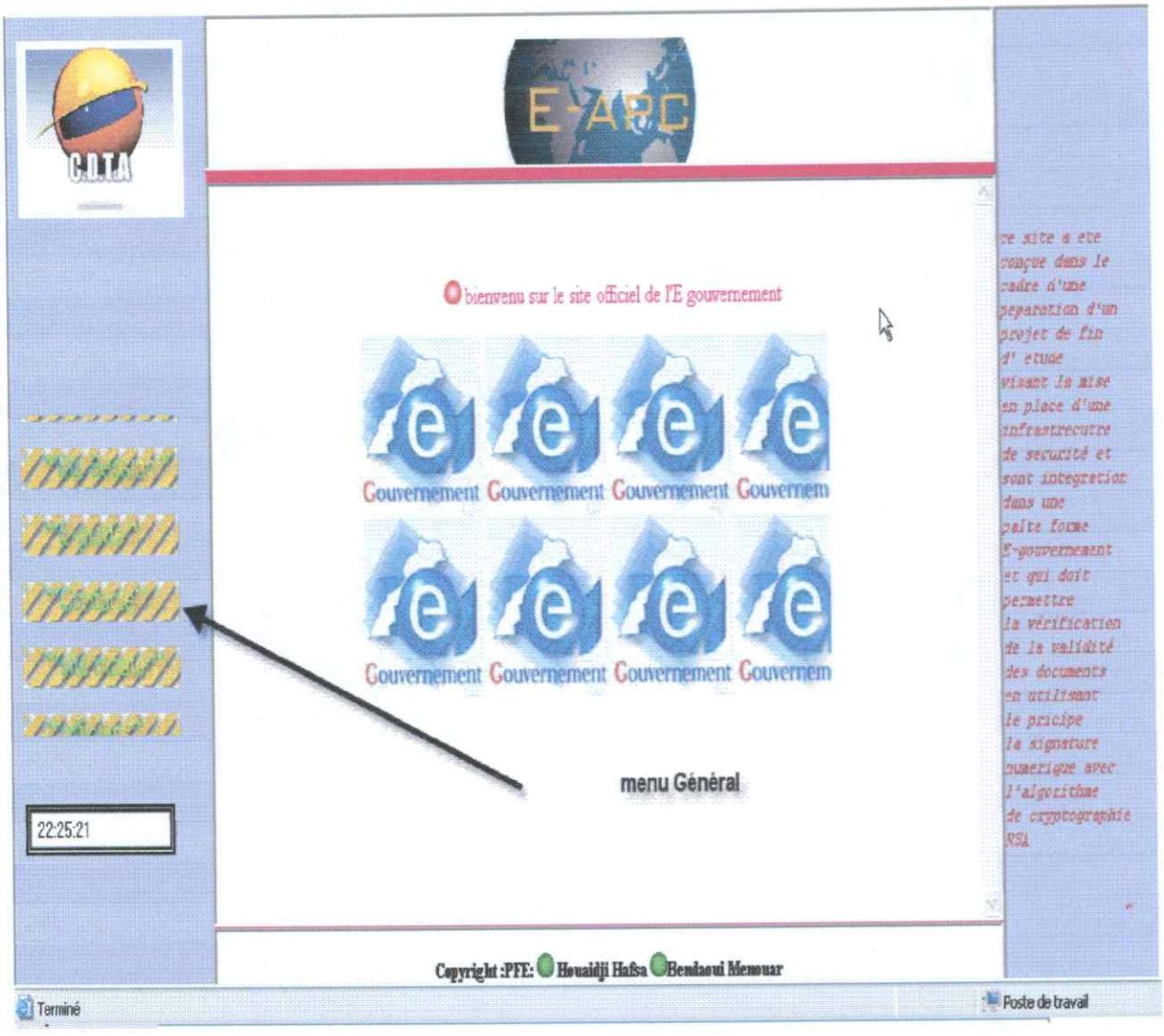
II.2. Système local :

II.2.1 la classe « Crypter » : Ce système est basé essentiellement sur une classe « crypter » qui est chargé de la signature des documents en utilisant le document généré et la clé publique stockée dans la base de donnés (récupérée auprès du système central).

```
public class Crypter {

        public  RSAPublicKey publicKey=null;
       static  public  RSAPrivateKey privateKey=null;

//      ---------------------
       public  Crypter(){.....};
public Crypter(String ){.....}
//      ---------
 public RSAPublicKey getPublicKey() {.....  }

 public byte[] getPublicKeyInBytes() {.....       }

 public RSAPrivateKey getPrivateKey() {.....          }


  public byte[] getPrivateKeyInBytes() {.....    }

 public void setPublicKey(RSAPublicKey ) {.....       }

public void setPublicKey(byte[] ) {.....                }

public void setPrivateKey(RSAPrivateKey ) {.....      }

 public  void  setPrivateKey(String ) {.....}

//      ------------------------------------------------------------
  private static byte[] addOneByte(byte[] ){.....}
//      ------------------------------------------------
static    public  BigInteger crypt(BigInteger , RSAPrivateKey) {.....  }
 //---------------------------------------
public  static    byte[] getTableaubyte(String ) throws IOException{.....   }

  public static void genereFichierSignature(byte[] ,String)throws IOException{..... };

public static void SignerFichier(String ,RSAPrivateKey ,String) throws IOException,
NoSuchAlgorithmException {.....  }

}
```

II.2.2. Réalisation - Système local :

La page d'accueil de site local est présenté ci après .c'est cette page que permet de gérer les APC.



Page D'accueil du site local

II.2.2.1. Administrateur local :

L'authentification de l'administrateur local se fait via un formulaire. **(Figure V.10)**

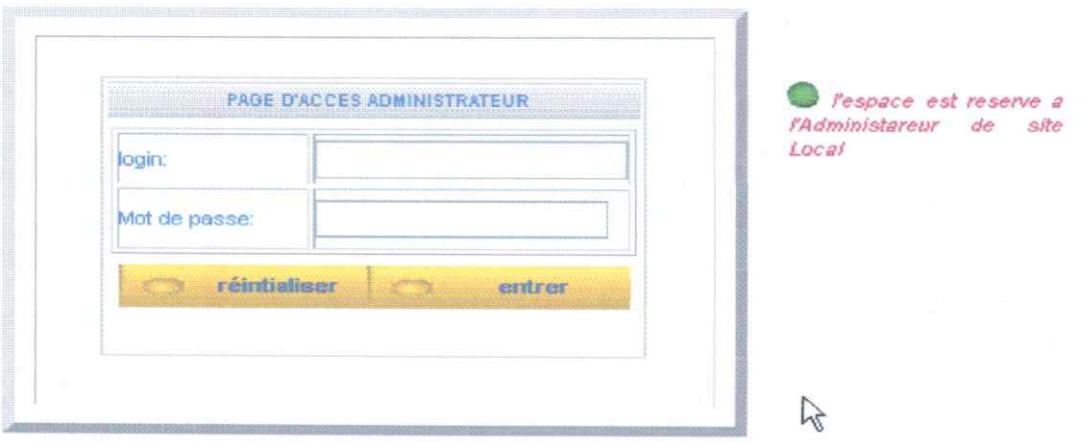


Figure V. 10 : Authentification administrateur

Après l'authentification une page indique les différentes tâches à réaliser tel que : nouveau compte officie, nouveau compte citoyen, mise a jour des clés, mise a jour des citoyens, mise ajour des officiers, mise à jour des investisseurs **(Figure V.11)**

Figure V. 11 : Index administrateur

Après avoir sélectionné l'une de ces option, un formulaire s'affiche. On remplit les différents champs et on clique sur« insérer »pour ajouter un nouveau enregistrement. En cas d'erreurs de frappe on clique sur « Réinitialiser » si non pour retourner à la page index on clique sur « retour ».

Pour le choix « nouveau compte officier » **(Figure V.12) ;**

Pour le choix « nouveau compte citoyen » **(FigureV.13) ;**

Pour le choix « mise à jour des officiers » **(FigureV.14) ;**

Pour le choix « mise à jour des citoyens » **(Figure V.15) ;**

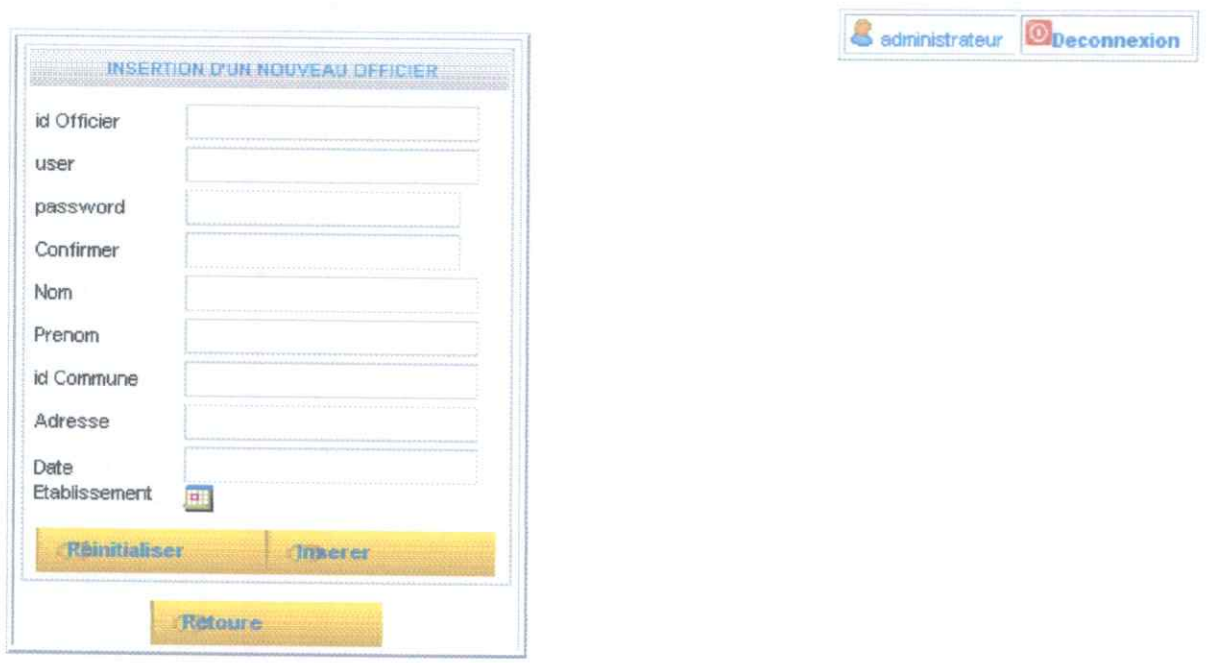


Figure V. 12 : Insertion d'un Nouveau officier

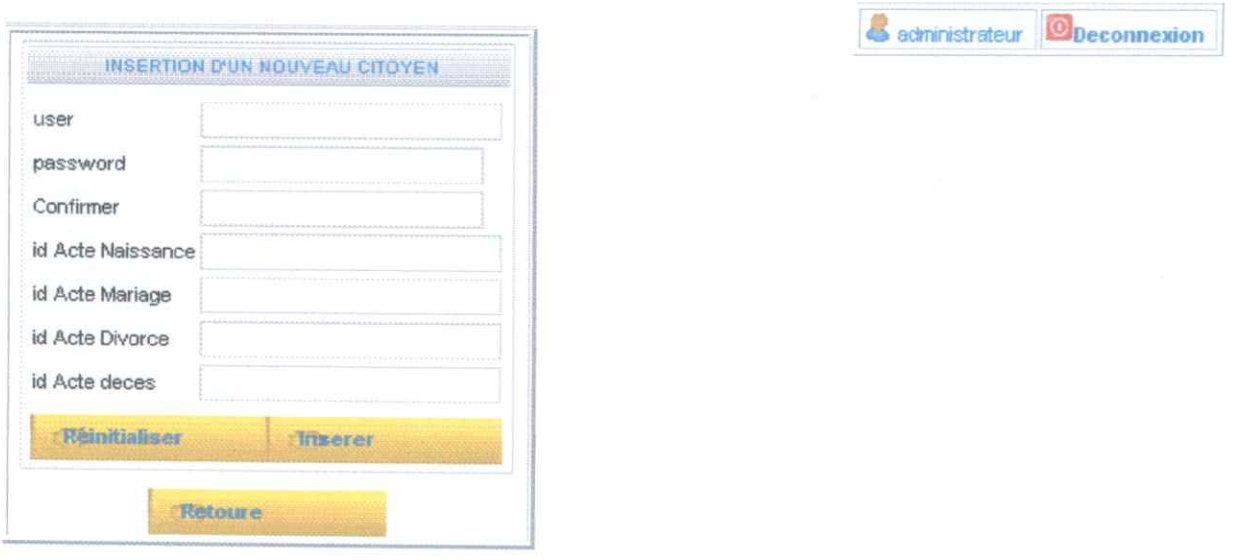


Figure V. 13 : Insertion d'un Nouveau citoyen

Figure V. 14 : Mise à jour des citoyens

Figure V. 15 : Mise à jour des officiers

Pour l'option «mise à jour des clés de cryptage » **(Figure V.16)** on identifie le fichier de la clé privé et celui la clé publique, En suite, on charge les clés dans la base de données en cliquant sur « charger les clés ».

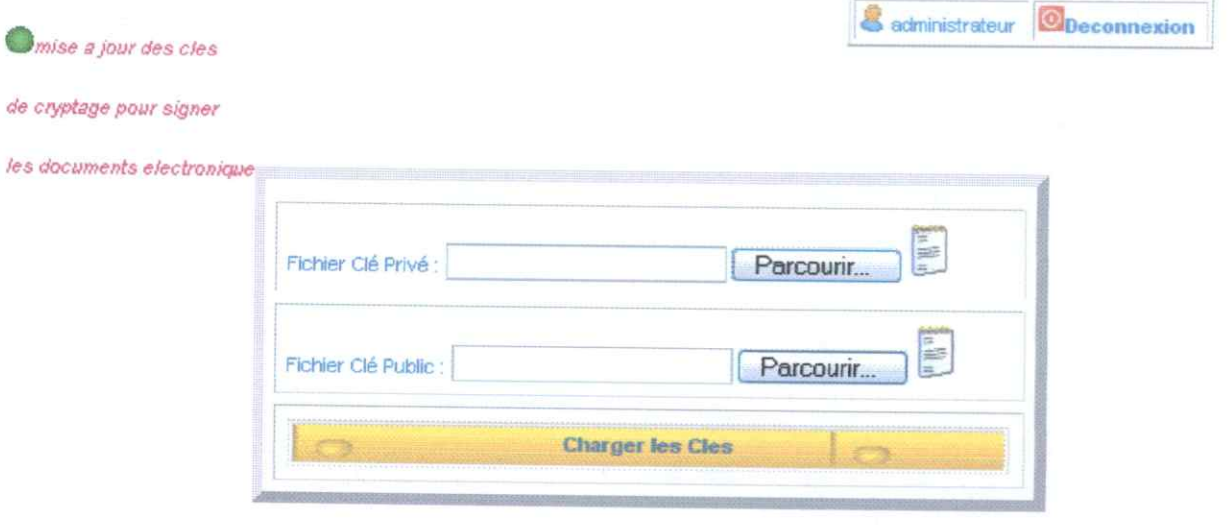


Figure V. 16 : Mise à jour des clés de cryptage

II.2.2.2. Officier:

Après avoir cliquer sur « officier»dans le menu générale chaque officier d'état civil devra s'identifier on tapant son login et son mot de passe **(Figure V.17) ;**

Un menu apparaît indiquant les différentes actions qui peut être accomplies **(FigureV.18) ;**

Après avoir sélectionné l'une de ces options, un formulaire s'affiche. On remplit les différents champs et on clique sur« insérer »pour ajouter un nouveau enregistrement. En cas d'erreurs de frappes,on clique sur « Réinitialiser », sinon pour retourner à la page index, on clique sur « retour ».

Pour le choix « nouveau acte de naissance » **(Figure V.19) ;**

Pour le choix « nouveau citoyen » **(Figure V.20) ;**

Pour le choix « nouveau acte de décès » **(Figure V.21) ;**

Pour le choix « nouveau acte de mariage » **(Figure V.22) ;**

Pour le choix « nouveau acte de divorce » **(Figure V.23) ;**

Et pour la mise à jours des enregistrements on fait d'abord la recherche ;

Pour le choix « mise à jour des actes de naissance» **(Figure V.24) ;**

Pour le choix « mise à jour des actes des décès » **(Figure V.25) ;**

Pour le choix « mise à jour des actes des mariages » **(Figure V.26) ;**

Pour le choix « mise à jour des actes des divorces » **(Figure V.27) ;**

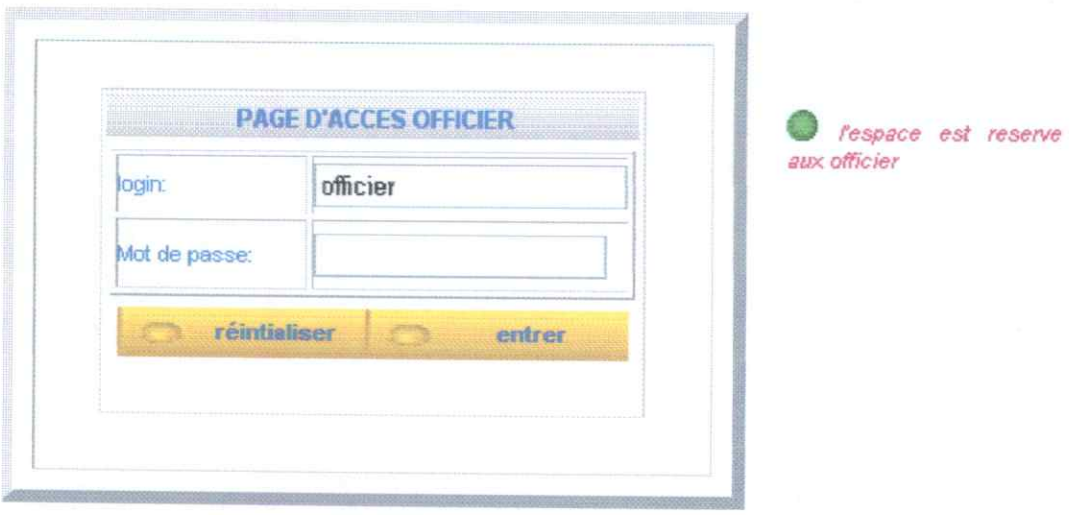


Figure V. 17 : Authentification Officier

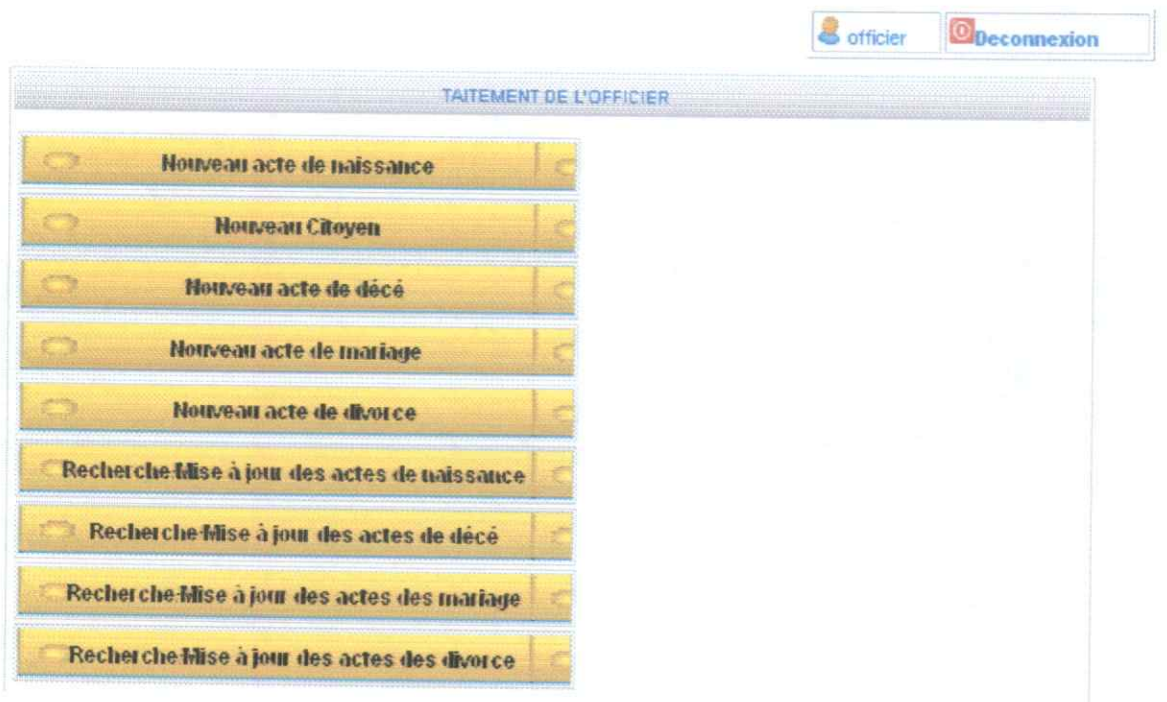


Figure V. 18 : Menu de Traitement Officier

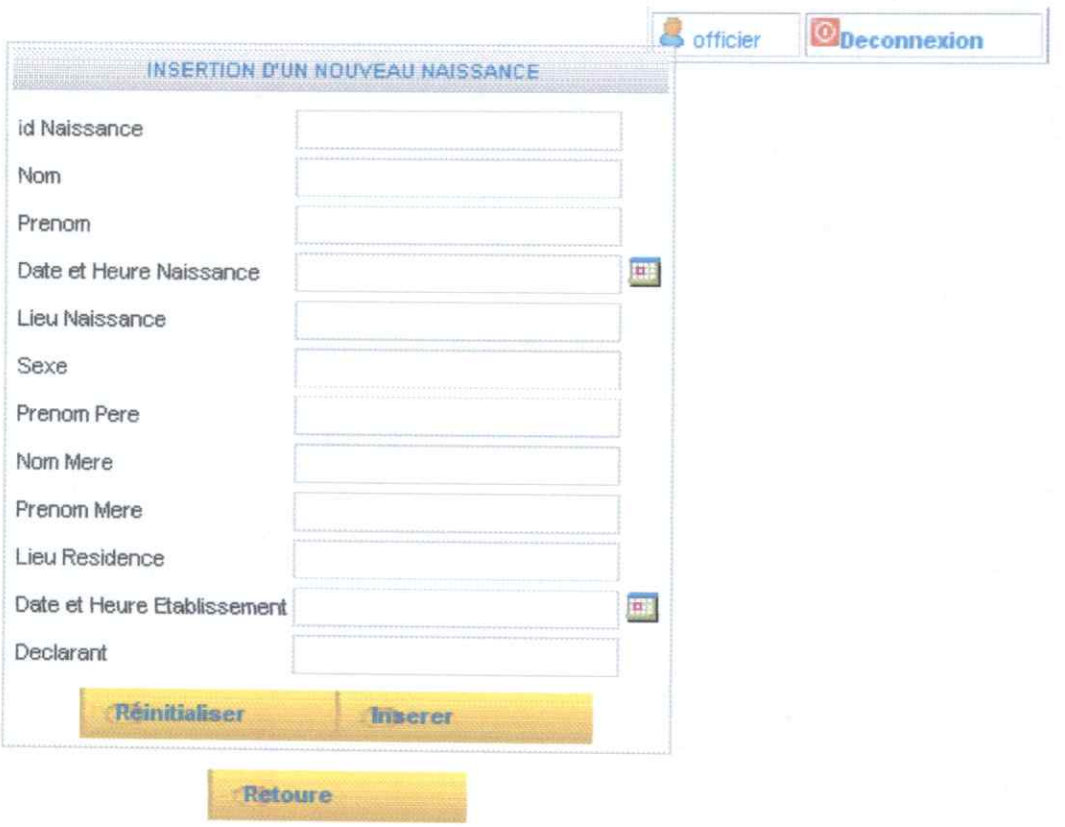


Figure V. 19 : Insertion Nouvel Naissance

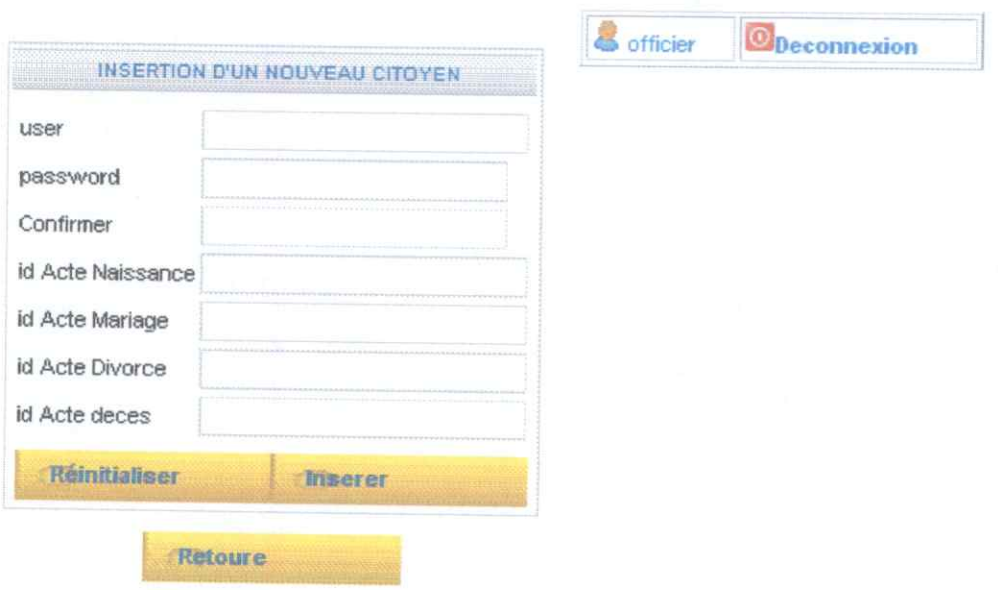


Figure V. 20 : Insertion d'un Nouveau Citoyen

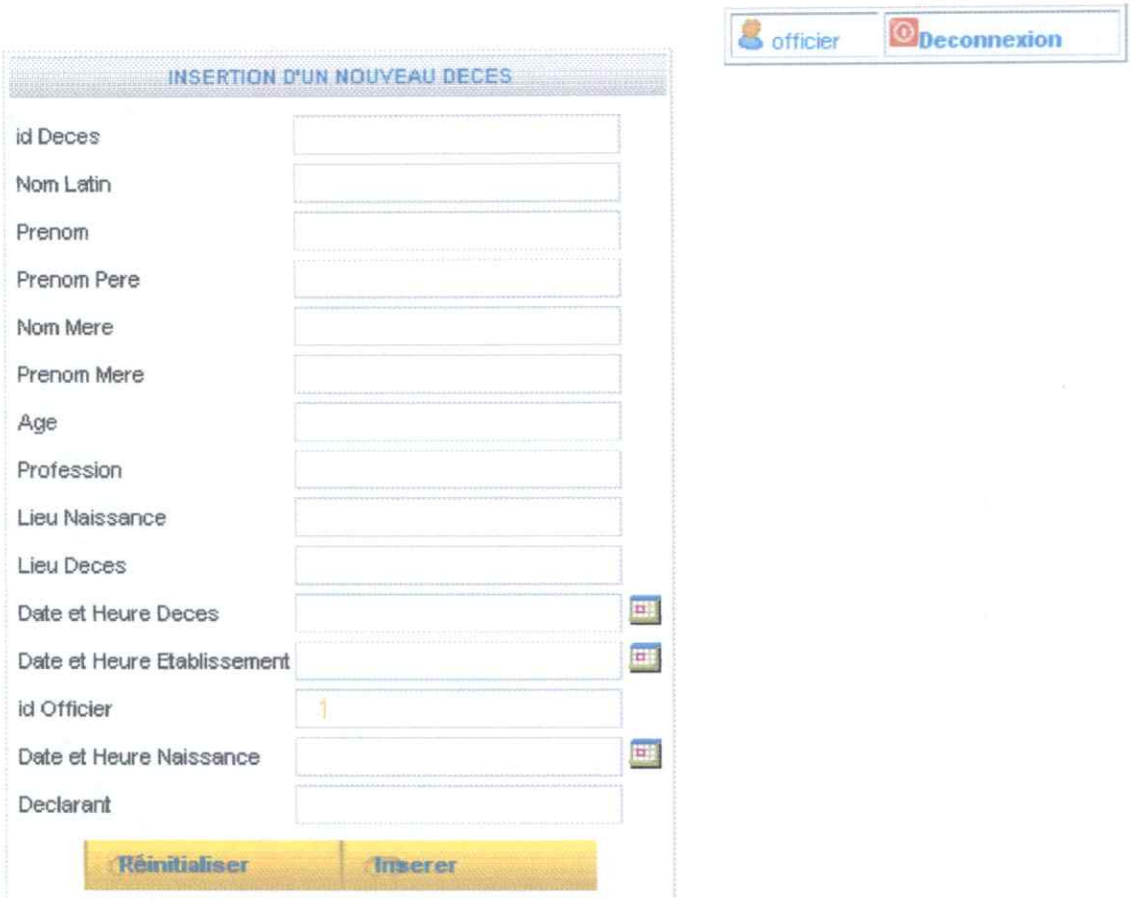


Figure V. 21 : Insertion Nouveau Décès

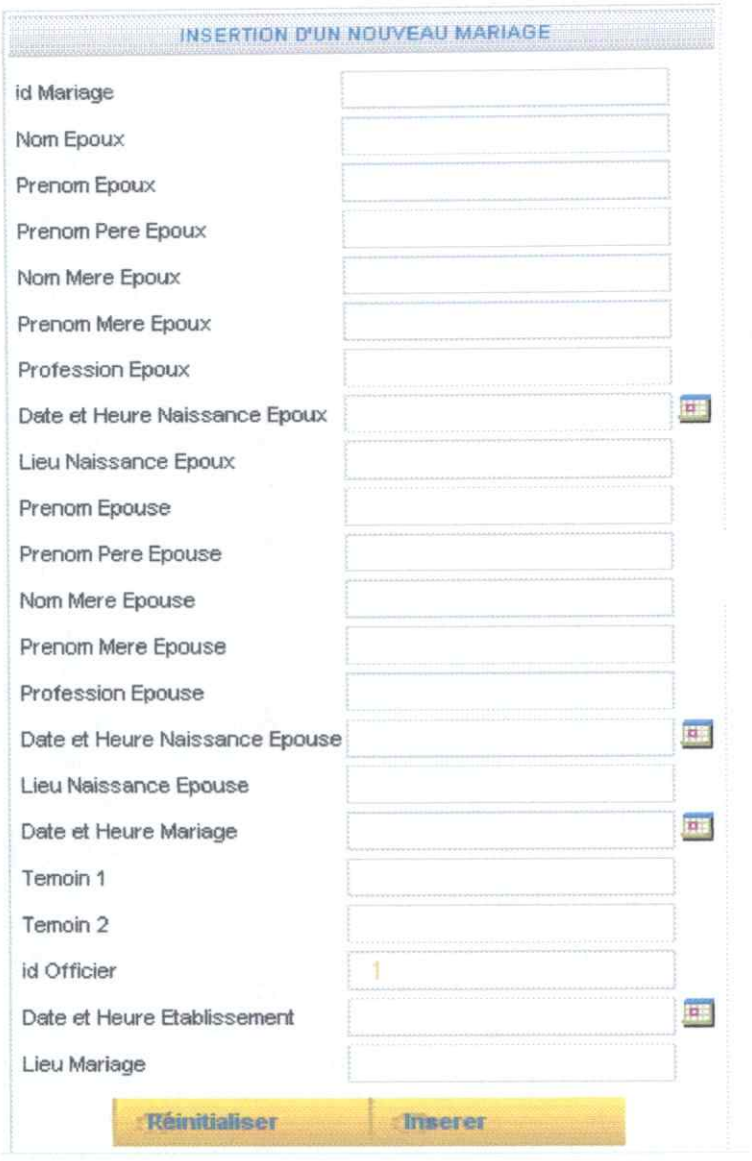


Figure V. 22 : Insertion Nouveau Acte Mariage

Figure V. 23 : Insertion Nouveau Divorce

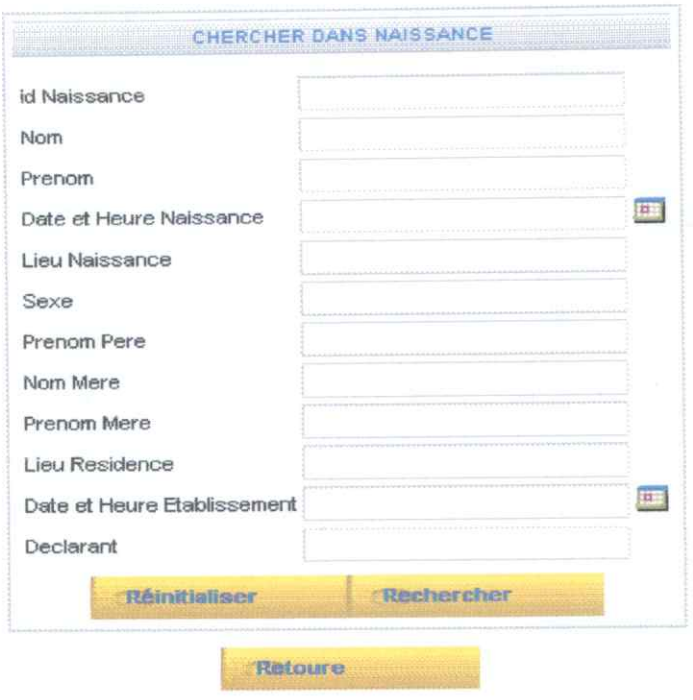


Figure V. 24 : Recherche Mise à Jour Actes Divorces

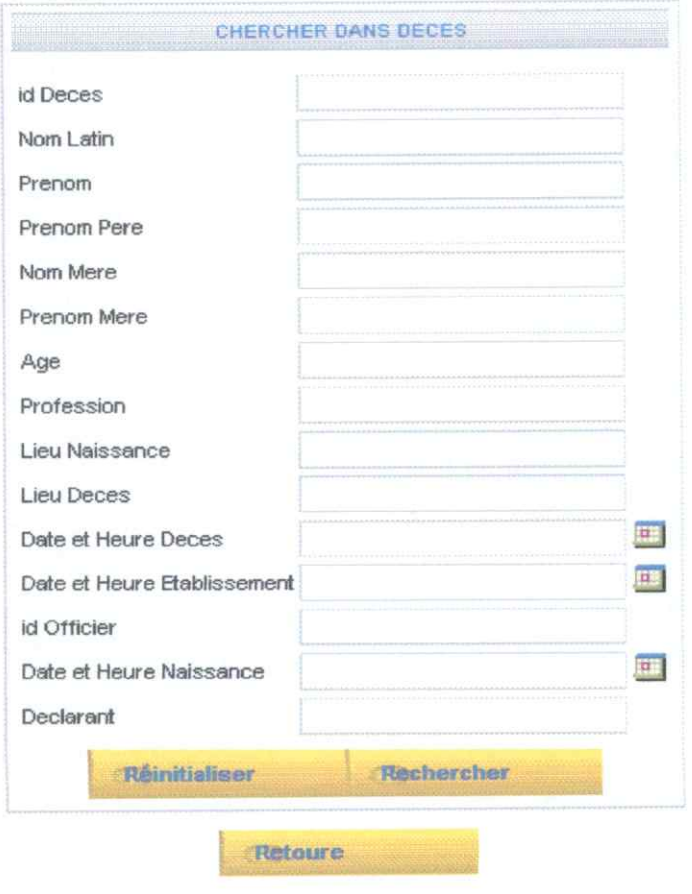


Figure V. 25 : Recherche Mise à Jour Des Actes De Décès

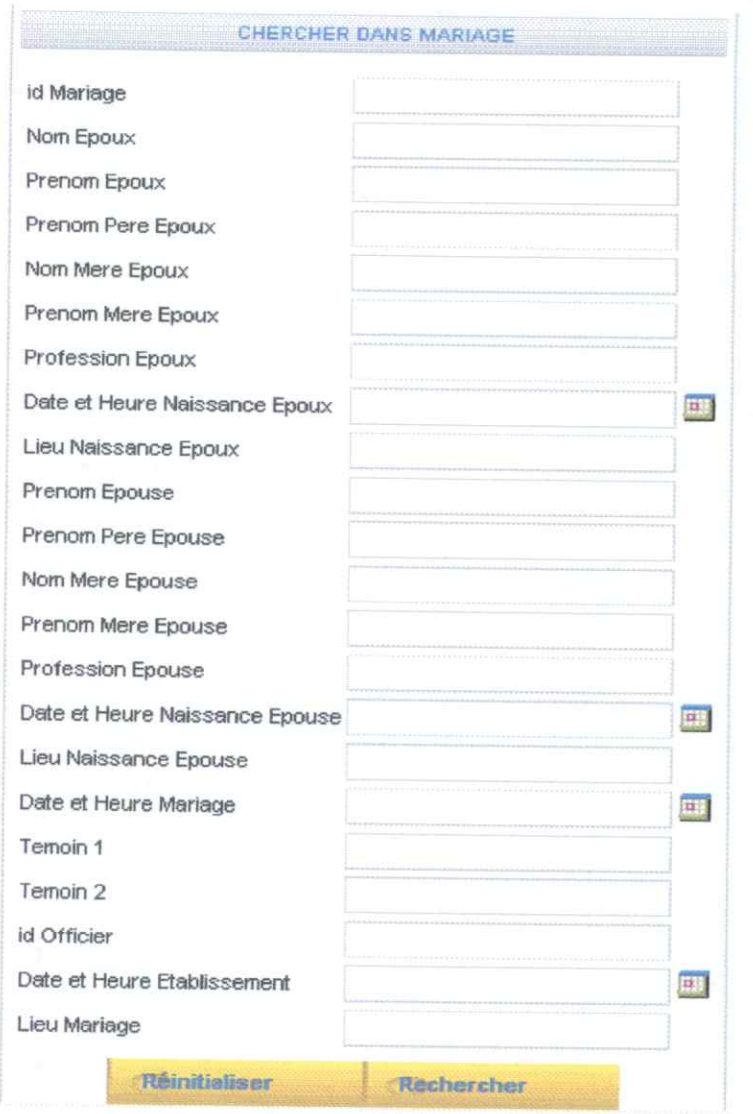


Figure V. 26 : Recherche Mise à Jour Actes De Mariage

Figure V. 27 : Recherche / Mise à Jour Actes Divorce

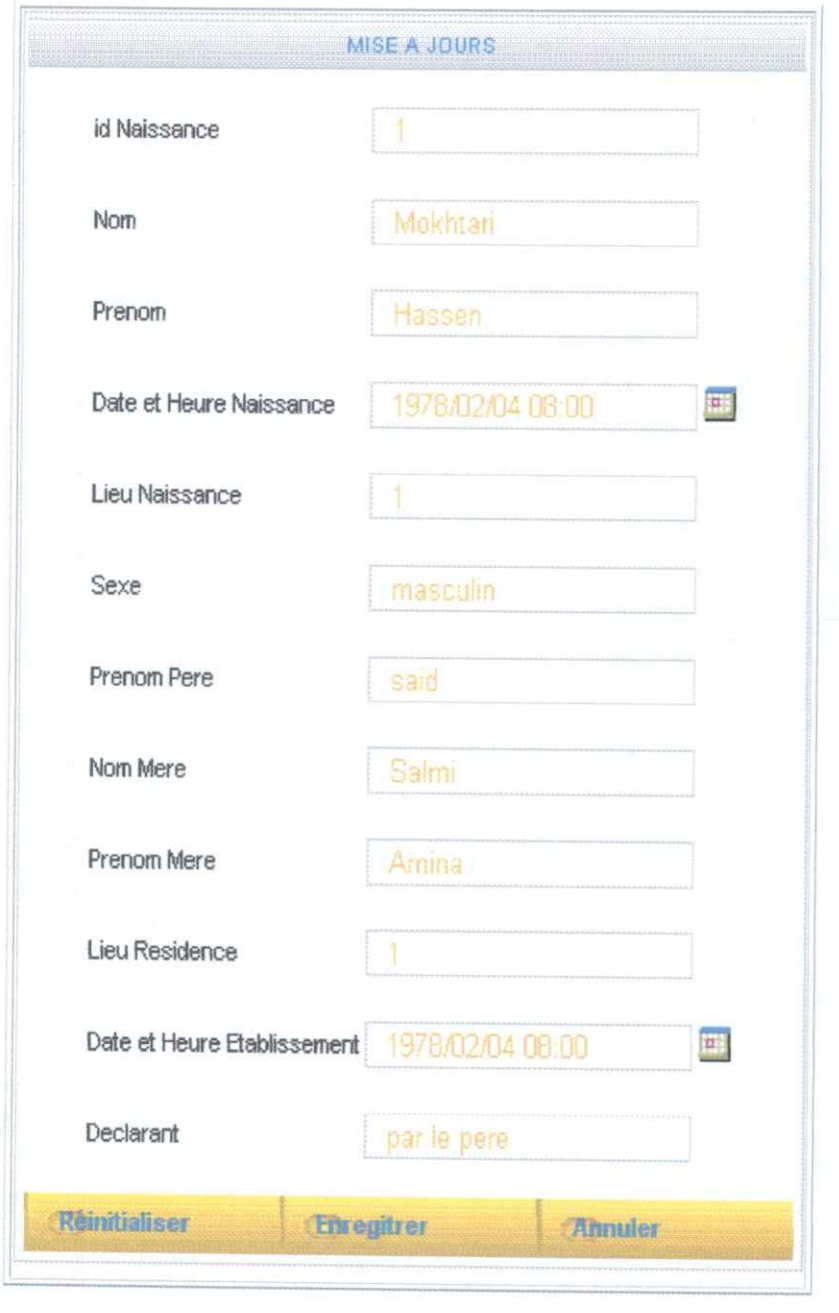


Figure V. 28 : Mise à Jour Actes Des Naissances

II.2.2.3. Citoyen:

Après avoir sélectionné « Ma Session » dans le menu général, chaque citoyen doit s'identifier lui aussi en tapant son login et son mot de passe **(Figure V.29) ;**

Une page alors apparaît montrant les différents documents qu'il peut commander par un simple choix **(Figure V.30) ;**

« Générer acte naissance », « générer acte mariage »ou « générer acte Divorce » ;

On pourra apercevoir dans la page de téléchargement du document deux liens : un pour le document sélectionné et l'autre pour la signature numérique correspondante à ce dernier

(Figure V.31) ; téléchargement des deux fichiers sous forme d'un dossier ZIP a partir d'un serveur FTP (FileZilla)qui est un serveur Open source se fait par un clique sur l'icône correspondante à chaque fichier .

Un exemple d'un document généré **(Figure V.32) ;**Et le téléchargement de la signature associé **(Figure V.33).**

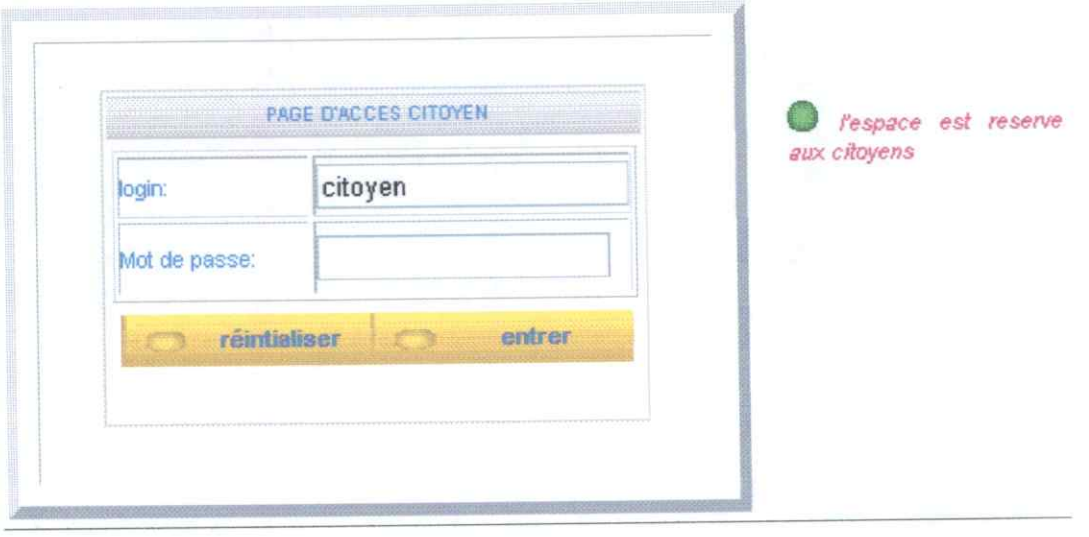


Figure V. 29 : Authentification Citoyen

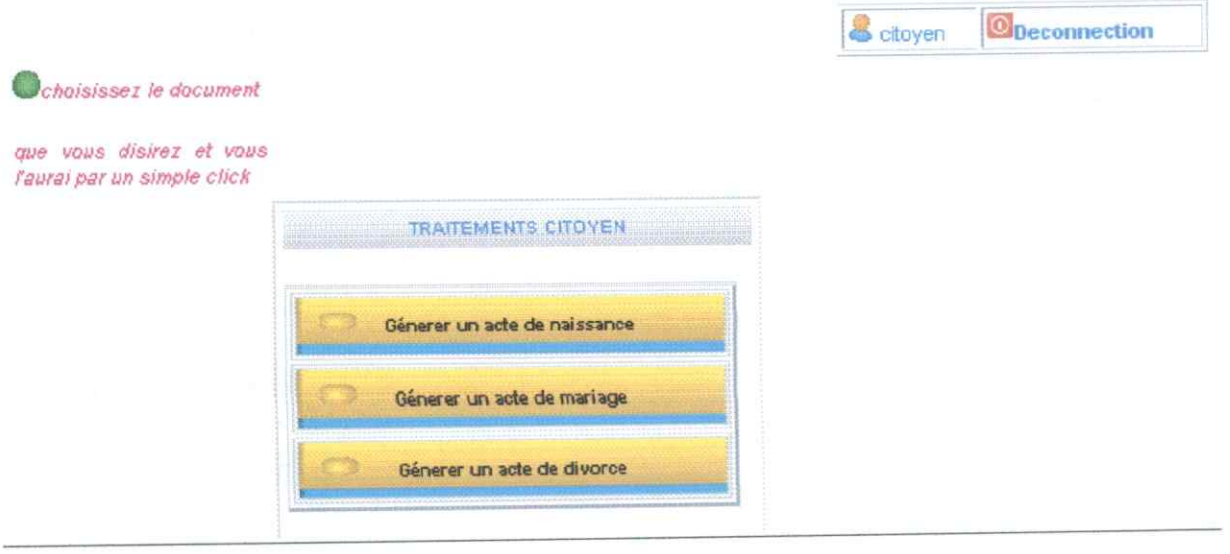


Figure V. 30 : Liste Des Fichiers à Extraire

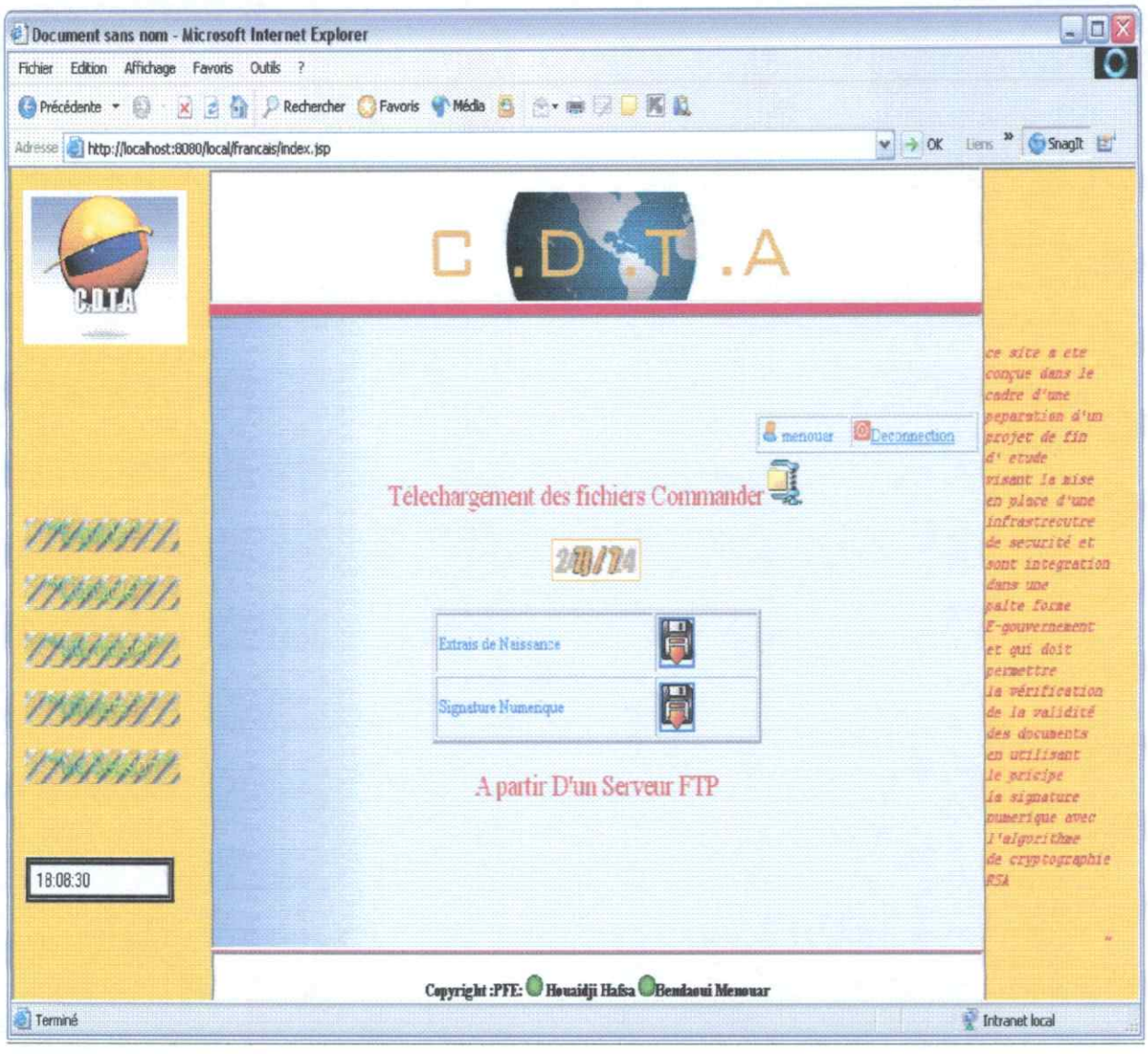


Figure V. 31 : Page de téléchargement des Fichiers

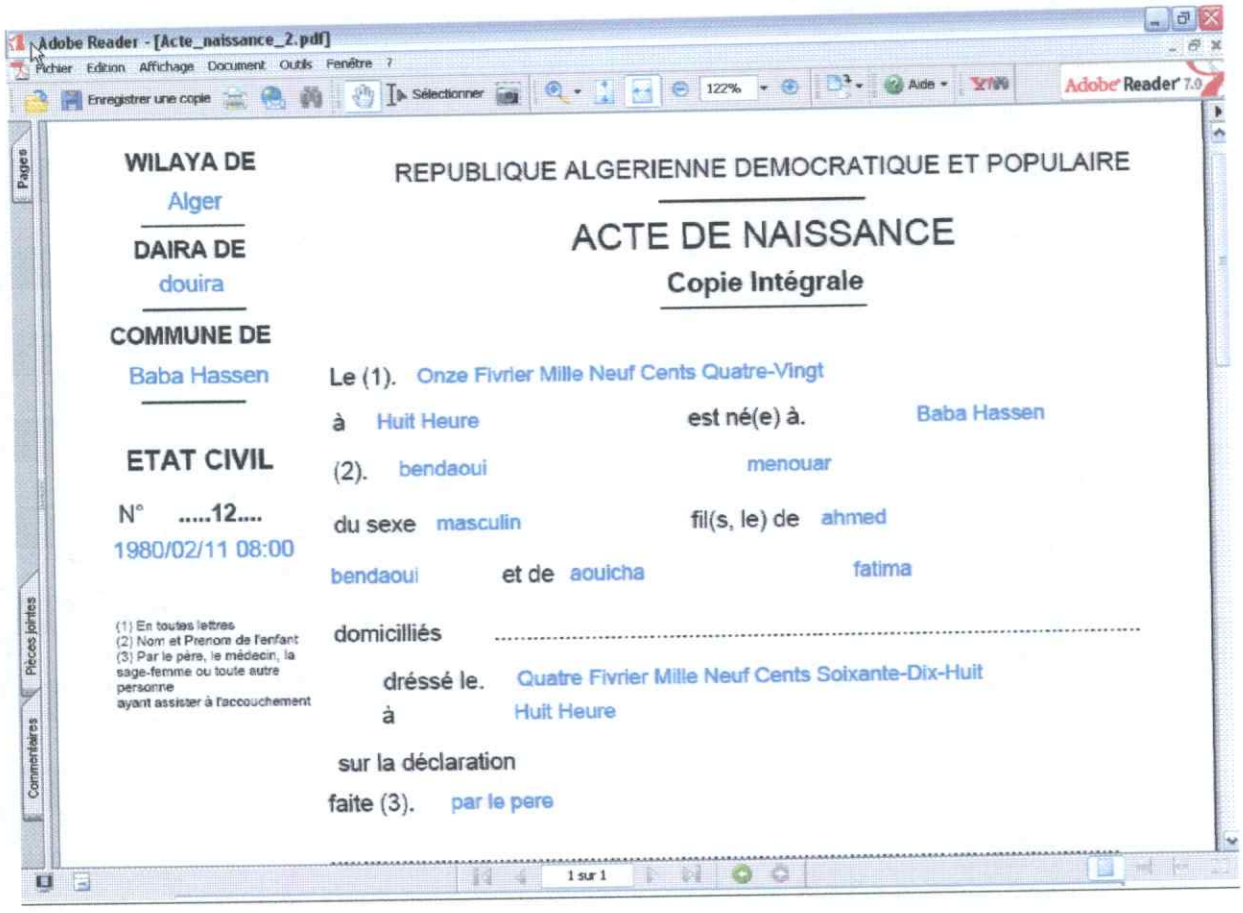


Figure V. 32 : Visualisation de document généré

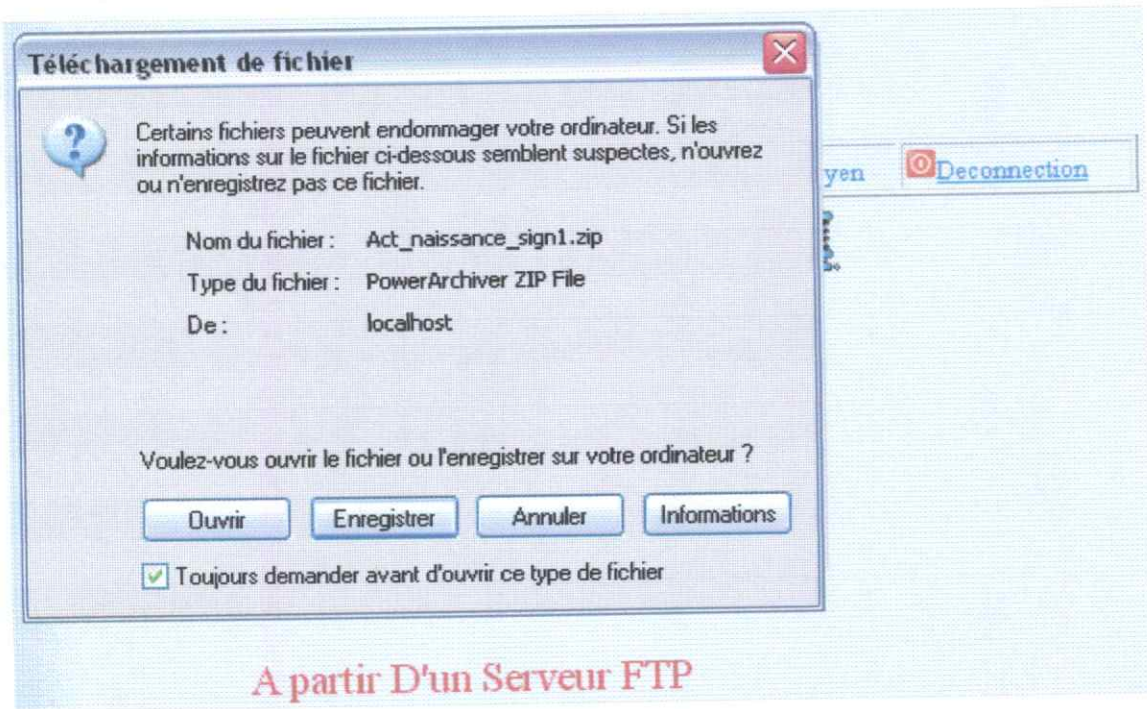


Figure V. 33 : Téléchargement de la signature numérique

III. Conclusion :

Nous avons déployé notre application de e-gouvernance pour l'échange de documents d'état civil. Nous avons pour cela crée deux système : central et local.
Chacun d'eux à nécessiter la création du dispositif de sécurité des documents électroniques.
Dans un premier temps, nous avons créé les diverses bases de données d'utilisateur. Puis avons procédé à la création de documents « fiches d'état civil ».
Puis enfin, avons commandé et échangé les différents documents en réseau.

Les technologies de l'information permettent de fournir des services de base aux populations, dans différents domaines, on peut citer parmi eux la télé éducation, le commerce électronique et la cybergouvernement **(e-service, e-gouvernement, e-commerce, e-vote etc..).** Ces services ont connu un expansion sans précédant et doivent être dotés d'un certain niveau de sécurité pour permettre la protection de la vie privé des individus et la protection des informations concernant les organisations et leurs états.

Dans ce travail nous avant d'abord présenté le domaine étudié « eGouvernement » et ses particularités.

Avec ces définitions et les défis de sécurité qui suivent son évolution nous avons abordés les techniques de sécurité après nous avons choisi suite à cette étude une infrastructure assurant la sécurité du dispositif de e-gouvernance.

Parmi les solutions de sécurité existantes (signature numérique, cryptage, certificat numérique, protocole SSL), nous avons opté pour une solution qui répond aux exigences fixées par la e-gouvernance. Ceci nous a amené à appliquer une stratégie de sécurité en utilisant le principe de la signature numérique avec l'algorithme « RSA »

Ainsi quand un citoyen reçoit un document, on lui donne les moyens d'être sur de l'identité de l'expéditeur (c'est l'authentification). Par ailleurs, quand un citoyen reçoit un document on lui garantit que les données sont complets (c'est l'intégrité des données).

En fin, quand un citoyen reçoit un document d'une APC, il détient aussi la preuve que cette dernière, et elle seule, le lui a envoyé (c'est la non répudiation).

Pour installer une telle infrastructure, nous devons disposer de clé de cryptage, étant donné que notre pays ne possède pas une autorité de certification délivrant ces clés, nous avant mis en place un système centralisé qui gène les clés de cryptage.

Cependant, une gestion plus sécurisée des clés nécessiterait de créer une telle autorité de certification dans un avenir proche. Par ailleurs, encore dans la perspective d'améliorer la sécurité, il serait intéressant d'intégrer la signature numérique dans le document lui-même.

Annexe

- **Définition :**

Semence

Une semence est une valeur utilisée pour initialiser un algorithme de génération de nombres pseudo aléatoires.

Codes d'authentification du message

Les signatures numériques calculées à partir d'une cryptographie à clé secrète sont appelées des codes d'authentification du message.

- **Exemple de chiffrement RSA**

Pour avoir une idée du fonctionnement de RSA, nous utiliserons les nombres premiers 11 et 13 pour générer n=143 (dans une véritable application de RSA, il faudrait choisir des nombres de plusieurs centaines de chiffres). Il faut sélectionner e,

Qui doit être premier avec (p-1) (q-1) = 120. Comme les seuls facteurs premiers de 120 sont 2, 2 et 5, on peut définir e=7 Ensuite, il faut calculer d pour que 7d mod 120 égale 1. Pour cela, on peut multiplier par 7 les nombres 2, 3, 4...jusqu'à trouver une valeur pour laquelle 7x mod 120 égale 1. Dans notre cas, d vaut 103.

Il faudrait crypter que des nombres inférieurs à n. Nous nous limiterons au cryptage d'une valeur ASCII 7 bits à la fois. Dans une application réelle, il faudrait crypter au moins des valeurs de 64 bits.

Prenons le nombre 74, qui est le code ASCII de la lettre J. Pour crypter cette valeur, il faut calculer 74^7 mod 143, qui donne 35 (une calculette scientifique permet d'effectuer ce calcul).

Le texte crypter de la lettre J est le nombre 35. Décryptons 35 en calculant 35^{103} mod 143. Le résultat de l'opération 35^{103} mod 143 donne 74, qui est bien la valeur d'origine.

- **Terminologie**

Cryptologie La science des documents secrets. Elle recouvre tous les aspects Scientifiques, et plus particulièrement mathématiques, relatifs à la Cryptologie et à la cryptanalyse.

Cryptanalyse L'art et la science de décrypter les messages secrets.

Chiffrement Le processus de transformation d'un message de telle manière à le Rendre incompréhensible pour toute personne non autorisée

Cryptogramme Le résultat du processus de chiffrement.

Cryptologue Celui qui écrit un cryptogramme.

Déchiffrement Le processus de reconstruction du texte clair à partir du texte Chiffré, par des personnes autorisées.

Décryptement Le processus de reconstruction du texte clair à partir du texte Chiffré, par des personnes non autorisées (sans connaître la clef).

Clef de chiffrement Un mot, un nombre ou une phrase qui est utilisé par un algorithme De cryptologie pour chiffrer ou déchiffrer un message.

Stéganographie La dissimulation du message dans un ensemble de données d'apparence anodine.

- **JasperReport**

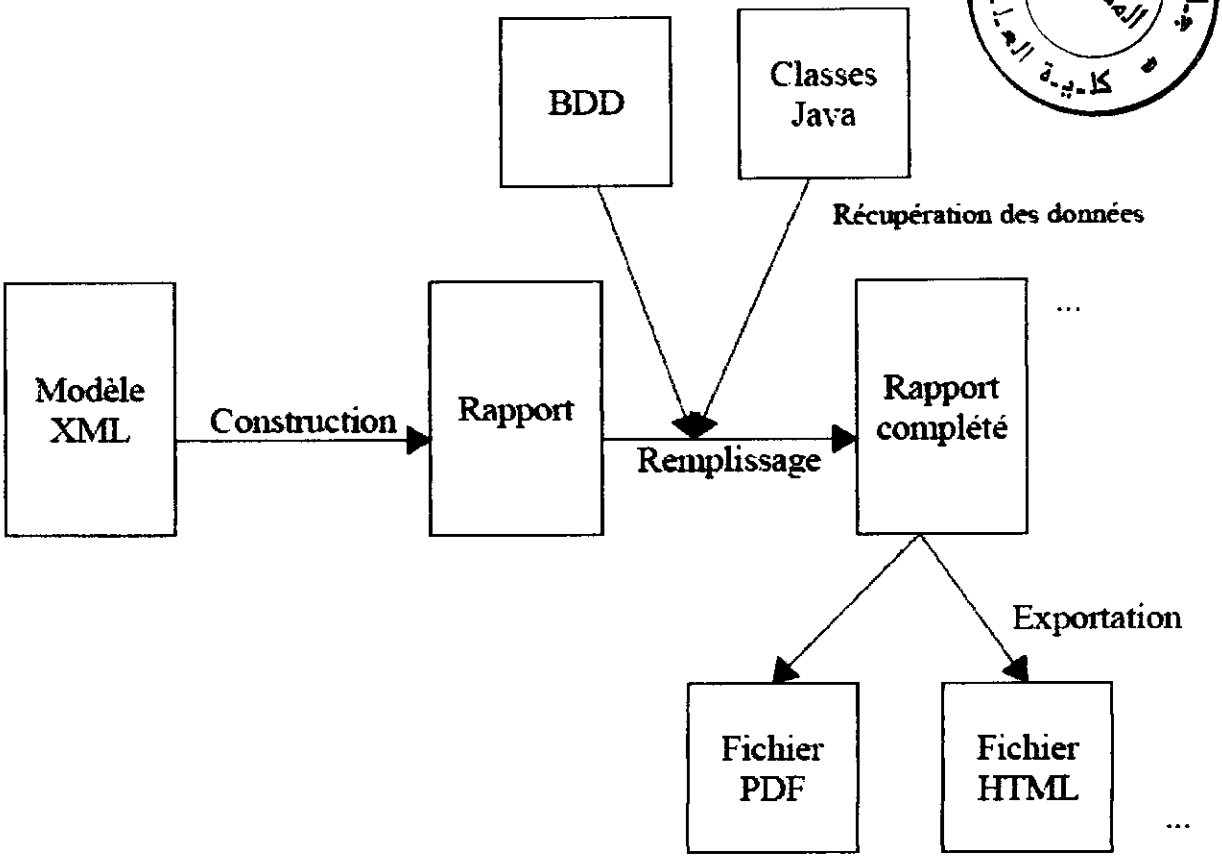
La création de rapports avec JasperReports se déroule généralement en 4 étapes :

• l'obtention d'un fichier modèle XML (à l'aide d'éditeurs graphiques comme iReport ou OpenReports Designer)

• la construction du rapport à partir du modèle

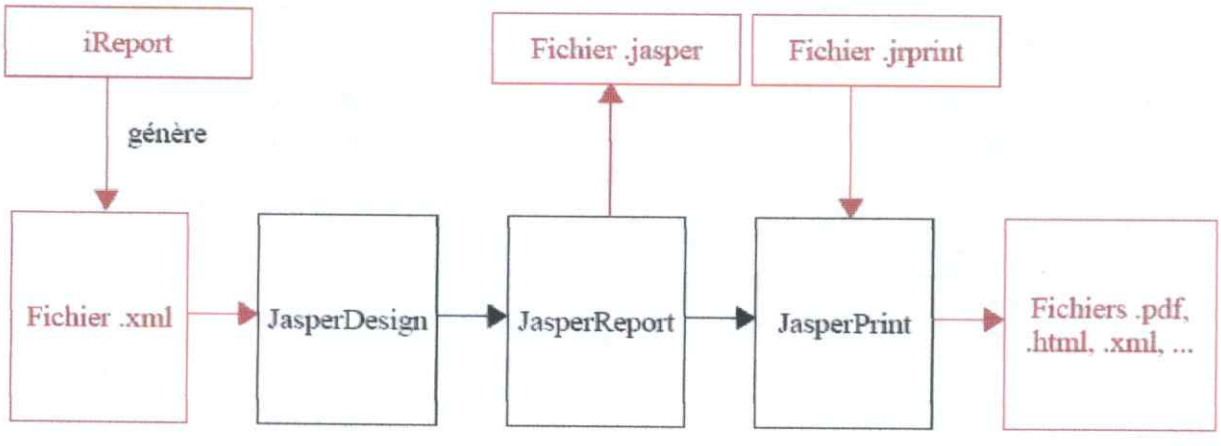
• le remplissage des différents champs du rapport avec les données en provenance de diverses sources (bases de données, classes Java, ...)

• l'exportation du résultat dans plusieurs formats possibles (PDF, HTML, ...)



Le fonctionnement de JasperReports est relativement simple. En effet, tous les concepts tournent autour du langage Java. Une fois le modèle XML (JasperDesign) compilé, il est chargé dans un objet Java (JasperReport) qui peut lui-même être sérialisé et stocké dans un fichier (avec l'extension .jasper). Cet objet sérialisé est alors utilisé lorsque l'application désire compléter le rapport avec des données. En fait, la définition du rapport nécessite la compilation de toutes les expressions Java déclarées dans le modèle XML. Le résultat obtenu après le processus de remplissage des champs est un nouvel objet Java (JasperPrint) qui représente le document final.

Celui-ci peut être stocké sur disque pour un usage ultérieur (sous forme sérialisée et avec l'extension .jrprint), directement imprimé ou encore transformé dans un format lisible (PDF,HTML, ...).



Bibliographie

- **[ghe 06]:** Solange ghernaouti Guide de la cybersécurité pour les pays en développement. 2006
- **[ghe 03]:** Solange ghernaouti L'annuaire suisse de politique de développement. 2003
- **[nto 05]:** A. Ntoko Mandate and activities in cybersécurity. 2005
- **[leg 07]:** Patrick Legand Aspects de la cryptologie au cours de l'Histoire. 2007
- **[jaw 01]:** Jamie Jaworski Java security. 2001
- **[Lss 02]:** auteur L'art du secret, dossier pour la science. Edition Française de scientifique Américain. Juliet/Oct . 2002
- **[Psc 04]:** Paul Dubois, Stefan Hinz, Carsten Pedersen MySQL - Guide officiel. 2004
- **[kof 05]:** Michael Kofler MySQL 5 : Guide de l'administrateur et du développeur. 2005
- **[chd 03]:** Christophe Cachat et David Carella PKI Open Source Déploiment et administration. 2003
- **[bru 04]:** Bruno Martin Codage, Cryptographie et applications. 2004
- **[dum 01]:** Duane K.Fields, Mark A.Kolb JSP Java Server Pages Developpement de site web dynamiques.

Webographie

- **[Ref00]:** http://ec.europa.eu/information_society/eeurope/2005/doc/all_about/egov_communication_fr.pdf.
- **[Ref01]:** www.internet-observatory.be/internet_observatory/pdf/advices/advice_fr_002.pdf
- **[Ref02]:** http://www.rsasecurity.com/
- **[Ref03]:** www.conventions.coe.int/Treaty/FR/Treaties/Html/185.htm
- **[Ref04]:** www.certification.tn
- **[Ref05]:** http://fr.wikipedia.org/wiki/javaserverPage
- **[Ref06]:** http://fr.wikipedia.org/wiki/MySQL
- **[Ref07]:** www.dgfktc.org
- **[Ref08]:** http://www.atolcd.com/jasperreport_web.pdf
- **[Ref09] :**http://www.journaldunet.com/encyclopedie/definition/972/34/20/tomcat.shtml
- **[Ref10]:** http://www.generation-nt.com/filezilla-server-0-9-21-telechargement-26050.html