

**REPUBLIQUE DU SENEGAL**  
**Un Peuple - Un But – Une Foi**



\*\*\*\*\*

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR, DE LA RECHERCHE ET DE L'INNOVATION



**U.A.H.B**  
**UNIVERSITE AMADOU HAMPATE BA DE DAKAR**

\*\*\*\*\*

Départements des Sciences et Technologies de l'Information  
et de la Communication

**MEMOIRE DE FIN DE CYCLE POUR L'OBTENTION  
DU DIPLOME DE LICENCE PROFESSIONNELLE**

**MISE EN PLACE D'UNE PLATEFORME DE GESTION  
DES FAITS DE L'ETAT CIVIL**

**Cas : COMMUNE DE DALIFORT-FOIRAIL**

**OPTION : INFORMATIQUE**

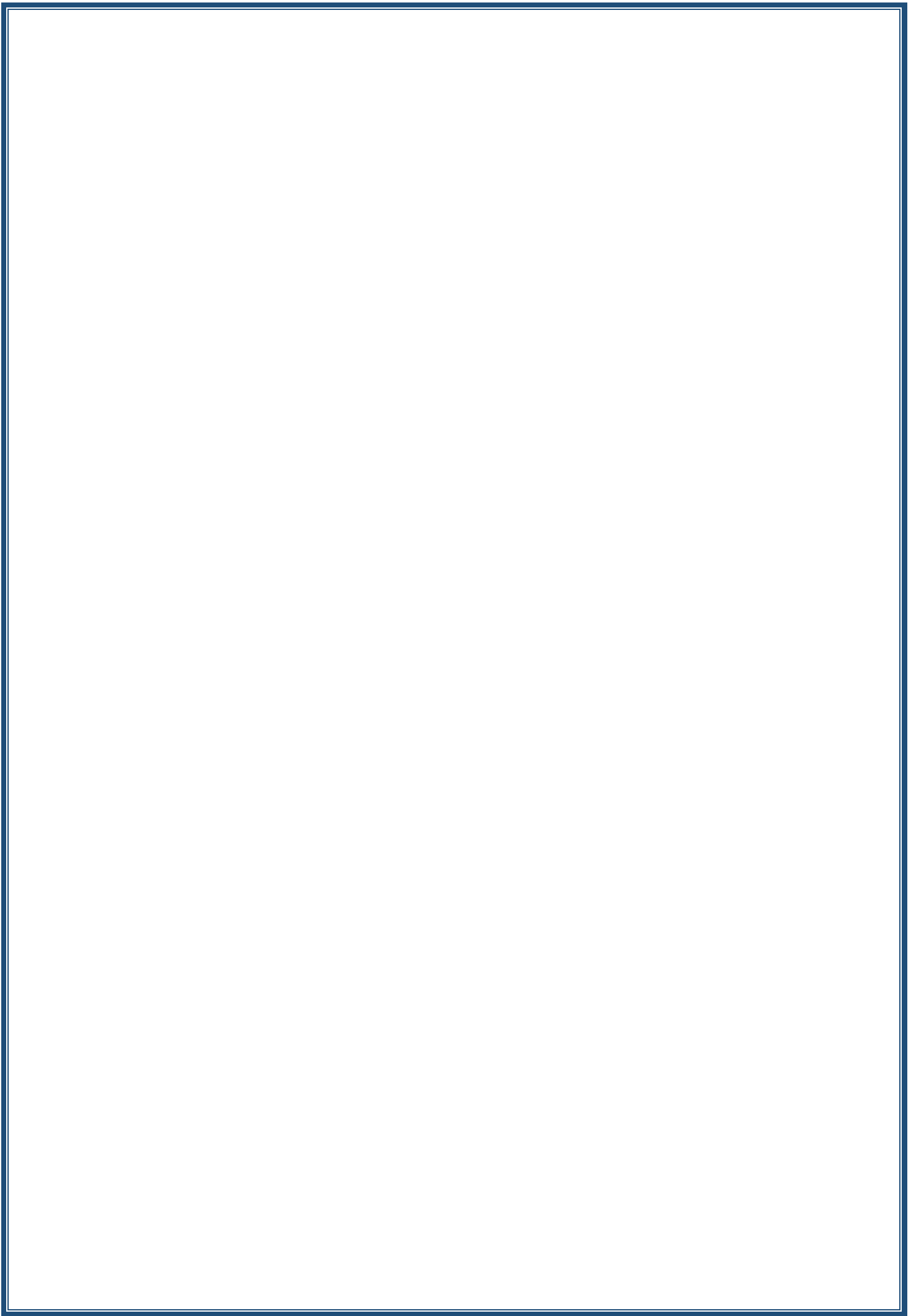
**Réalisée et Soutenue par :**

**Mlle GAYE Farmata**

**Encadreur :**

**M. Lamine DIOP**

***Année académique 2020-2021***



## **DEDICACES**

*Je dédie ce présent mémoire à :*

*Mes très chers parents, source de vie d'amour et d'affection*

*Mes chers frères et sœurs, source de joie et de bonheur*

*Toute ma famille, source d'espoir et de motivation*

*Et à tous mes amis*

## **REMERCIEMENTS**

*C'est avec un grand plaisir que nous réservons cette page en signe de notre sincère gratitude envers tous ceux qui nous ont participés, de près ou de loin au bon déroulement de ce travail.*

*Tout d'abord, ce travail ne serait pas aussi riche et n'aurait pas pu avoir le jour sans l'aide et l'encadrement de **Mr Lamine DIOP et Ibrahima LO** qui, grâce à leur simplicité, leur entière disponibilité et leurs conseils nous ont permis de parachever ce travail dans des conditions satisfaisantes ;*

***A tous nos enseignants** : Qui jugeront ce projet, où ils y trouveront l'expression de notre grande considération et notre plus profond respect.*

***A Mr ALOUIS DIATTA** chef de service de la mairie pour son accompagnement dans l'accomplissement de ce travail ;*

*Nous ne saurions oublier notre parrain **Pape Sidy CISSE, Ndiaya Samb NDAO** qui, depuis le début de notre formation nous guide vers le chemin de la réussite ;*

***A Mes amis** de la promotion STIC 2017/2020 particulièrement à **Maimouna Aidara Faye NDIAYE** ;*

*Nous réitérons nos remerciements à l'endroit de notre **FAMILLE STIC** ;*

*A tous ceux qui ont participé tant soit peu à la réussite du projet.*

**Merci !!!**

## **LISTE DES ABREVIATIONS**

**CSS** : Cascading Style Sheets (feuilles de style en cascade)

**CTI** : Centre Technique d'Informatique

**EDI** : Environnement de Développement intégré

**HTML**: Hypertext Markup Language

**HTTP**: Hypertext Transfer Protocol

**JS** : JavaScript

**MCC** : Modèle conceptuel de la communication

**MCD** : Modèle Conceptuel des données

**MCT** : Modèle conceptuel des traitements

**MERISE** : Méthode d'Etude de Réalisation Informatique par Sous-Ensemble

**MOT** : Modèle Organisationnel des traitements

**MVC** : Modèle-Vue-Contrôleur

**OMT** : Object Modeling Technique (technique de modélisation objet)

**OOSE**: Object Oriented Software Engineering

**PHP**: Hypertext Preprocessor

**SGBD** : Système de Gestion de Base de Données

**SI** : Système d'Information

**SQL** : Structure Query Langage (langage de requête structurée)

**SSI** : Sécurité des systèmes d'information

**UAHB** : Université Amadou Hampaté Ba

**UML** : Unified Modeling Language

**LISTE DES TABLEAUX**

<b>TABLEAU I : DESCRIPTION TEXTUELLE « AUTHENTIFICATION »</b> .....	39
<b>TABLEAU II : DESCRIPTION TEXTUELLE « ENREGISTRER NAISSANCE »</b> .....	41
<b>TABLEAU III : DESCRIPTION TEXTUELLE « ENREGISTRER MARIAGE »</b> .....	43
<b>TABLEAU IV : DESCRIPTION TEXTUELLE « ENREGISTRER DECES »</b> .....	45
<b>TABLEAU V : TABLEAU FINANCIER</b> .....	66

**LISTE DES FIGURES**

<b>FIGURE 1 : SITUATION GEOGRAPHIQUE DE LA MAIRIE</b> .....	10
<b>FIGURE 2 : ORGANIGRAMME DE LA MAIRIE</b> .....	11
<b>FIGURE 3 : REPRESENTATION DES DIFFERENTS DIAGRAMMES D'UML</b> .....	23
<b>FIGURE 4 : FORMALISME D'UN DIAGRAMME DE CAS D'UTILISATION</b> .....	24
<b>FIGURE 5 : FORMALISME D'UN DIAGRAMME DE CLASSE</b> .....	25
<b>FIGURE 6 : EXEMPLE DE DIAGRAMME DE DEPLOIEMENT</b> .....	26
<b>FIGURE 7 : FORMALISME D'UN DIAGRAMME DE SEQUENCE</b> .....	27
<b>FIGURE 8 : EXEMPLE DE DIAGRAMME D'ACTIVITE</b> .....	27
<b>FIGURE 09 : DIAGRAMME DE CAS D'UTILISATION « S'AUTHTENTIFIER »</b> .....	33
<b>FIGURE 10 : DIAGRAMME DE CAS D'UTILISATION « GESTION DES NAISSANCES »</b> .....	34
<b>FIGURE 11 : DIAGRAMME DE CAS D'UTILISATION « GESTION DES MARIAGES »</b> .....	35
<b>FIGURE 12 : DIAGRAMME DE CAS D'UTILISATION &lt;&lt;GESTION DES DECES&gt;&gt;</b> .....	36
<b>FIGURE 13 : DIAGRAMME DE CAS D'UTILISATION &lt;&lt; DEMANDE PIECES &gt;&gt;</b> .....	37
<b>FIGURE 14 : DIAGRAMME DE SEQUENCE &lt;&lt;AUTHENTIFICATION&gt;&gt;</b> .....	38
<b>FIGURE 15 : DIAGRAMME DE SEQUENCE &lt;&lt;ENREGISTRER-NAISSANCE&gt;&gt;</b> .....	40
<b>FIGURE 16 : DIAGRAMME DE SEQUENCE &lt;&lt;ENREGISTRER-MARIAGE&gt;&gt;</b> .....	42
<b>FIGURE 17 : DIAGRAMME DE SEQUENCE « ENREGISTRER-DECES »</b> .....	44
<b>FIGURE 18 : DIAGRAMME DE CLASSE</b> .....	46
<b>FIGURE 19 : PRESENTATION DE L'ARCHITECTURE 1-TIER</b> .....	57
<b>FIGURE 20: PRESENTATION DE L'ARCHITECTURE</b> .....	59
<b>FIGURE 21 : PRESENTATION DE L'ARCHITECTURE 3-TIER</b> .....	60
<b>FIGURE 22 : PRESENTATION DE L'ARCHITECTURE N-TIERS</b> .....	57
<b>FIGURE 23 : DIAGRAMME DE COMMUNICATION DU PATRON MVC</b> .....	59
<b>FIGURE 24 : AUTHENTIFICATION</b> .....	60
<b>FIGURE 25 : AJOUT UTILISATEUR</b> .....	60
<b>FIGURE 26 : CHANGER MOT DE PASSE</b> .....	61
<b>FIGURE 27 : ACCUEIL OFFICIER</b> .....	61
<b>FIGURE 28 : ACCUEIL AGENT</b> .....	62
<b>FIGURE 29 : ENREGISTRER NAISSANCE</b> .....	62
<b>FIGURE 30 : DIAGRAMME DE DEPLOIEMENT</b> .....	67

## Table des matières

<b>INTRODUCTION .....</b>	<b>7</b>
<b>PREMIERE PARTIE :CADRE THEORIQUE ET METHODOLOGIQUE .....</b>	<b>9</b>
<b>CHAPITRE I : LE CADRE THEORIQUE.....</b>	<b>10</b>
<b>SECTION 1 : PRESENTATION DE LA MAIRIE .....</b>	<b>10</b>
I.    PRESENTATION .....	10
II.   SITUATION GEOGRAPHIQUE.....	10
III.  ORGANISATION .....	11
<b>SECTION 2 : PRESENTATION DU SUJET .....</b>	<b>12</b>
I.    CONTEXTE .....	12
II.   LA PROBLEMATIQUE.....	12
III.  OBJECTIS DE LA RECHERCHE.....	13
IV.  DELIMITATION DU CHAMP DE L'ETUDE .....	13
<b>CHAPITRE II : LE CADRE METHODOLOGIQUE .....</b>	<b>14</b>
<b>SECTION 1 : METHODE D'INVESTIGATION.....</b>	<b>14</b>
I.    METHODES .....	14
II.   TECHNIQUES .....	14
<b>SECTION 2 : ETUDE COMPARATIVE DES METHODES D'ANALYSE ET DE CONCEPTION .....</b>	<b>15</b>
I.    PRESENTATION .....	15
II.   COMPARAISON .....	17
III.  CHOIX DE LA METHODE .....	17
<b>DEUXIEME PARTIE :ANALYSE ET CONCEPTION .....</b>	<b>19</b>
<b>CHAPITRE I : ANALYSE.....</b>	<b>20</b>
<b>SECTION 1 : PREREQUIS THEORIQUES .....</b>	<b>20</b>
I.    APPROCHE FONCTIONNELLE .....	20
II.   APPROCHE ORIENTEE OBJET .....	20
III.  FORMALISMES DES DIFFERENTS DIAGRAMMES D'UML .....	22
<b>SECTION 2 : ANALYSE DU SYSTEME.....</b>	<b>28</b>
I.    RECENCEMENT ET ANALYSE DES BESOINS .....	28
II.   DESCRIPTIONS DES PROCESSUS .....	28
III.  DIAGRAMME USE CASE.....	33
<b>CHAPITRE II : CONCEPTION .....</b>	<b>38</b>
<b>SECTION 1 : DIAGRAMMES DE SEQUENCE.....</b>	<b>38</b>

SECTION 3 : DIAGRAMMES DE CLASSE .....	46
<b>TROISIEME PARTIE : IMPLEMENTATION ET DEPLOIEMENT .....</b>	<b>47</b>
<b>CHAPITRE I : IMPLEMENTATION .....</b>	<b>48</b>
SECTION 1 : ENVIRONNEMENT .....	48
SECTION 2 : ETUDE DE QUELQUES LANGAGES DE PROGRAMMATION .....	48
I.    DEFINITION .....	48
II.   EXEMPLE DE QUELQUES LANGAGES DE PROGRAMMATION .....	49
III.  ETUDE DE FRAMEWORK .....	53
SECTION 3 : ETUDE COMPARATIVE DE QUELQUES SGBD .....	54
I.    PRESENTATION .....	54
II.   EXEMPLES DE QUELQUES SGBD .....	54
III.  CHOIX ET JUSTIFICATION .....	56
SECTION 3 : ARCHITECTURE APPLICATIVE .....	56
I.    DEFINITION .....	56
II.   EXEMPLE D'ARCHITECTURE .....	56
SECTION 4 : MVC(MODELE-VUE-CONTROLEUR) .....	58
<b>CHAPITRE II : DEPLOIEMENT .....</b>	<b>60</b>
SECTION 1 : PRESENTATION DE LA PLATEFORME .....	60
SECTION 2 : GESTION DE LA SECURITE .....	63
I.    DEFINITION ET ROLE .....	63
SECTION 3 : ESTIMATION COUT .....	66
SECTION 3 : DIAGRAMME DE DEPLOIEMENT .....	67
<b>CONCLUSION .....</b>	<b>68</b>

# INTRODUCTION

Il est vrai que le monde connaît une avancée technologique dans tous les secteurs grâce à l'ordinateur qui est défini comme l'équipement informatique de traitement automatique des données comprenant les organes nécessaires à son fonctionnement autonome. Elle joue un rôle important dans le développement du travail. Jusqu'à présent, l'ordinateur reste le moyen le plus sûr et facile pour le traitement et la sauvegarde des informations. De ce fait, l'invention de l'ordinateur a permis d'informatiser les systèmes des données dans différentes entreprises et services de l'État.

L'état-civil constitue à la fois un élément fondamental dans la vie des citoyens et une source de données démographiques essentielle. En conséquence, la modernisation, l'informatisation des centres et la sécurisation des données et faits d'état-civil doivent occuper une place de premier plan dans les politiques publiques et les stratégies de développement économique et social du pays. Au Sénégal, cette problématique est d'autant plus importante que des exemples de dégradation volontaire (actes de vandalisme) ou non (mauvaises conditions d'archivage combinées à l'usure du temps) des documents d'état-civil existent avec toutes les conséquences néfastes pour les citoyens qui en sont victimes et qui se retrouvent ainsi dans l'impossibilité de se faire établir par exemple un extrait de naissance. Cette situation aurait pu être évitée si l'état-civil était modernisé, informatisé et d'avantage sécurisé au Sénégal.

L'objectif de mon projet de fin de cycle présenté dans ce rapport est la conception et réalisation d'une plateforme pour informatiser la gestion des faits de l'état civil.

Ainsi cette étude se fera en trois (3) parties :

- ❖ Nous verrons dans la première partie « **Cadre Théorique et Méthodologique** »
- ❖ Dans la deuxième partie « **Analyse et Conception** »
- ❖ Dans la troisième partie « **Implémentation et Déploiement** »

# **PREMIERE PARTIE :**

## **Cadre Théorique**

### **Et**

## **Méthodologique**

## CHAPITRE I : LE CADRE THEORIQUE

### SECTION 1 : PRESENTATION DE LA MAIRIE

#### I. PRESENTATION

L'idée de la création de cette localité est venue d'un ressortissant malien dénommé Demba Diop. Ce dernier arriva au Sénégal en 1927, il fit connaissance avec un italien dénommé Ernest Christian, un fleuriste et exportateur de matériaux ferreux en Europe en 1937. C'est par l'intermédiaire de cet italien qu'il aménagea un jardin horticole en 1941 à Dalifort et s'y fixa de manière pérenne en 1943.

En 1957, Dalifort connut sa vraie occupation avec l'arrivée de ses premières communautés. Cette expansion spontanée et progressive fut possible grâce à l'action combinée de la migration rurale et urbaine. Le cadre institutionnel doit son existence à l'approfondissement d'un long processus de décentralisation initié par le gouvernement sénégalais depuis 1966 pour assurer une gestion de proximité et une responsabilisation accrue des populations à la base. C'est dans cette optique que la ville de Pikine fut divisée en 16 communes d'arrondissement par décret n° 96-795 du 30 Aout 1996.

De nos jours, Dalifort est l'une de ces communes. A l'instar des autres, elle est dotée de la personnalité morale et financière. La Commune d'Arrondissement de Dalifort-Foirail est créée le 30 Avril 1996 par décret n°96745.

#### II. SITUATION GEOGRAPHIQUE

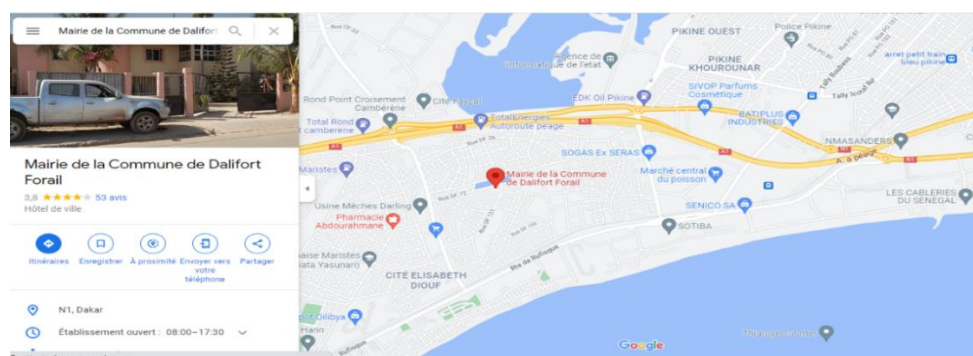


Figure 1 : Situation géographique de la mairie

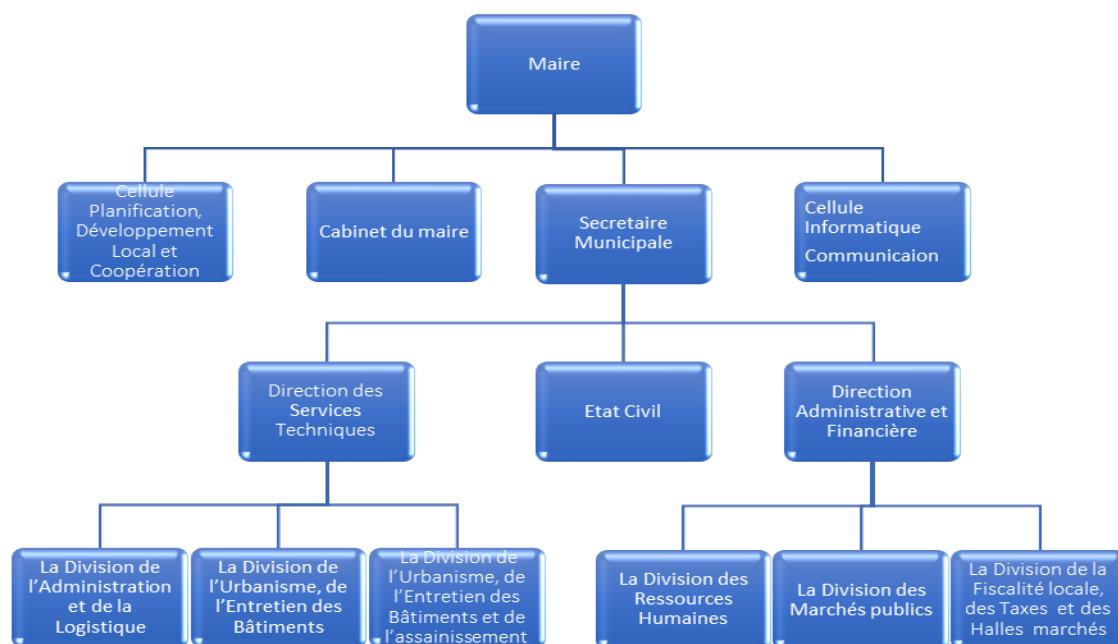
### III. ORGANISATION

Pour assurer la gestion quotidienne et la réalisation de ses actions publiques locales, la commune est dotée d'une organisation composée de services administratifs et techniques. Elle peut décider de gérer certaines activités en créant les services municipaux nécessaires,

La commune peut aussi déléguer la gestion d'une activité à un établissement public, une société privée ou une association.

Pour mettre en commun certaines actions et réduire leurs coûts, la ville peut décider de s'associer avec d'autres communes qui mettent alors en commun leurs moyens.

#### ORGANIGRAMME DE LA MAIRIE



**Figure 2 : Organigramme de la mairie**

## **SECTION 2 : PRESENTATION DU SUJET**

### **I. CONTEXTE**

L'état civil est au sens large, l'ensemble des qualités qui assignent à une personne sa place dans la société et la différencie des autres au point de l'exercice des droits civils. Autrement dit, c'est l'ensemble des éléments relatifs à une personne permettant d'identifier un individu. Par extension, c'est l'appellation donnée aux services administratifs d'une commune qui reçoivent les déclarations et qui conservent les registres concernant les naissances, les reconnaissances d'enfants naturels, les mariages et les décès.

Malgré son importance, de nombreux dysfonctionnements ont été constatés dans la gestion des faits d'état civil du Sénégal principalement sur les modalités de délivrance des actes et la conservation des registres, mais aussi des fraudes sur les documents d'état civil qui faussent les statistiques.

Pour mieux gérer ces problèmes, la gestion de l'état civil nécessite un fonctionnement plus fiable qui repose sur une gestion numérique. Ainsi la mise en place d'une plateforme est donc nécessaire pour faciliter le travail. Pour ce faire, nous nous sommes assignés comme mission de mettre en place une application pour la gestion des faits d'état civil.

Pour mener à bien notre projet, nous allons utiliser un ensemble de notions relatives aux méthodes de conception des systèmes d'informations et aux méthodes de conduite de projet.

### **II. LA PROBLEMATIQUE**

Les centres d'état civil du Sénégal offrent de nombreux services à la population sénégalaise. Cependant, malgré le travail en chaîne utilisé dans la quasi-totalité des processus, on y constate de nombreux dysfonctionnements notamment :

- ✦ **Les doublons** : le numéro de déclaration est unique, il arrive des fois que deux personnes aient le même numéro de déclaration suite à une erreur lors de la déclaration
- ✦ **La perte des données** : il est à noter que les centres sont confrontés à un problème d'espace pour bien gérer l'archivage. Les registres de l'état civil sont stockés dans des

armoires, ce qui conduit généralement à des pertes de registres causées soit par les inondations, soit par un incendie et les plus vieux sont illisibles ou déchires.

- ✦ **La délivrance** : on note également une certaine lenteur lors de la délivrance des actes par exemples, les demandes se font le matin et les retraits le soir et parfois même elle dure plus longtemps que prévus. Ceci s'explique par le fait qu'actuellement les recherches se font manuellement et la conservation des registres laisse à désirer d'où la lenteur dans les processus de recherche.

### **III. OBJECTIS DE LA RECHERCHE**

#### **A. OBJECTIFS PRINCIPALE**

Pour qu'un objectif soit pertinent et clair, il doit comporter la fonction souhaitée et le secteur d'activité visé. L'objectif principal pour ce mémoire est d'offrir à l'état civil un moyen de pouvoir numériser ses données.

#### **B. OBJECTIFS SPECIFIQUES**

Pour atteindre l'objectif principal, nous nous sommes assignées les objectifs spécifiques suivants :

- Permettre l'unicité des numéros de déclarations
- Permettre une déclaration rapide et fiable
- Permettre une meilleure conservation des actes
- Réduire le temps de recherche lors d'une demande
- Réduire le temps d'attente d'une demande d'acte d'état civil
- Evitez les fraudes.

### **IV. DELIMITATION DU CHAMP DE L'ETUDE**

La commune de dalifort-forail est composée de 4 services tous reliés au cabinet du maire. Notre dynamique du travail nous obligeant à nous rabattre sur un aspect limite du cadre de l'étude, nous nous sommes intéressées au service de l'état civil plus précisément les déclarations de naissance.

## CHAPITRE II : LE CADRE METHODOLOGIQUE

### SECTION 1 : METHODE D'INVESTIGATION

#### I. METHODES

La méthode, comme un ensemble d'opérations intellectuelles par laquelle une discipline scientifique cherche à atteindre un ou plusieurs objectifs poursuivis, les démontrer et les vérifier.

Pour y arriver nous avons utilisé les méthodes ci-après :

- **La méthode MERISE** : c'est une méthode informatique utilisée pour la conception du système d'information informatisé. C'est aussi une méthode de conception des projets informatique. Elle nous a permis d'analyser le système de gestion des enfants de l'Etat Civil en séparant les données de traitement.
- **La méthode structuro-fonctionnelle** : c'est une méthode qui permet d'étudier la structure et le fonctionnement d'une entreprise. Elle nous a permis de faire connaissance de la structure organisationnelle de l'Etat Civil, le fonctionnement des différents services.
- **Méthode comparative** : nous a permis d'établir une comparaison entre l'analyse de gestion de l'existant qui était manuelle et celle informatisée en vue de prendre une décision selon les avantages et les désavantages de chacune.

#### II. TECHNIQUES

##### A. OBSERVATIONS DIRECT

Partant de cette technique, nous avons récolté quelques informations en étant sur le terrain.

Cette étude nous a permis d'obtenir de nouvelles informations que nous avons utilisées dans notre travail. Les objectifs étaient de :

- capter des comportements au moment où ils se produisent
- recueillir de données concrètes et parfois quantifiables
- avoir une certaine authenticité des comportements, comparée aux paroles et aux écrits.

## **B. TECHNIQUE INTERVIEW**

Nous avons eu des interviews avec les acteurs concernés comme l'officier de la mairie, son adjoint et quelques agents.

## **C. TECHNIQUE DES QUESTIONNAIRES**

Ces personnes ont pu répondre au questionnaire élaboré et leurs réponses nous ont permis de recueillir des informations riches et spontanées, plus brutes et plus fraîches, nécessaires à notre travail. Avec cette technique, nous avons utilisé les questionnaires ouverts et les questionnaires fermes.

# **SECTION 2 : ETUDE COMPARATIVE DES METHODES D'ANALYSE ET DE CONCEPTION**

## **I. PRESENTATION**

Le concepteur du projet, à l'aide des méthodes de développement, traduit le besoin des utilisateurs en des symboles appartenant au jardin informatique. Plusieurs méthodes permettent de faire ces représentations mais les plus connus sont **MERISE ET UML**.

### **A. MERISE**

Issue de l'analyse systémique, la méthode MERISE (Méthode d'Etude de Réalisation Informatique par Sous-Ensemble) date de 1978 – 1979, à la demande du ministère de l'industrie, et a surtout été utilisée en France, par le SSI de ses membres fondateurs (Sema Métra, ainsi que par la CGI Informatique) et principalement pour les envergures, notamment des grandes administrations publiques ou privées.

La méthode Merise présente un ensemble de formalisme et de règles destinées à modéliser de manière indépendante les données et les traitements de systèmes d'information. Ces modèles ne sont qu'une base de réflexion pour le concepteur et un moyen de communication entre les différents acteurs du système d'information.

Merise est une méthode de conception, de développement et de réalisation d'application informatique. Son but est d'arriver à concevoir un système d'information. La méthode MERISE est basée sur la séparation des données et des traitements à effectuer en plusieurs modèles conceptuels, organisationnels et physiques.

La méthode Merise comprend trois (3) niveaux :

- Le niveau conceptuel
- Le niveau organisationnel
- Le niveau opérationnel

Mais, malgré ces avantages, on reproche à MERISE d'utiliser un formalisme jugé complexe. En plus de cela, cette méthode est essentiellement franco-française et n'a pas eu beaucoup de succès à l'étranger où les méthodes anglo-saxonnes sont plus présentes.

## **B. UML**

UML (Unified Modeling Language) est un langage de modélisation graphique à base de pictogrammes. Il est utilisé pour le développement logiciel en orientée objet. Il fournit les fondements pour spécifier, construire, visualiser et décrire les artifices d'un système logiciel. Pour cela, UML se base sur une sémantique précise et sur une notation graphique expressive. Il définit des concepts de base et offre également des mécanismes d'extension de ces concepts. UML est un langage de modélisation objet. En tant que tel, il facilite l'expression et la communication de modèles en fournissant un ensemble de symboles et de règles qui régissent l'assemblage de ces symboles.

UML permet de modéliser de manière claire et précise la structure et le comportement d'un système indépendamment de toute méthode ou de tout langage de programmation. Les créateurs d'UML insistent tout particulièrement sur le fait qu'UML est un langage de modélisation et non Une méthode. UML est la fusion des langages Booch, OMT et OOSE il s'est rapidement imposé à la fois auprès des utilisateurs et sur le terrain de la normalisation.

UML n'est pas un langage propriétaire et est accessible à tous et les fabricants d'outils ainsi que les entreprises de formation peuvent librement en faire usage.

UML est donc un langage visuel qui permet de modéliser et de communiquer à propos de systèmes, par l'intermédiaire de diagrammes et de textes.

## **II. COMPARAISON**

Le choix de l'un ou de l'autre se fait selon trois (3) axes à savoir l'accessibilité, la précision et l'exploitabilité mieux adaptée pour son cas.

Pour le premier axe (**accessibilité**) MERISE présente l'intérêt d'avoir des modèles logiques moins détaillés facilement compréhensibles. Tandis qu'UML conçu pour s'adapter à n'importe quel langage de Programmation Orientée Objet (POO), présente plusieurs modèles (diagrammes) dont leurs compréhensions nécessitent une grande attention.

En ce qui concerne le deuxième critère (**précision**), MERISE est moins préférable. Malgré sa clarté, il manque une précision du fait qu'elle est éloignée du langage donc difficile à implémenter alors qu'UML intègre les éléments communs des différents langages, sa volonté est d'être fidèle à la réalisation finale. Elle est beaucoup plus complète avec ses différents diagrammes.

Pour en finir avec l'**exploitabilité**, MERISE est une méthode plus généraliste. Elle donne une vue globale de la solution sans autant entrer dans les petits détails. Contrairement à UML qui est conçu pour l'implémentation objet avec ses différents détails et sa portabilité (s'adapte à n'importe quelle plateforme) elle est donc plus exploitable.

L'une ou l'autre présente des avantages et des inconvénients. Il est réservé au concepteur de choisir la méthode la mieux adaptée pour son cas. Si on cherche la précision et l'exploitabilité UML devance MERISE. Tandis que, si c'est la clarté et l'accessibilité qui sont en question MERISE est préférable.

## **III. CHOIX DE LA METHODE**

Pour mener à bien notre projet nous devons choisir une méthode qui nous permettra de modéliser notre système en la structurant en objets. Ainsi nous allons opter pour la modélisation **UML**. Comme nous l'avons dit UML est un langage de modélisation unifié qui est beaucoup plus orienté objet que les autres méthodes énoncées précédemment.

La maintenance est également beaucoup plus facile avec UML car ici lors de la modélisation on ne sépare pas les données des traitements. Avec la modélisation UML, au lieu d'avoir une structure de données manipulée par des fonctions, nous avons une entité autonome composée de données et de traitements associées à ces données.

# DEUXIEME PARTIE :

ANALYSE

ET

CONCEPTION

## CHAPITRE I : ANALYSE

### SECTION 1 : PREREQUIS THEORIQUES

#### I. APPROCHE FONCTIONNELLE

Les méthodes fonctionnelles, également qualifiées de méthodes structurées, trouvent leur origine dans les langages procéduraux. Elles mettent en évidence les fonctions à assurer et proposent une approche hiérarchique descendante et modulaire.

Ces méthodes utilisent intensivement les raffinements successifs pour produire des spécifications dont l'essentiel est sous forme de notation graphique en diagrammes de flots de données. Le plus haut niveau représente l'ensemble du problème (sous forme d'activités, de données ou de processus, selon la méthode). Chaque niveau est ensuite décomposé en respectant les entrées et sorties du niveau supérieur. La décomposition se poursuit jusqu'à arriver à des composants maîtrisables.

L'approche fonctionnelle dissocie le problème de la représentation des données, du problème du traitement de ces données. Sur la figure, les données du problème sont représentées sur la gauche. Des flèches transversales matérialisent la manipulation de ces données par des sous fonctions. Cet accès peut être direct (c'est parfois le cas quand les données sont regroupées dans une base de données), ou peut être réalisé par le passage de paramètres depuis le programme principal.

**Exemple de méthode fonctionnelle : MERISE**

#### II. APPROCHE ORIENTEE OBJET

Cette approche considère le logiciel comme une collection d'objets dissociés, identifiés et possédant des caractéristiques bien définies. Une caractéristique est soit un attribut (une donnée caractérisant l'état de l'objet), soit une entité comportementale de l'objet (une fonction). La fonctionnalité du logiciel émerge alors de l'interaction entre les différents objets qui le constituent. L'une des particularités de cette approche est qu'elle rapproche les données et leurs traitements associés au sein d'un unique objet. Ce dernier est caractérisé par plusieurs notions :

- **L'identité** : L'objet possède une identité, qui permet de le distinguer des autres objets, indépendant de son état. On construit généralement cette identité grâce à un identifiant découlant naturellement du problème ;
- **Les attributs** : Il s'agit des données caractérisant l'objet. Ce sont des variables stockant des informations sur l'état de l'objet
- **Les méthodes** : Elles caractérisent le comportement d'un objet, c'est-à-dire l'ensemble des actions appelées opérations qu'il est à mesure de réaliser. Ces opérations permettent de le faire réagir aux sollicitations extérieures (ou d'agir sur les autres objets). De plus, les opérations sont étroitement liées aux attributs, car leurs actions peuvent dépendre des valeurs des attributs, ou bien les modifier. La difficulté de cette modélisation repose sur la création d'une représentation abstraite, sous forme d'objets, d'entités ayant une existence matérielle (chien, voiture, ampoule, personne...) ou bien virtuelle (client, temps...). La Conception Orientée Objet (COO) est la méthode qui conduit à des architectures logicielles fondées sur des objets du système plutôt que sur la fonction qu'il est censé réaliser. En somme, l'approche orientée objet associe les données et leurs traitements. D'autres concepts importants sont spécifiques à cette approche et participent à la qualité du logiciel. Ces concepts sont les suivants :
- **Notion de classe** : Une classe est un type de données abstrait qui précise des caractéristiques (attributs et méthodes) communes à toute une famille d'objets et qui permet d'instancier des objets possédant ces caractéristiques.
- **Encapsulation** : Elle consiste à masquer les détails d'implémentation d'un objet en définissant une interface. Cette dernière est la vue externe d'un objet, elle définit les services accessibles aux utilisateurs. L'encapsulation facilite l'évolution d'une application car elle stabilise l'utilisation des objets : on peut modifier l'implémentation des attributs d'un objet sans modifier son interface, et donc la façon dont il est utilisé. Mais aussi, elle garantit l'intégrité des données, car elle permet d'interdire ou de restreindre, l'accès direct aux attributs des objets ;
- **Héritage, spécialisation, généralisation et polymorphisme** :
  - L'héritage est un mécanisme de transmission des caractéristiques d'une Classe (ses attributs et méthodes) vers une sous-classe. Une classe peut être spécialisée en d'autres classes, afin d'y ajouter des caractéristiques spécifiques ou d'en adapter certaines.

Plusieurs classes peuvent être généralisées en une qui les factorise, afin de regrouper leurs caractéristiques communes ;

- Ainsi, la spécialisation et la généralisation permettent de construire des Hiérarchies de classes. L'héritage peut être simple ou multiple. Elle évite la duplication et encourage la réutilisation ;

- Le polymorphisme représente la faculté d'une méthode à pouvoir S'appliquer à des objets de classes différentes. Il augmente la généricité, et donc la qualité su code.

➤ **Agrégation** : Il s'agit d'une relation entre deux classes, spécifiant que les objets de l'une sont des composants de l'autre. Une relation d'agrégation permet donc de définir des objets composés d'autres objets. L'agrégation permet donc d'assembler des objets de base afin de construire des objets plus complexes ;

➤ **Composition** : Elle est en revanche une variante d'agrégation qui influe sur la structure des instances qu'elle relie. Elle comporte deux caractéristiques suivantes :

- Elle n'est pas partageable, un objet ne peut appartenir qu'à un seul composite  
À la fois ;

- Le cycle de vie de ses parties est fortement lié à celui du composite : la destruction Du composite entraîne en particulier la destruction de ses parties.

### **Exemple de méthode fonctionnelle : UML**

## **III. FORMALISMES DES DIFFERENTS DIAGRAMMES D'UML**

Comme n'importe quel type de projet, un projet informatique nécessite une phase d'analyse, suivi d'une étape de conception. Dans la phase d'analyse, on cherche d'abord à bien comprendre et à décrire de façon précise les besoins des utilisateurs ou des clients. Que souhaitent-ils faire avec le logiciel ? Quelles fonctionnalités veulent-ils ? Pour quel usage ?

Comment l'action devrait-elle fonctionner ? C'est ce qu'on appelle « l'analyse des besoins ».

Après validation de notre compréhension du besoin, nous imaginons la solution. C'est la partie analyse de la solution. Dans la phase de conception, on apporte plus de détails à la solution et on cherche à clarifier des aspects techniques, tels que l'installation des différentes parties logicielles à installer sur du matériel.

Ainsi, UML définit 9 types de diagrammes dans deux catégories de vues, les vues statiques et les vues dynamiques.

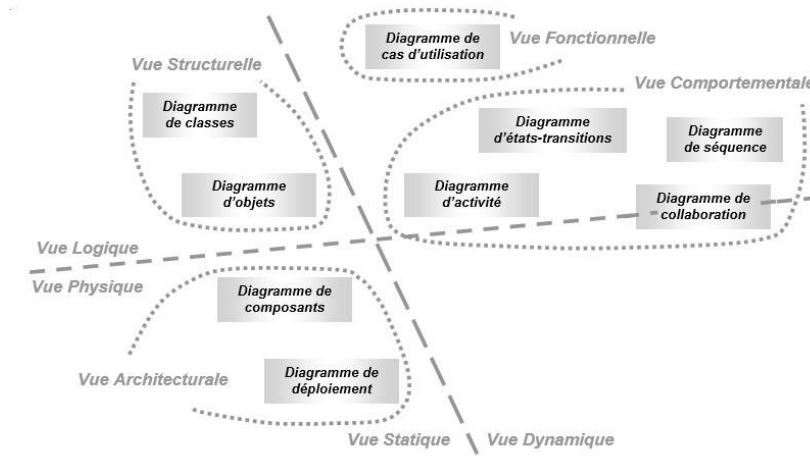


Figure 3 : représentation des différents diagrammes d'UML

Le point de vue de modélisation tel qu'il a été présenté en préambule de ce document est principalement supporté en UML par le choix des vues du sujet qui doivent être décrites et donc par le choix des diagrammes au sein de ses vues. Afin de mieux appréhender la signification et l'intérêt de ces différentes vues on peut en donner les définitions partielles suivantes :

- **Vue logique** : facette théorique du sujet. Cette vue, déconnectée de toute considération matérielle, fournit une description basée sur les rôles, les responsabilités et les relations associées.
- **Vue physique** : aspect organique et matériel du sujet. Cette vue fournit une description basée sur les composants physiques qui implémentent réellement le sujet.
- **Vue statique** : facette relative à l'agencement des composants. Cette vue s'apparente à une description figée du sujet fournissant une photographie de sa physionomie.
- **Vue dynamique** : facette liée à l'activité du sujet. Cette vue fournit une représentation vivante du sujet et décrit ses comportements et ses réactions.
- **Vue fonctionnelle** : facette relative aux contextes d'activité du sujet. Cette vue particulière relève d'une description externe du sujet en tant que fournisseurs de services à l'environnement et aux utilisateurs.

- **Vue structurelle** : aspect organisationnel logique du sujet. Cette vue décrit la structuration des différents constituants fonctionnels du système.
- **Vue comportementale** : facette liée au fonctionnement du sujet et de ses composants. Cette vue fournit une description des actions, processus et modes de fonctionnement sur divers niveaux de granularité.
- **Vue architecturale** : aspect organisationnel physique du sujet. Cette vue décrit la structuration des différents composants matériels qui constituent organiquement le sujet

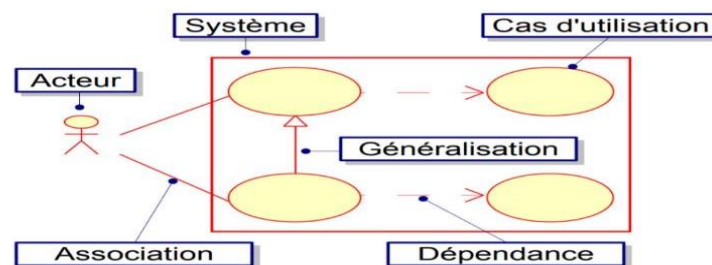
## **A. VUES STATIQUES DU SYSTEME**

### **1. DIAGRAMME DE CAS D'UTILISATION**

Les use cases permettent de structurer les besoins des utilisateurs et les objectifs correspondants d'un système. Ils centrent l'expression des exigences du système sur ses utilisateurs : ils partent du principe que les objectifs du système sont tous motivés.

La détermination et la compréhension des besoins sont souvent difficiles car les intervenants sont noyés sous de trop grandes quantités d'informations : il faut clarifier et organiser les besoins des clients (les modéliser). Pour cela, les cas d'utilisation identifient les utilisateurs du système (acteurs) et leurs interactions avec le système. Ils permettent de classer les acteurs et structurer les objectifs du système.

Les use cases ne doivent donc en aucun cas décrire des solutions d'implémentation. Leur but est justement d'éviter de tomber dans la dérive d'une approche fonctionnelle, où l'on liste une litanie de fonctions que le système doit réaliser.

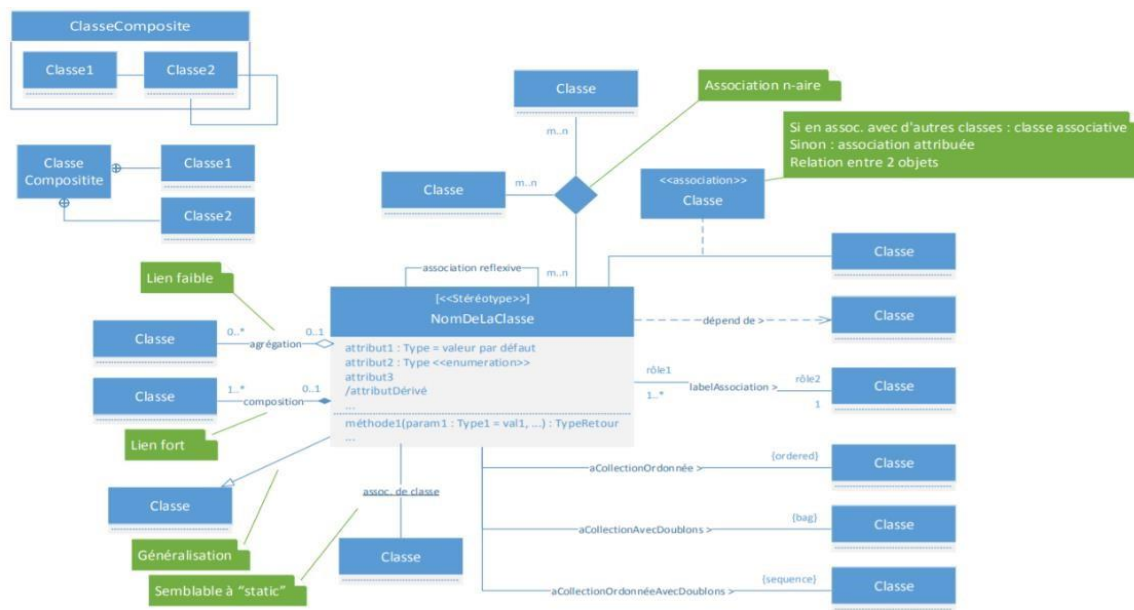


**Figure 4 : formalisme d'un diagramme de cas d'utilisation**

## 2. DIAGRAMME DE CLASSES

Le diagramme de classes exprime la structure statique du système en termes de classes et de relations entre ces classes.

L'intérêt du diagramme de classe est de modéliser les entités du système d'information, le diagramme de classe permet de représenter l'ensemble des informations finalisées qui sont gérées par le domaine. Ces informations sont structurées, c'est-à-dire qu'elles ont regroupées dans des classes. Le diagramme met en évidence d'éventuelles relations entre ces classes.

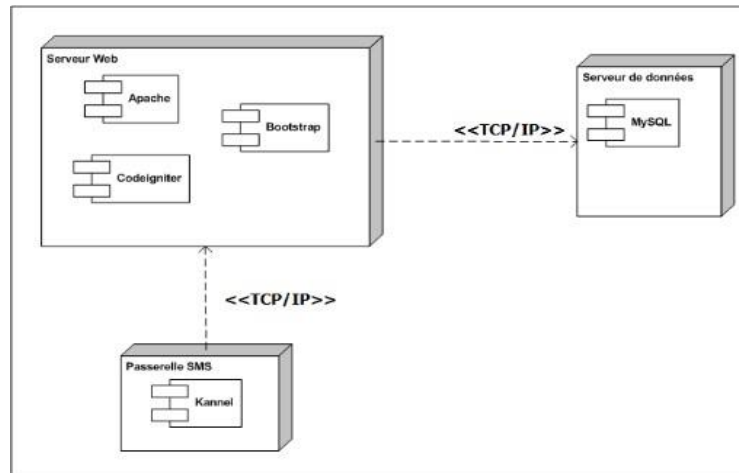


**Figure 5 : formalisme d'un diagramme de classe**

## 3. DIAGRAMMES DE DEPLOIEMENT

Les diagrammes de déploiement montrent la disposition physique des différents matériels (les nœuds) qui entrent dans la composition d'un système et la répartition des instances de composants, processus et objets qui « vivent » sur ces matériels.

Les diagrammes de déploiement sont donc très utiles pour modéliser l'architecture physique d'un système.



**Figure 6 : exemple de diagramme de déploiement**

## **B. VUES DYNAMIQUES DU SYSTEME**

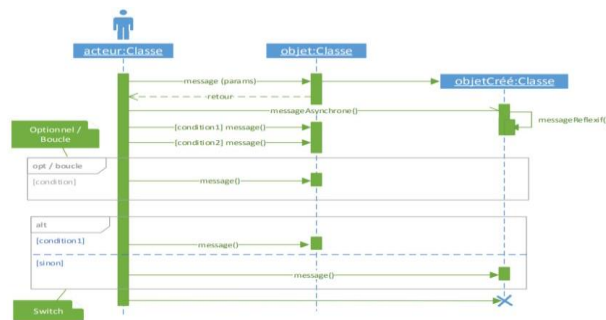
### **1. DIAGRAMMES DE SEQUENCE**

Le diagramme de séquence est une variante du diagramme de collaboration. Par opposition aux diagrammes de collaboration, les diagrammes de séquence possèdent intrinsèquement une dimension temporelle mais ne représentent pas explicitement les liens entre les objets. Ils privilégient ainsi la représentation temporelle à la représentation spatiale et sont plus aptes à modéliser les aspects dynamiques du système. En revanche, ils ne rendent pas compte du contexte des objets de manière explicite, comme les diagrammes de collaboration.

Le diagramme de séquence permet de visualiser les messages par une lecture de haut en bas.

L'axe vertical représente le temps, l'axe horizontal les objets qui collaborent. Une ligne verticale en pointillés est attachée à chaque objet et représente sa durée de vie.

Les messages sont représentés comme dans le diagramme de collaboration. (NB : un message de retour sera représenté avec des traits en pointillés)

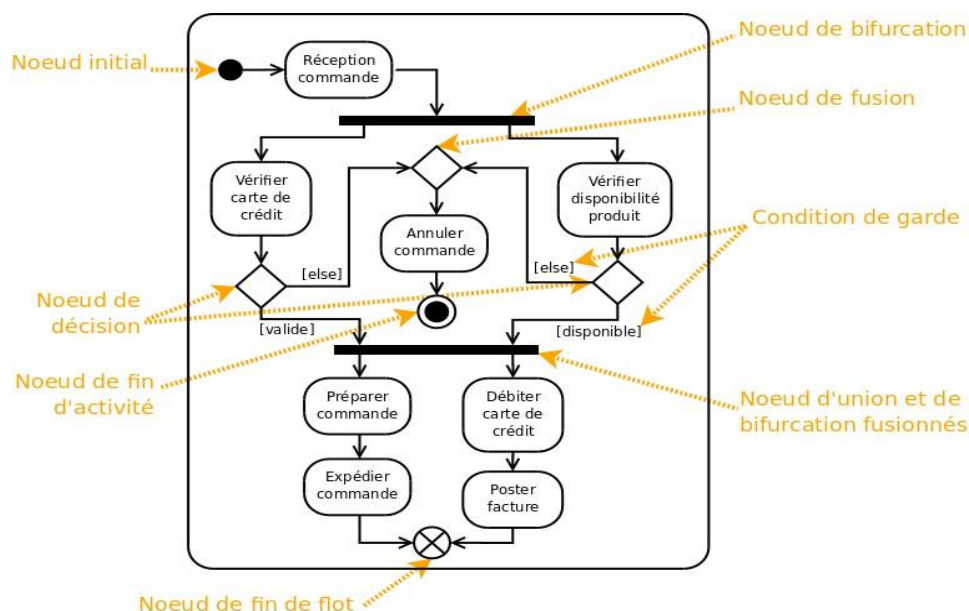


**Figure 7 : formalisme d'un diagramme de séquence**

## 2. DIAGRAMMES D'ACTIVITES

Le diagramme d'activité est attaché à une catégorie de classe et décrit le déroulement des activités de cette catégorie. Le déroulement s'appelle "flot de contrôle". Il indique la part prise par chaque objet dans l'exécution d'un travail. Il sera enrichi par les conditions de séquençement. Il pourra comporter des synchronisations pour représenter les déroulements parallèles.

La notion de couloir d'activité va décrire les responsabilités en répartissant les activités entre les différents acteurs opérationnels.



**Figure 8 : exemple de diagramme d'activité**

## **SECTION 2 : ANALYSE DU SYSTEME**

### **I. RECENCEMENT ET ANALYSE DES BESOINS**

#### **A. BESOINS FONCTIONNELS**

Après une étude détaillée de système, cette partie est réservée à la description des exigences fonctionnelles des différents acteurs de l'application. Ces besoins se regroupent dans les diagrammes des cas d'utilisation.

Les besoins de l'état civil :

- ✓ Gestion des Utilisateurs
- ✓ Gestion des naissances
- ✓ Gestion des mariages
- ✓ Gestion des décès

#### **B. BESOINS NON FONCTIONNELS**

Les besoins non fonctionnels sont indispensables et permettent l'amélioration de la qualité du progiciel. Ils agissent comme des contraintes sur les solutions, mais leur prise en considération fait éviter plusieurs incohérences dans le système. Ce dernier doit répondre aux exigences suivantes :

- Authentification : le système doit permettre à l'utilisateur de saisir son login et son mot de passe pour l'accès. Cette opération assure la sécurité du système et limite le nombre d'utilisateurs.
- Ergonomie : le système doit offrir aux utilisateurs une interface qui soit le plus riche possible afin de limiter le nombre d'écrans. Par ailleurs, l'interactivité devra être adaptée (usage du clavier, menu, etc.)

### **II. DESCRIPTIONS DES PROCESSUS**

#### **❖ Processus de déclaration de naissance**

Il est obligatoire de porter à la connaissance de l'officier de l'état civil aux fins d'enregistrement dans les registres de naissances. Cette déclaration permet à la personne de devenir citoyen. Cette démarche est obligatoire à la naissance de l'enfant mais peut être accomplie à tout moment de la vie.

La déclaration de naissance permet d'obtenir immédiatement et gratuitement le volet n°1 de l'acte de naissance et l'inscription de la naissance sur le livret de famille.

### **Qui peut déclarer la naissance ?**

- Le Père ou la mère
- Les ascendants ou les proches parents
- Le médecin, la sage-femme ou toute personne ayant assistée à l'accouchement
- Le chef de village ou délégué de quartier
- Le Procureur de la République

### **Quels sont les documents à fournir ?**

Le déclarant doit se présenter muni de sa pièce d'identité et celles des parents, d'un certificat d'accouchement ou les références de deux témoins majeurs ayant assistés à l'accouchement et du livret de famille s'il en dispose.

**NB :** Les deux témoins doivent se présenter avec le déclarant au moment de la déclaration munis de leurs pièces d'identités.

**NB :** A titre exceptionnel, le certificat d'accouchement peut être remplacé par un certificat administratif pour déclaration de naissance obtenu dans les brigades de gendarmerie.

### **Quand déclarer la naissance ?**

Il existe trois (3) formes de déclarations : déclaration normale, déclaration tardive, et déclaration après jugement.

#### **✓ Déclaration Normale :**

Elle doit se faire dans un délai maximum d'un mois et 15 jours après la naissance selon la loi. Le déclarant se présente auprès du service de l'état civil muni de sa carte nationale

d'identité (CNI) avec ou sans les cartes nationales d'identité des parents et les documents justificatifs de la naissance.

**NB :** Si les parents n'ont pas de pièces d'identités, les deux témoins doivent se présenter avec le déclarant au moment de la déclaration munis de leurs pièces d'identités.

**✓ Déclaration tardive :**

Lorsqu'un mois et quinze jours se sont écoulés depuis une naissance sans qu'elle ait fait l'objet d'une déclaration, l'agent d'état civil peut néanmoins enregistrer une déclaration tardive. Pour faire une déclaration tardive le déclarant est tenu d'apporter avec lui les C.N.I des deux témoins, en plus des informations pour une déclaration normale, les numéros d'identification nationaux (N.I.N) des deux témoins seront enregistrés sur l'acte et en tête de l'acte dressé tardivement doit être mentionné « inscription de déclaration tardive »

**NB :** Si la déclaration tardive concerne une naissance de l'année précédente, ses mentions seront portées sur le registre de l'année en cours.

**✓ Déclaration après jugement d'autorisation d'inscription :**

Passé le délai d'un an après la naissance, l'officier de l'état civil ne peut dresser un acte d'état civil que s'il y est autorisé par une décision du juge de paix rendue. Dans ce cas, le déclarant apporte avec lui, en plus des informations concernant une déclaration normale, un justificatif de l'avis du tribunal régional. L'officier d'état civil enregistrera sur l'acte les informations de la déclaration normale, le numéro de jugement d'autorisation et le date du jugement.

**❖ Processus de déclaration des décès**

**Qui peut faire la démarche ?**

- Les proches parents du défunt ou tout autre personne possédant sur l'état civil du défunt les renseignements nécessaires à l'établissement de l'acte de décès.
- Le chef de village ou délégué de quartier
- Les autorités administratives ou judiciaires

**NB :** de 30 à 45 jours après le décès, les délégués ou chefs de villages sont tenus de faire à l'officier de l'état civil les déclarations omises par les parents. Cette démarche est obligatoire même pour les étrangers décédés au Sénégal.

### **Quels sont les documents à fournir ?**

Le déclarant doit se présenter muni de sa pièce d'identité d'un certificat de genre du mort ou les références de deux témoins majeurs et du livret de famille s'il en dispose.

**NB :** Les deux témoins doivent se présenter avec le déclarant au moment de la déclaration munis de leurs pièces d'identités.

**NB :** En cas de mort violente (accident, meurtre), fournir aussi le procès-verbal de constat de décès établi par l'officier de police judiciaire.

### **Quand faire la déclaration ?**

Elle peut être faite du jour du décès à un an après le décès. Lorsqu'elle est faite au-delà du 45ème jour, elle est appelée déclaration tardive.

Au-delà d'un an il faut demander un jugement d'autorisation d'inscription auprès du tribunal

Le procureur de la république peut à toute époque et en dehors des délais prévus faire la déclaration d'un décès dont il aurait connaissance et qui n'aurait pas été enregistrée

Il existe trois (3) types de déclarations : déclaration normale, déclaration tardive, et déclaration après jugement.

#### **✓ Déclaration Normale :**

- L'année, la date, l'heure, et le lieu du décès.
- N.I.N(s'il y'a lieu), Prénom, Nom, Domicile, Profession de la personne décédée
- N.I.N(s'il y'a lieu), Prénom, Nom, Domicile, Profession de ses parents,
- Les prénoms, nom du ou des conjoints si la personne décédée était mariée, divorcée ou veuve (ou un certificat de mariage)
- N.I.N(s'il y'a lieu), Prénom, Nom, Domicile, Profession du déclarant et son degré de parentés avec la personne décédée (s'il y'a lieu).

#### **✓ Déclaration tardive :**

En plus des informations pour une déclaration normale, les numéros d'identification nationaux (N.I.N) des témoins seront enregistrés

**✓ Déclaration après jugement d'autorisation d'inscription :**

Dans ce cas, le déclarant apporte avec lui, en plus des informations concernant une déclaration normale, un justificatif de l'avis du tribunal régional. L'officier d'état civil enregistrera sur l'acte les informations de la déclaration normale, le numéro de jugement d'autorisation et le date du jugement.

**❖ Processus de déclaration des mariages :**

Lorsque deux personnes souhaitent se marier elles peuvent demander à ce que le mariage soit célébré ou constaté.

Dans le cas où le mariage est célébré publiquement par un officier de l'état civil du centre dont relève le lieu de domicile ou résidence d'un des deux conjoints, l'acte constatant le mariage est immédiatement établi par l'officier qui en délivre le volet 1 et deux livrets de familles.

Dans le cas où le mariage est constaté on note trois (3) types de déclarations :

Déclaration normale, déclaration tardive, déclaration sur jugement. Pratiquement elles se font de la même façon que celle de la déclaration des naissances et des décès seuls les délais changent.

**✓ Déclaration Normale :**

Elle doit se faire dans un délai maximum d'un mois et 15 jours après le mariage. Les informations à enregistrer sur le registre des actes de mariage sont :

- N.I.N, Prénom, Nom, Domicile, Profession des époux.
- N.I.N, Prénom, Nom, Domicile, Profession de leurs parents,
- La convention des époux du paiement d'une dot sous condition du mariage,
- Le choix du régime matrimonial,
- Les prénoms, nom du ou des précédents conjoints de chacun des époux (s'il y'a lieu)
- N.I.N, Prénom, Nom, Domicile, Profession des témoins.

✓ **Déclaration tardive :**

Dans un délai de maximum six (6) mois à partir de la date du mariage, En plus de ces informations citées ci-dessus, l'officier de l'état civil mentionnent en tête de l'acte « déclaration tardive » et fait mention en marge que les témoins attestent avoir assisté à l'échange des consentements et à la conclusion du mariage.

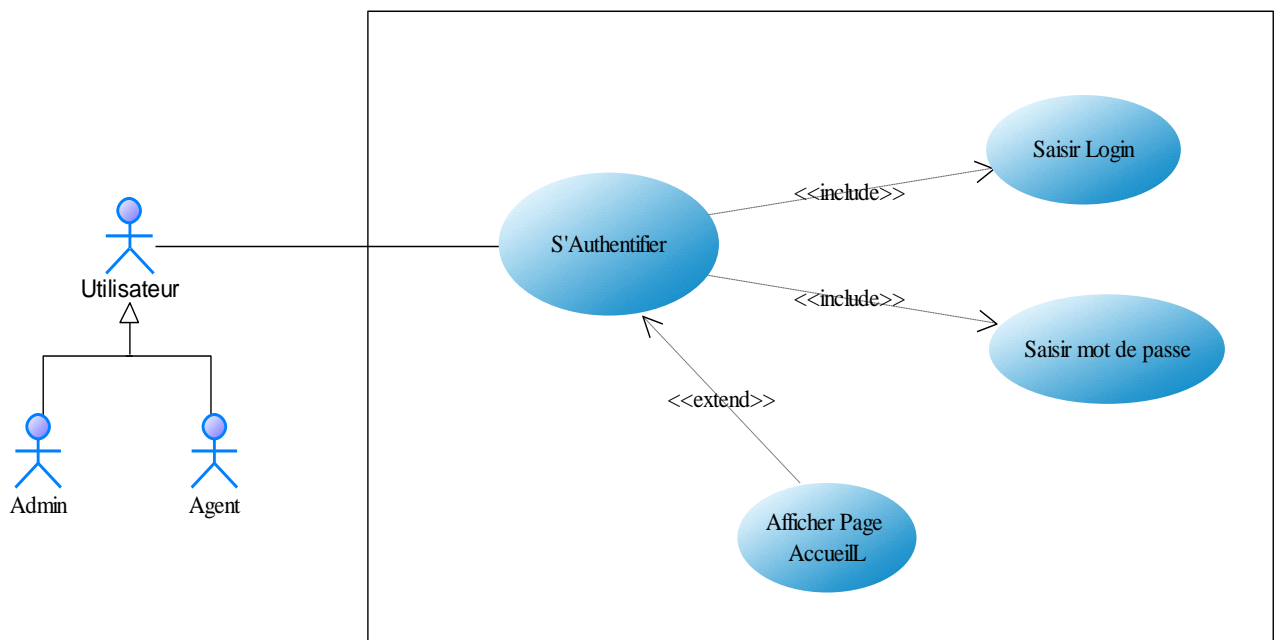
✓ **Déclaration après jugement d'autorisation d'inscription :**

Après six (6) mois de la date du mariage, en plus de ces informations citées ci-dessus, L'officier d'état civil enregistrera sur l'acte les informations de la déclaration normale, le numéro de jugement d'autorisation et le date du jugement.

NB : Pour chaque déclaration (naissance, mariage, décès), sera attribué au concerne un numéro de déclaration (N° registre) qui lui est unique. Après enregistrement de la déclaration dans les registres, l'officier délivré au solliciteur le volet 1, l'extrait et le bulletin de l'acte

### III. DIAGRAMME USE CASE

#### CAS 1 : S'AUTHTENTIFIER



**Figure 09 : Diagramme de cas d'utilisation « S'Authentifier »**

CAS 2 : Gestion des Naissance

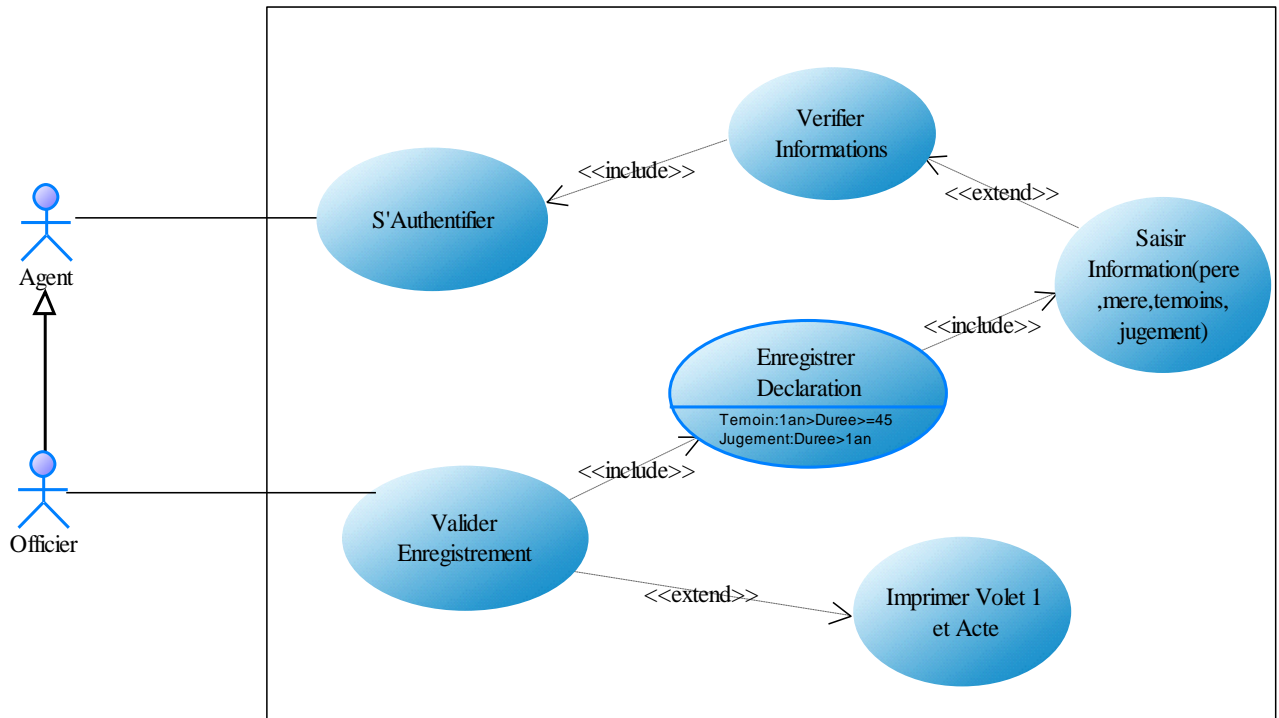


Figure 10 : Diagramme de cas d'utilisation « Gestion des Naissances »

CAS 3 : Gestion des Mariages

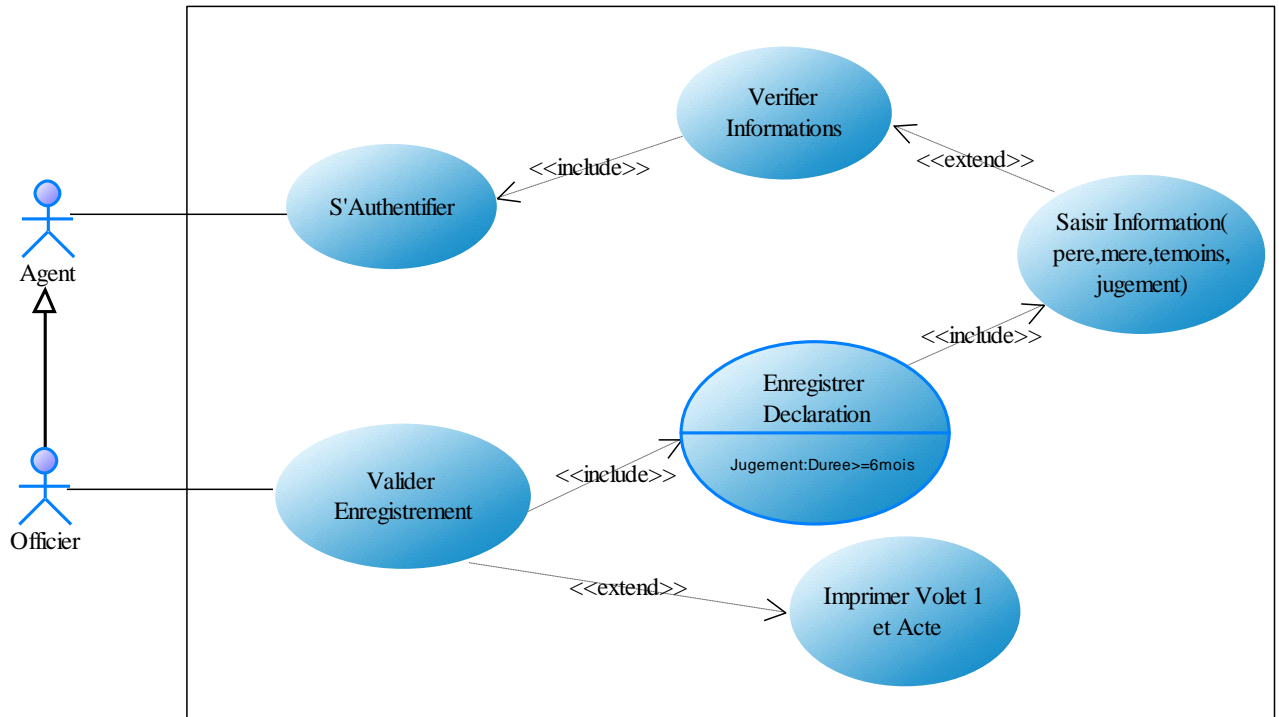


Figure 11 : Diagramme de cas d'utilisation « Gestion des Mariages »

CAS 4 : Gestion des Décès

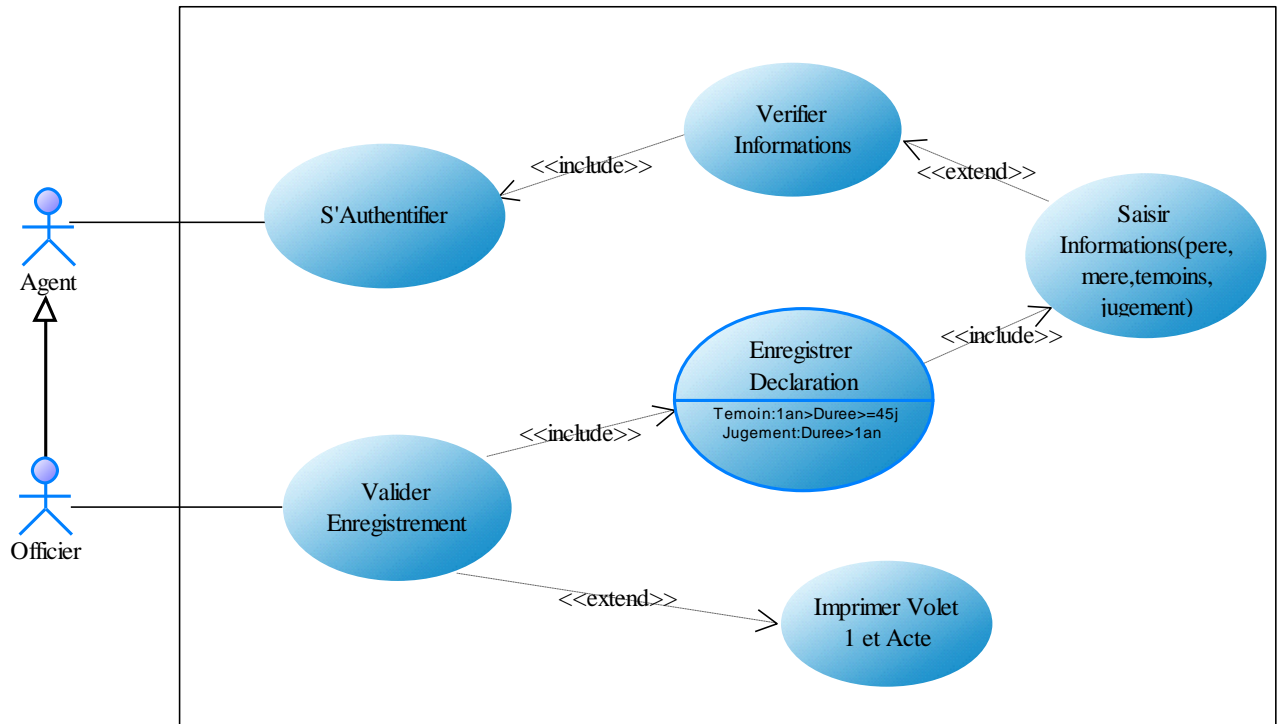


Figure 12 : Diagramme de cas d'utilisation <<Gestion des Décès>>

CAS 5 : Demande Pièces

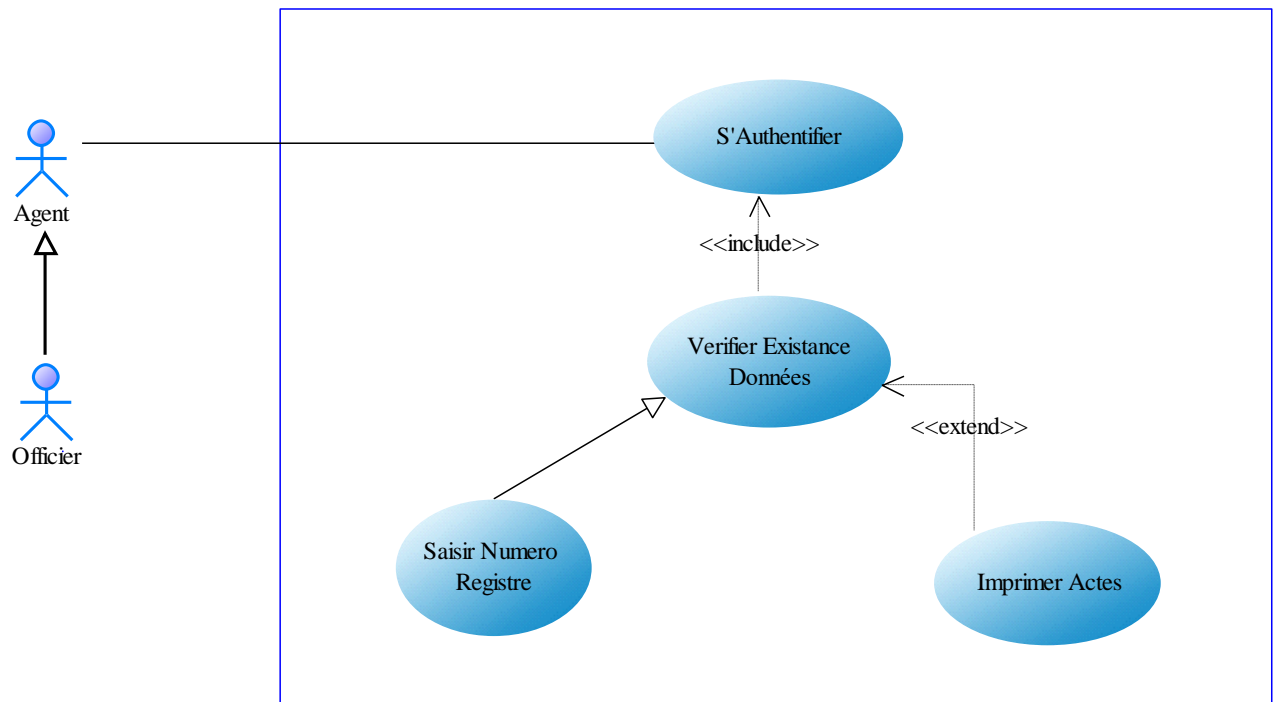


Figure 13 : Diagramme de cas d'utilisation << Demande Pièces >>

## CHAPITRE II : CONCEPTION

### SECTION 1 : DIAGRAMMES DE SEQUENCE

#### CAS 1 : AUTHENTIFICATION

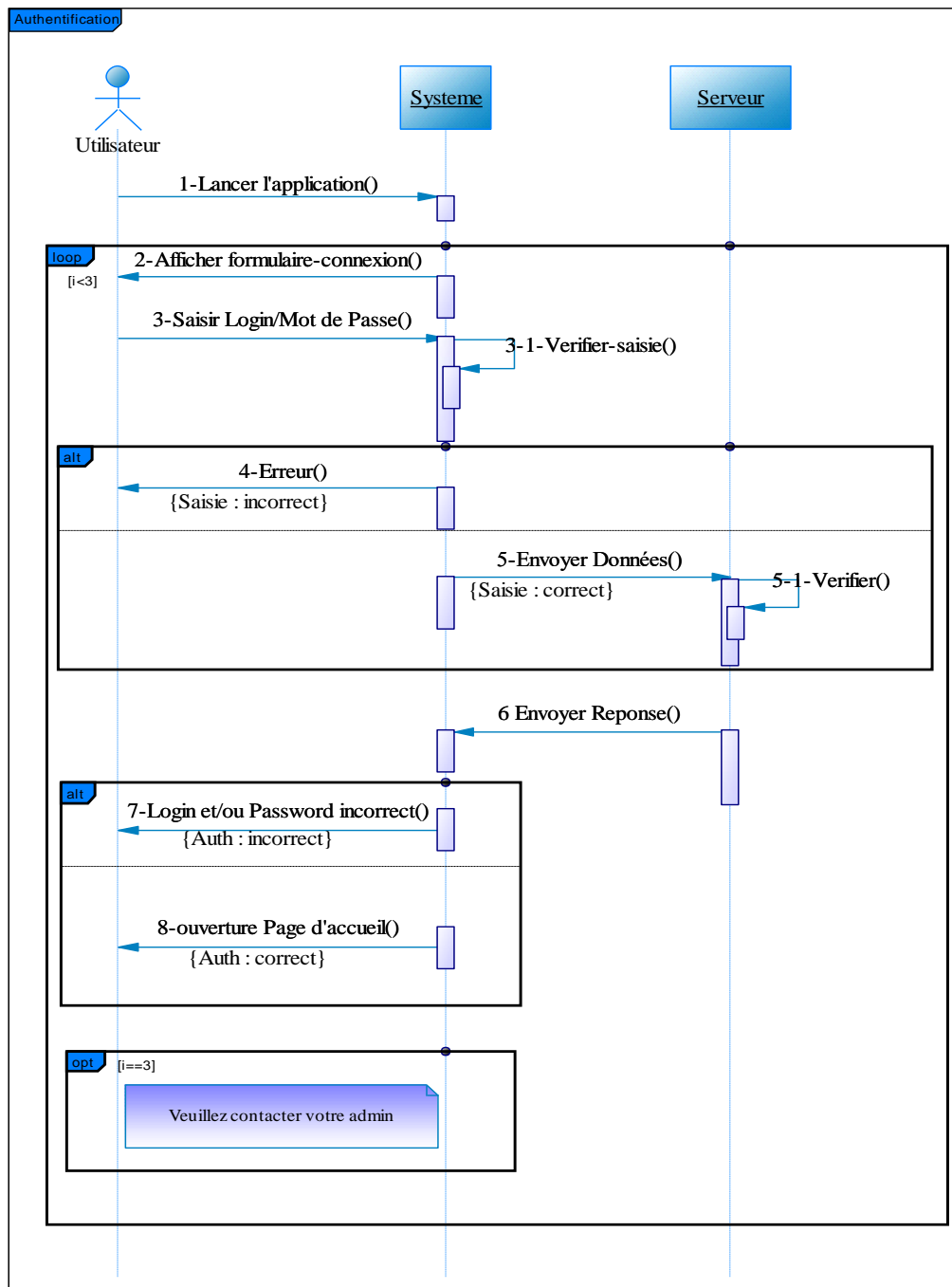
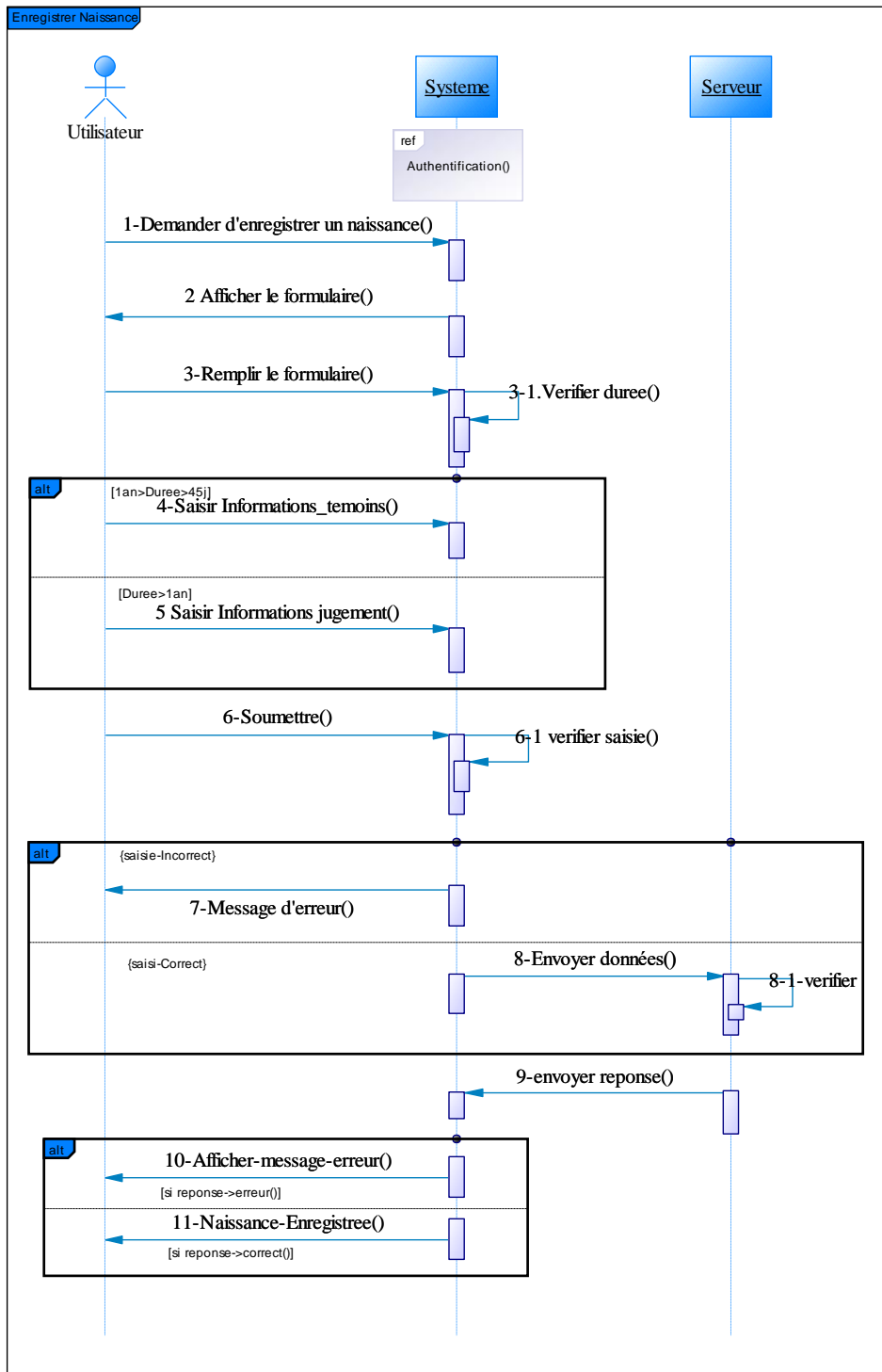


Figure 14 : Diagramme de séquence <<Authentication>>

**Tableau I : Description textuelle « Authentification »**

<b>Cas d'utilisation</b>	S'Authentifier
<b>Scenario principal</b>	Authentification
<b>Auteur</b>	Utilisateur
<b>Précondition</b>	L'interface d'authentification doit être affichée
<b>Scenario nominal</b>	<ol style="list-style-type: none"> <li>1. L'utilisateur lance l'application</li> <li>2. Le système affiche la page d'authentification</li> <li>3. L'utilisateur doit saisir son nom d'utilisateur et son mot de passe et demander l'accès à l'application. <ol style="list-style-type: none"> <li>3.1. Le système contrôle ses informations.</li> </ol> </li> <li>4. Si les informations sont incorrectes le système affiche un message d'erreur.</li> <li>5. Si les informations sont correctes, le système envoie l'email et le mot de passe au serveur. <ol style="list-style-type: none"> <li>5.1. Un test doit être réalisé, celui d'existence et de compatibilité de l'email, le mot de passe.</li> </ol> </li> <li>6. Le serveur envoie le résultat de vérification du système</li> <li>7. Si l'authentification est incorrecte, le système lui signale en affichant un message d'erreur.</li> <li>8. Sinon, il accède à la page d'accueil.</li> <li>9. Après 3 tentatives le système lui demande de contactez l'admin.</li> </ol>
<b>Post condition</b>	L'utilisateur s'est connecté.

## CAS 2 : ENREGISTRER-NAISSANCE

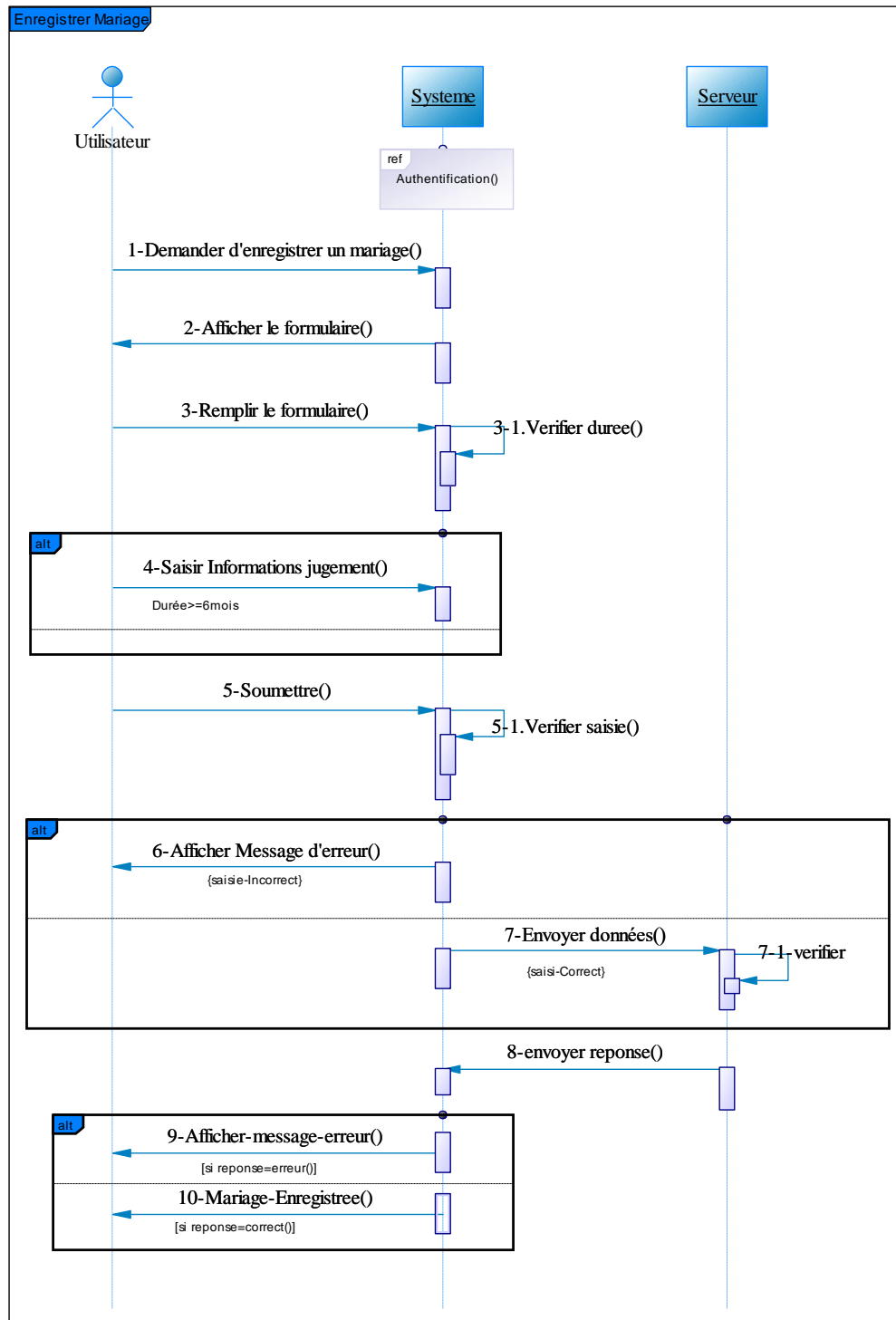


**Figure 15 : Diagramme de séquence <<Enregistrer-Naissance>>**

**Tableau II : Description textuelle « Enregistrer Naissance »**

<b>Cas d'utilisation</b>	Gestion des Naissances
<b>Scenario principal</b>	Enregistrer-naissance
<b>Auteur</b>	Agent ou Officier
<b>Précondition</b>	L'agent doit être authentifié
<b>Scenario nominal</b>	<ol style="list-style-type: none"> <li>1. L'agent demande à enregistrer une naissance</li> <li>2. Le système affiche le formulaire</li> <li>3. L'agent doit saisir les informations demandées. <ol style="list-style-type: none"> <li>3.1. Le système contrôle la durée entre la date de naissance et la date de déclaration.</li> </ol> </li> <li>4. . Si la 1an&lt;durée, l'utilisateur doit saisir les informations du jugement.</li> <li>5. L'utilisateur soumet la saisie <ol style="list-style-type: none"> <li>6-1. Le système vérifie la saisie</li> </ol> </li> <li>6. S'il y a erreur, le système le lui signale en affichant un message d'erreur.</li> <li>7. Sinon, le système envoie les données au serveur</li> <li>8. Le serveur envoie le résultat au système.</li> <li>9. Le système affiche une notification</li> </ol>
<b>Post condition</b>	Naissance enregistrée

**CAS 3 : ENREGISTRER-MARIAGE**

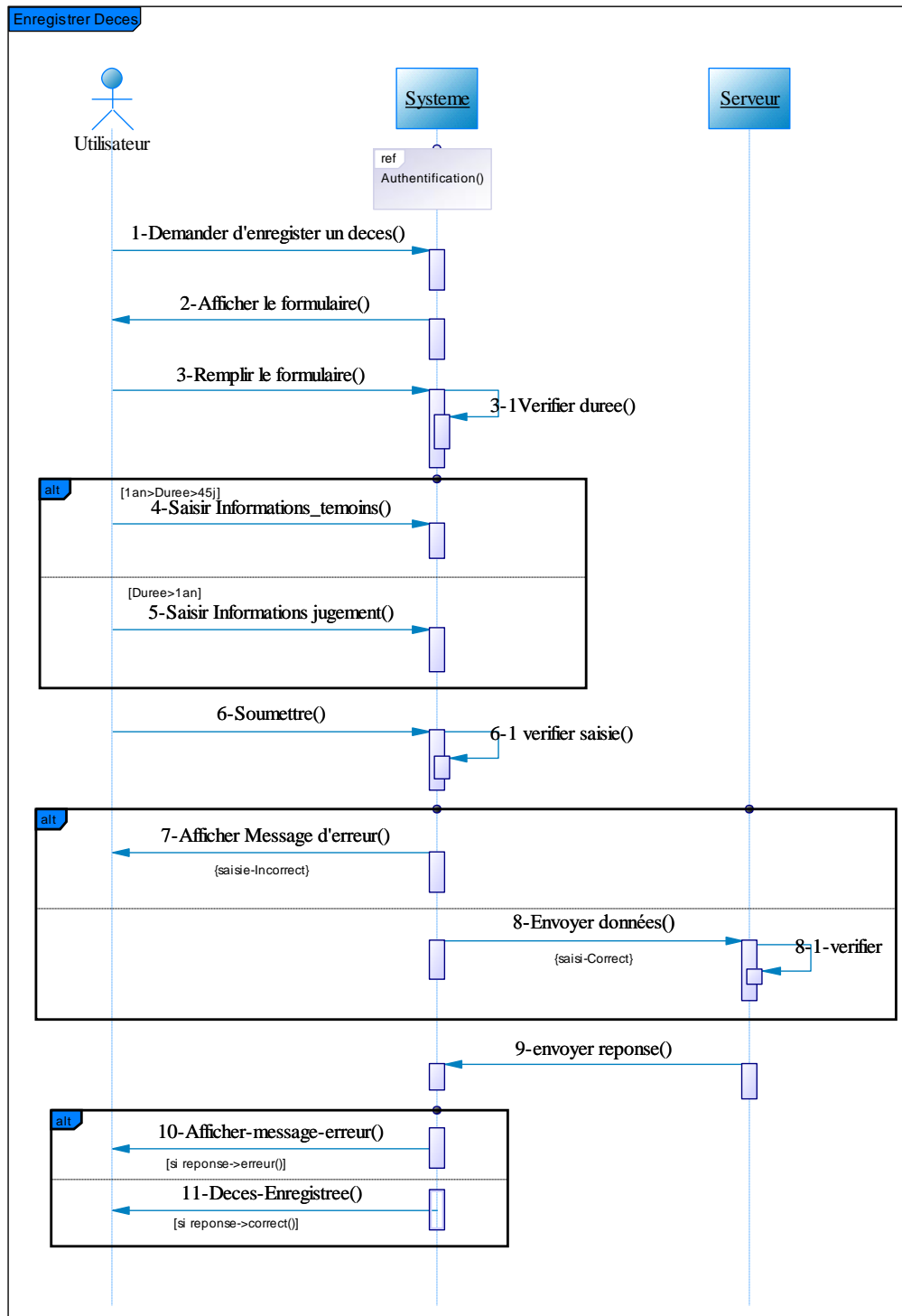


**Figure 16 : Diagramme de séquence <<Enregistrer-Mariage>>**

**Tableau III : Description textuelle « Enregistrer Mariage »**

<b>Cas d'utilisation</b>	Gestion des Mariages
<b>Scenario principal</b>	Enregistrer-mariage
<b>Auteur</b>	Utilisateur
<b>Précondition</b>	L'utilisateur doit être authentifié
<b>Scenario nominal</b>	<ol style="list-style-type: none"> <li>1. L'utilisateur demande à enregistrer un mariage</li> <li>2. Le système affiche le formulaire</li> <li>3. L'agent doit saisir les informations demandées. <ol style="list-style-type: none"> <li>3.1. Le système contrôle la durée entre la date du mariage et la date de déclaration.</li> </ol> </li> <li>4. Si la durée &gt; 6 mois, l'agent doit saisir les informations du jugement.</li> <li>5. Si la 45j &lt; durée &lt; 6 mois, l'agent doit saisir une mention.</li> <li>6. L'agent soumet la saisie <ol style="list-style-type: none"> <li>5-1. Le système vérifie la saisie</li> </ol> </li> <li>7. S'il y a erreur, le système le lui signale en affichant un message d'erreur.</li> <li>8. Sinon, le système envoie les données au serveur</li> <li>9. Le système affiche une notification</li> </ol>
<b>Post condition</b>	Mariage enregistré

**CAS 4 : ENREGISTRER-DECES**

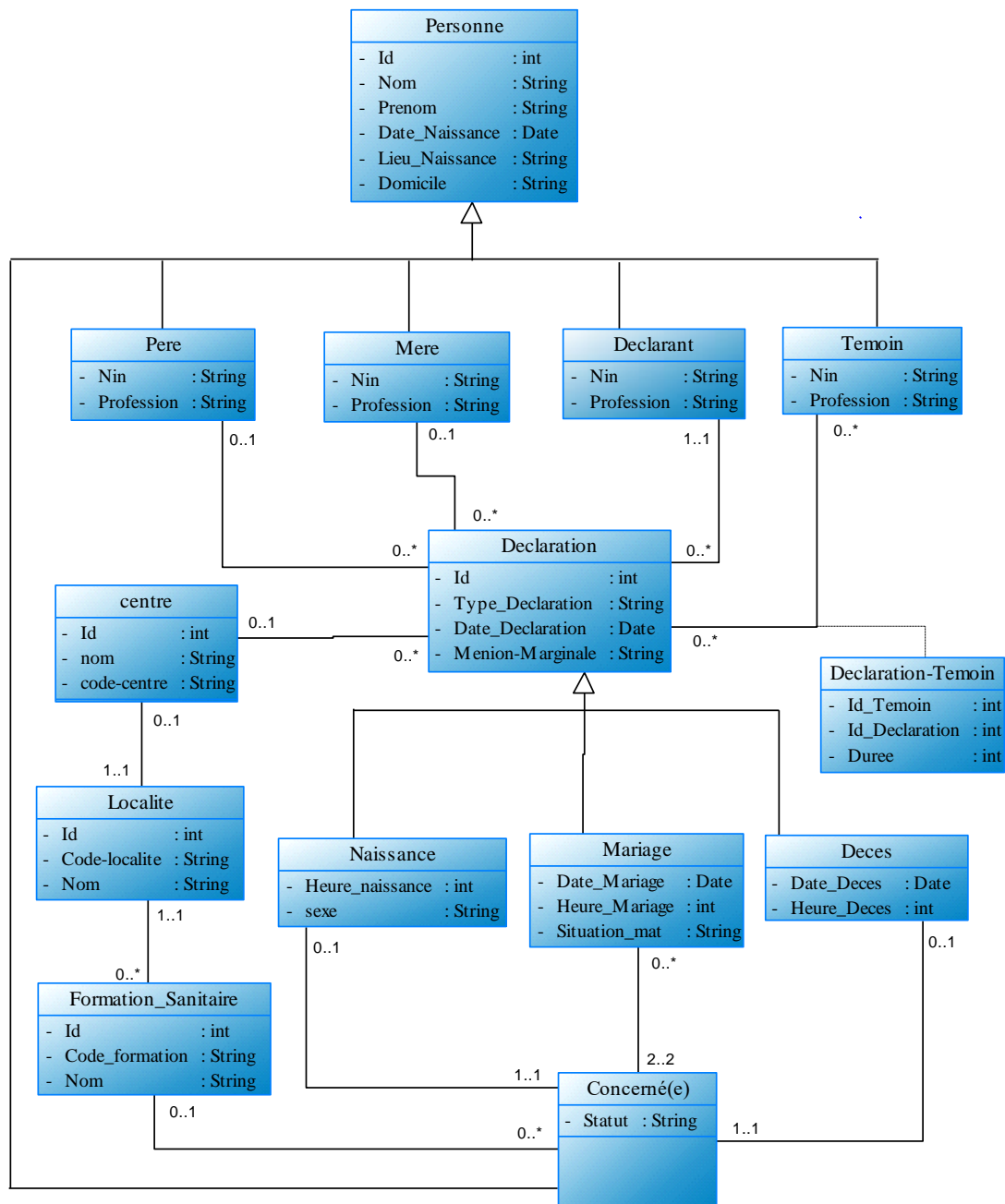


**Figure 17 : Diagramme de séquence « Enregistrer-Décès »**

**Tableau IV : Description textuelle « Enregistrer Décès »**

<b>Cas d'utilisation</b>	Gestion des Décès
<b>Scenario principal</b>	Enregistrer-décès
<b>Auteur</b>	Utilisateur
<b>Précondition</b>	L'utilisateur doit être authentifié
<b>Scenario nominal</b>	<ol style="list-style-type: none"> <li>1. L'utilisateur demande à enregistrer un décès</li> <li>2. Le système affiche le formulaire</li> <li>3. L'utilisateur doit saisir les informations demandées. <ol style="list-style-type: none"> <li>3.1. Le système contrôle la durée entre la date de décès et la date de déclaration.</li> </ol> </li> <li>4. Si la 1an&gt;durée&gt;45j, l'utilisateur doit saisir les informations des témoins.</li> <li>5. . Si la 1an&lt;durée, l'utilisateur doit saisir les informations du jugement.</li> <li>6. L'utilisateur soumet la saisie <ol style="list-style-type: none"> <li>6-1. Le système vérifie la saisie</li> </ol> </li> <li>7. S'il y a erreur, le système le lui signale en affichant un message d'erreur.</li> <li>8. Sinon, le système envoie les données au serveur <ol style="list-style-type: none"> <li>8-1. Le serveur effectue une vérification.</li> </ol> </li> <li>9. Le serveur envoie le résultat au système.</li> <li>10. S'il y'a erreur le système le lui signale en affichant un message d'erreur</li> <li>11. Sinon l'utilisateur est informe que la saisie est enregistrée et envoyée à l'officier ou son cessionnaire pour la validation.</li> </ol>
<b>Post condition</b>	Décès enregistré

### SECTION 3 : DIAGRAMMES DE CLASSE



**Figure 18 : Diagramme de Classe**

# TROISIEME PARTIE :

## IMPLEMENTATION

## ET

## DEPLOIEMENT

## CHAPITRE I : IMPLEMENTATION

### SECTION 1 : ENVIRONNEMENT

#### A. Eclipse



Eclipse IDE est un environnement de développement intégré libre (le terme Eclipse désigne également le projet correspondant, lancé par IBM) extensible, universel et polyvalent, permettant potentiellement de créer des projets de développement mettant en œuvre n'importe quel langage de programmation. Eclipse IDE est principalement écrit en Java (à l'aide de la bibliothèque graphique SWT, d'IBM), et ce langage, grâce à des bibliothèques spécifiques, est également utilisé pour écrire des extensions.

#### B. Visual Studio



Visual Studio est un environnement de développement intégré (IDE) développé par Microsoft pour développer une interface utilisateur graphique (GUI), une console, des applications Web, des applications Web, des applications mobiles, un cloud et des services Web, etc. Grâce à cet IDE, vous pouvez créer le code géré ainsi que le code natif. Il utilise les différentes plateformes de logiciels de développement logiciel Microsoft, telles que Windows Store, Microsoft Silverlight, Windows API, etc. JavaScript et beaucoup d'autres langues. Il fournit un support pour 36 langages de programmation différents. Il est disponible pour Windows ainsi que pour MacOS.

### SECTION 2 : ETUDE DE QUELQUES LANGAGES DE PROGRAMMATION

#### I. DEFINITION

La programmation dans le domaine informatique est l'ensemble des activités qui permettent l'écriture des programmes informatiques. C'est une étape importante de la conception de logiciel .

Pour écrire le résultat de cette activité, on utilise un langage de programmation.

## II. EXEMPLE DE QUELQUES LANGAGES DE PROGRAMMATION

### A. C



Le langage C a été inventé au cours de l'année 1972 dans les Laboratoires Bell. Il était développé en même temps que UNIX par Dennis Ritchie et Ken Thompson. Ken Thompson avait développé un prédécesseur de C, le langage B, qui est lui-même inspiré de BCPL. Dennis Ritchie a fait évoluer le langage B dans une nouvelle version suffisamment différente, en ajoutant notamment les types, pour qu'elle soit appelée C.

C'est un langage de programmation impératif et généraliste. Il est qualifié de langage de bas niveau dans le sens où chaque instruction du langage est conçue pour être compilée en un nombre d'instructions machine assez prévisibles en termes d'occupation mémoire et de charge de calcul. En outre, il propose un éventail de types entiers et flottants conçus pour pouvoir correspondre directement aux types de données supportés par le processeur. Enfin, il fait un usage intensif des calculs d'adresse mémoire avec la notion de pointeur.

Hormis les types de base, C supporte les types énumérés, composés, et opaques. Il ne propose en revanche aucune opération qui traite directement des objets de plus haut niveau (fichier informatique, chaîne de caractères, liste, table de hachage...). Ces types plus évolués doivent être traités en manipulant des pointeurs et des types composés. De même, le langage ne propose pas en standard la gestion de la programmation orientée objet, ni de système de gestion d'exceptions. Il existe des fonctions standards pour gérer les entrées-sorties et les chaînes de caractères, mais contrairement à d'autres langages, aucun opérateur spécifique pour améliorer l'ergonomie. Ceci rend aisé le remplacement des fonctions standards par des fonctions spécifiquement conçues pour un programme donné.

### B. C++



Le langage C++ est une amélioration du langage C. Bjarne Stroustrup, un ingénieur considéré comme l'inventeur du C++, a en effet décidé d'ajouter au langage C les propriétés de l'approche orientée objet (POO). Ce nouveau langage apparaît vers la fin des années 80 avec comme nom

« C with classes ». Celui-ci a ensuite été renommé en C++, clin d'œil au symbole d'incrémentation ++ du langage C, afin de signaler qu'il s'agit d'un langage C amélioré. C++ est une incrémentation de C, donc une amélioration.

Le langage C++ est un des langages les plus célèbres au monde. Très utilisé, notamment dans le secteur des jeux vidéo qui apprécie ses performances et ses possibilités, le C++ est désormais incontournable pour les développeurs.

Il y'a aussi une bibliothèque appelée Qt. Elle ajoute un très large éventail de possibilités au C++ : elle permet de créer des fenêtres et menus, mais aussi d'utiliser les fonctionnalités réseau de votre ordinateur.

### C.

Java est un langage de programmation orienté objet créé par James Gosling et Patrick Naughton, employés de Sun Microsystems, avec le soutien de Bill Joy (cofondateur de Sun Microsystems en 1982), présenté officiellement le 23 mai 1995 au SunWorld.

La société Sun a été ensuite rachetée en 2009 par la société Oracle qui détient et maintient désormais Java.

La particularité et l'objectif central de Java est que les logiciels écrits dans ce langage doivent être très facilement portables sur plusieurs systèmes d'exploitation tels que Unix, Windows, Mac OS ou GNU/Linux, avec peu ou pas de modifications, mais qui ont l'inconvénient d'être plus lourd à l'exécution (en mémoire et en temps processeur) à cause de sa machine virtuelle. Pour cela, divers plateformes et Framework associés visent à guider, sinon garantir, cette portabilité des applications développées en Java.

### D.

C# est un langage de programmation polyvalent, moderne et orienté objet, prononcé «C Sharp ». Il a été développé par Microsoft dirigé par Anders Hejlsberg et son équipe dans le cadre de l'initiative .Net et approuvé par l'Association européenne des constructeurs d'ordinateurs

(ECMA) et l'Organisation internationale de normalisation (ISO). Il est dérivé du C++ et très proche du Java dont il reprend la syntaxe générale ainsi que les concepts, y ajoutant des notions telles que la surcharge des opérateurs, les indexeurs et les délégués. Il est utilisé notamment pour développer des applications web sur la plateforme ASP.NET.

## **E. HTML**



L'HTML est un langage informatique utilisé sur l'internet. Ce langage est utilisé pour créer des pages web. L'acronyme signifie HyperText Markup Language, ce qui signifie en français "langage de balisage d'hypertexte". Cette signification porte bien son nom puisqu'effectivement ce langage permet de réaliser de l'hypertexte à base d'une structure de balisage.

Ce n'est pas à proprement parlé un langage de programmation, mais plutôt un langage qui permet de mettre en forme du contenu. Les balises permettent de mettre en forme le texte et de placer des éléments interactifs, tel des liens, des images ou bien encore des animations.

## **F. Javascript**



JavaScript a été créé en 1995 par Brendan Eich. Il a été standardisé sous le nom d'ECMAScript en juin 1997 par Ecma International dans le standard ECMA-262. JavaScript est un langage de programmation de scripts principalement employé dans les pages web interactives mais aussi pour les serveurs avec l'utilisation (par exemple) de Node.js. C'est un langage orienté objet à prototype, c'est-à-dire que les bases du langage et ses principales interfaces sont fournies par des objets qui ne sont pas des instances de classes, mais qui sont chacun équipés de constructeurs permettant de créer leurs propriétés, et notamment une propriété de prototypage qui permet d'en créer des objets héritiers personnalisés. En outre, les fonctions sont des objets de première classe. Le langage supporte le paradigme objet, impératif et fonctionnel.

Avec les technologies HTML et CSS, JavaScript est parfois considéré comme l'une des technologies cœur du World Wide Web. Le langage JavaScript permet des pages web interactives, et à ce titre est une partie essentielle des applications web. Une grande majorité des sites web l'utilisent, et la majorité des navigateurs web disposent d'un moteur JavaScript

dédié pour l'interpréter, indépendamment des considérations de sécurité qui peuvent se poser le cas échéant.

## **G. PHP**



Le langage PHP a été inventé par Rasmus LERDORF en 1995 pour son usage personnel. Autrefois abréviation de Personal HomePage devenue aujourd'hui HyperText Préprocesseur, PHP s'impose comme un standard dans le monde de la programmation web par ses performances, sa fiabilité, sa souplesse et sa rapidité. PHP a été inventé à l'origine pour le développement d'applications web dynamiques qui constituent encore le cas d'utilisation le plus courant et son point fort. Cependant, les évolutions qui lui ont été apportées jusqu'à aujourd'hui assurent à PHP une polyvalence non négligeable. PHP est par exemple capable d'interagir avec Java, de générer des fichiers PDF, d'exécuter des commandes Shell, de gérer des objets (au sens programmation orientée objet), de créer des images ou bien de fournir des interfaces graphiques au moyen de PHP GTK.

### **❖ CHOIX DU LANGAGE**

Pour le développement de notre plateforme, nous avons choisi le langage PHP.

#### **Pourquoi PHP ?**

- ✚ La plus importante est que nous bénéficions de nombreux avantages qui se traduisent en économie, et ce, sans compromis sur la qualité. Comment ? Tout d'abord, PHP est gratuit et ne nécessite aucune licence d'utilisation.
- ✚ Ensuite, PHP est le langage de programmation Web le plus utilisé au monde. Il existe une communauté de développeurs très active qui rend disponibles des dizaines de milliers de librairies PHP de grande qualité ainsi qu'une vaste quantité de documentation et tutoriels accessible à tous au bénéfice de chacun. Ces ressources facilitent notre travail et réduisent notre temps de programmation ce qui se traduit aussi en économie pour le client.
- ✚ En termes de rapidité et d'efficacité, PHP n'a rien à envier aux autres langages. Plusieurs portails très populaires et nécessitant beaucoup de performance l'utilisent.

Nous avons qu'a pensé à Facebook, Wikipédia, le réseau CBC, l'université Harvard, pour ne nommer que ceux-là. D'ailleurs, le populaire système de gestion de contenu WordPress est lui-même construit en PHP. Ce dernier représente à lui seul environ 80% des sites Internet sur la toile. Une si grande popularité n'est certainement pas une coïncidence.

### **III. ETUDE DE FRAMEWORK**

Un Framework est un ensemble d'outils et de composants logiciels organisés conformément à un plan d'architecture et des patterns, l'ensemble formant ou promouvant un « squelette » de programme, un canevas. Il est souvent fourni sous la forme d'une bibliothèque logicielle et accompagné du plan de l'architecture cible du Framework. Un Framework est conçu en vue d'aider les programmeurs dans leur travail. L'organisation du Framework vise la productivité maximale du programmeur qui va l'utiliser.

#### **❖ CHOIX DU FRAMEWORK : LARAVEL**



Laravel, créé par Taylor Otwell, initie une nouvelle façon de concevoir un Framework en utilisant ce qui existe de mieux pour chaque fonctionnalité. Par exemple, toute application web a besoin d'un système qui gère les requêtes HTTP. Plutôt que de réinventer quelque chose le concepteur de Laravel a tout simplement utilisé celui de Symfony en l'étendant pour créer un système de routage efficace. De la même manière, l'envoi des emails se fait avec la bibliothèque Swift Mailer. En quelque sorte Otwell a fait son marché parmi toutes les bibliothèques disponibles. Nous verrons comment cela est réalisé. Mais Laravel ce n'est pas seulement le regroupement de bibliothèques existantes, c'est aussi de nombreux composants originaux et surtout une orchestration de tout ça.

#### **Pourquoi Laravel ?**

- ✦ **Développement plus rapide** : Le choix du Framework Laravel accélère votre développement d'applications Web et vous permet donc de le faire vivre beaucoup plus rapidement que si vous utilisiez d'autres Framework.

- † **Sécurité top-Notch** : L'un des avantages les plus importants du choix de Laravel pour le développement de vos applications Web réside dans ses capacités à fournir une sécurité de haut niveau. Laravel est lui-même un Framework sécurisé qui ne permet à aucune application de logiciel malveillant ni de menace à la sécurité d'entrer dans l'application Web.
- † **Performances améliorées** : Une autre raison forte de choisir Laravel est sa capacité à fournir une excellente performance des applications Web.
- † **Grande communauté et beaucoup de documentation** : Un autre avantage pour Laravel est qu'il est open-source. Laravel possède une puissante communauté de développeurs et de sociétés de développement qui fournit en permanence une assistance pour la rendre plus flexible et évolutive.

### **SECTION 3 : ETUDE COMPARATIVE DE QUELQUES SGBD**

#### **I. PRESENTATION**

Un Système de Gestion de Base de Données (SGBD) est un logiciel qui permet de stocker des informations dans une base de données. Un tel système permet de lire, écrire, modifier, trier, transformer ou même imprimer les données qui sont contenus dans la base de données.

#### **II. EXEMPLES DE QUELQUES SGBD**

##### **A. MYSQL**

MySQL est un système de gestion de bases de données relationnelles. Le SQL dans "MySQL" signifie "Structured Query Language" : le langage standard pour les traitements de bases de données. MySQL est Open Source. Open Source (Standard Ouvert) signifie qu'il est possible à chacun d'utiliser et de modifier le logiciel. Tout le monde peut le télécharger sur Internet et l'utiliser sans payer aucun droit. Toute personne en ayant la volonté peut étudier et modifier le code source pour l'adapter à ses besoins propres. Toutefois, si vous devez intégrer MySQL dans une application commerciale, vous devez vous procurer une licence auprès de MySQL AB. Ce qui rend MySQL très intéressant pour les Webmasters est le nombre d'API (application Program interface) dont il dispose. Vous pouvez en effet l'intégrer dans des applications écrites en : C, C++, Eiffel, Java, Perl, PHP, Python, Ruby et Tcl.

## B. SQL SERVER



SQL Server est un système de gestion de base de données relationnelle (SGBDR) développé par Microsoft. Il est principalement conçu et développé pour concurrencer les bases de données MySQL et Oracle.

SQL Server prend en charge ANSI SQL, qui est le langage standard SQL (Structured Query Language). Cependant, SQL Server est fourni avec sa propre implémentation du langage SQL, T-SQL (Transact-SQL).

T-SQL est un langage propriétaire Microsoft appelé Transact-SQL. Il offre des possibilités supplémentaires de déclaration de variable, de traitement des exceptions, de procédure stockée, etc.

SQL Server Management Studio (SSMS) est le principal outil d'interface de SQL Server. Il prend en charge les environnements 32 bits et 64 bits.

## C. Microsoft Access



Microsoft Access est un système de gestion de base de données (SGBD) de Microsoft qui associe le moteur de base de données Microsoft Jet relationnel à une interface utilisateur graphique et à des outils de développement logiciel. Il fait partie de la suite d'applications Microsoft Office, incluse dans les éditions Professionnelle et ultérieure, ou vendue séparément. Microsoft Access stocke les données dans son propre format basé sur le moteur de base de données Access Jet. Il peut également importer ou relier directement des données stockées dans d'autres applications et bases de données.

Les développeurs de logiciels, les architectes de données et les utilisateurs expérimentés peuvent utiliser Microsoft Access pour développer des logiciels d'application. Comme d'autres applications Microsoft Office, Access est pris en charge par Visual Basic pour Applications (VBA), langage de programmation à base d'objets pouvant référencer divers objets, notamment DAO (Data Access Objects), ActiveX Data Objects et de nombreux autres composants ActiveX. Les objets visuels utilisés dans les formulaires et les rapports exposent

leurs méthodes et propriétés dans l'environnement de programmation VBA. Les modules de code VBA peuvent également déclarer et appeler des opérations du système d'exploitation Windows.

### **III. CHOIX ET JUSTIFICATION**

Pour le développement de notre plateforme, nous avons choisi le **MySQL** comme base de données.

#### **Pourquoi MySQL ?**

Il est **multi-thread** : on peut exécuter plusieurs processus en même temps.

**Multi-utilisateur** : qui fonctionne aussi bien sur Windows que sur Linux ou Mac OS.

Accessibles en utilisant de nombreux langages serveur, dont PHP, que nous utiliserons.

C'est un logiciel **libre** : open source

## **SECTION 3 : ARCHITECTURE APPLICATIVE**

### **I. DEFINITION**

En informatique, architecture désigne la structure générale inhérente à un système informatique, l'organisation des différents éléments du système (logiciels et/ou matériels et/ou humains et/ou informations) et des relations entre les éléments. Cette structure fait suite à un ensemble de décisions stratégiques prises durant la conception de tout ou partie du système informatique, par l'exercice d'une discipline technique et industrielle du secteur de l'informatique dénommée elle aussi architecture, et dont le responsable est l'architecte informatique.

Un « niveau » peut également être appelé « couche ». Trois couches impliquées dans l'application, à savoir la couche présentation, la couche métier et la couche données. Voyons chaque couche en détail :

### **II. EXEMPLE D'ARCHITECTURE**

Les différentes architectures existantes sont :

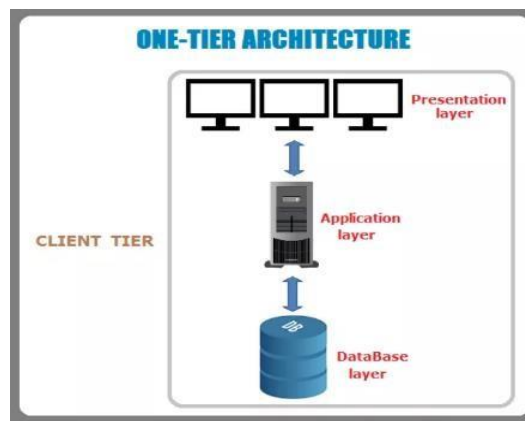
❖ Architecture 1-Tier

- ❖ Architecture 2-Tier
- ❖ Architecture 3-Tier
- ❖ Architecture N-Tier

### **A. ARCHITECTURE 1-TIERS**

Une architecture à un niveau implique de placer tous les composants requis pour une application logicielle ou une technologie sur un serveur ou une plate-forme unique.

Fondamentalement, une architecture à un niveau conserve tous les éléments d'une application, y compris l'interface, le middleware et les données backend, au même endroit. Les développeurs considèrent ces types de systèmes comme le moyen le plus simple et le plus direct.



**Figure 19 : Présentation de l'architecture 1-tier**

### **B. ARCHITECTURE 2-TIERS**

L'architecture à deux niveaux est divisée en deux parties :

- Application client (niveau client)
- Base de données (niveau données)

Le système client gère les couches de présentation et d'application et le système de serveur gère la couche de base de données. Il est également appelé application client-serveur. La communication a lieu entre le client et le serveur. Le système client envoie la demande au système serveur et le système serveur traite la demande et renvoie les données au système client.

### **Les avantages**

- Elle permet l'utilisation d'une interface utilisateur riche.
- Elle a permis l'appropriation des applications par l'utilisateur.
- Elle a introduit la notion d'interopérabilité.

### **Les inconvénients**

- On ne peut pas soulager la charge du poste client, qui supporte la grande majorité des traitements applicatifs.
- Le poste client est fortement sollicité, il devient de plus en plus complexe et doit être mis à jour régulièrement pour répondre aux besoins des utilisateurs.
- Les applications se prêtent assez mal aux fortes montées en charge car il est difficile de modifier l'architecture initiale.
- La relation étroite qui existe entre le programme client et l'organisation de la partie serveur complique les évolutions de cette dernière.
- Ce type d'architecture est grandement rigidifié par les coûts et la complexité de sa maintenance.

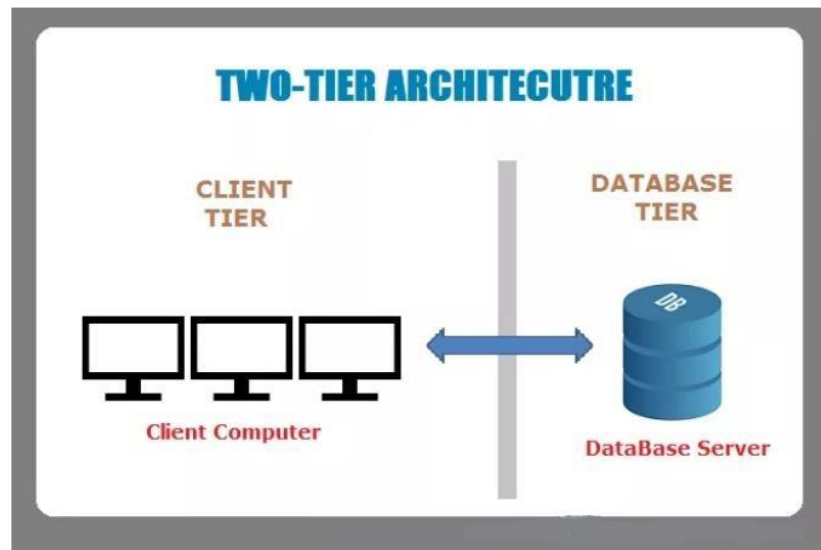


Figure 20: Présentation de l'architecture

### C. ARCHITECTURE 3-TIERS

L'architecture à trois niveaux est divisée en trois parties :

- Couche présentation (niveau client)
- Couche application (niveau métier)
- Couche base de données (niveau de données)

Le système client gère la couche Présentation, le serveur d'application gère la couche d'application et le système serveur gère la couche de base de données.

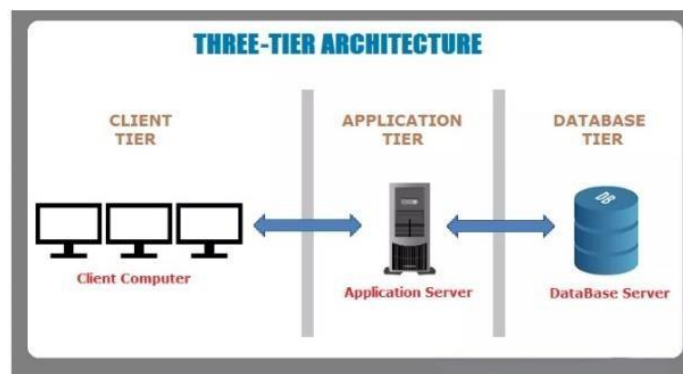
#### Les avantages

- Les requêtes clients vers le serveur sont d'une plus grande flexibilité que dans celles de l'architecture 2-tiers basées sur le langage SQL.
- Cette flexibilité permet à une entreprise d'envisager dans le cadre d'une architecture 3tiers une grande souplesse pour l'introduction de toutes nouvelles technologies.

- D'un point de vue développement, la séparation qui existe entre le client, le serveur et le SGBD permet une spécialisation des développeurs sur chaque tiers de l'architecture.
- Plus de flexibilité dans l'allocation des ressources ; la portabilité du tiers serveur permet d'envisager une allocation et ou modification dynamique au gré des besoins évolutifs au sein d'une entreprise.

### **Les inconvénients**

- Une expertise de développement à acquérir qui semble plus longue que dans le cadre d'une architecture 2-tiers.
- Les coûts de développements d'une architecture 3-tiers sont plus élevés que pour du 2tiers, au début semble-t-il, d'après une étude du cabinet Gartner.



**Figure 21 : Présentation de l'architecture 3-tier**

### **C. ARCHITECTURE N-TIERS**

L'architecture à plusieurs niveaux s'appelle également architecture à plusieurs niveaux car le logiciel est conçu pour que les fonctions de traitement, de gestion des données et de présentation soient séparées physiquement et logiquement. Cela signifie que ces différentes fonctions sont hébergées sur plusieurs machines ou clusters, garantissant ainsi que les services sont fournis sans partage de ressources et que, par conséquent, ces services sont fournis à pleine capacité.

Le « N » dans l'architecture n-tiers nommée fait référence à n'importe quel nombre à partir de 1.

### **Les avantages**

- ❖ **Sécurisé** : vous pouvez sécuriser chacun des trois niveaux séparément en utilisant différentes méthodes.
- ❖ **Facile à gérer** : vous pouvez gérer chaque niveau séparément, en ajoutant ou en modifiant chaque niveau sans affecter les autres niveaux.
- ❖ **Évolutif** : si vous devez ajouter plus de ressources, vous pouvez le faire par niveau, sans affecter les autres niveaux.
- ❖ **Flexible** : outre l'évolutivité isolée, vous pouvez également étendre chaque niveau de la manière dictée par vos besoins.

## La valeur ajoutée des architectures n-tiers

Cette séparation par couches de responsabilités sert à découpler au maximum une couche de l'autre afin d'éviter l'impact d'évolutions futures de l'application.

Par exemple : si l'on est amené à devoir changer de base de données relationnelle, seule la couche d'accès aux données sera impactée, la couche de service et la couche de présentation ne seront pas concernées car elles auront été découplées des autres.

En bref, avec une architecture à plusieurs niveaux, vous pouvez adopter de nouvelles technologies et ajouter plus de composants sans avoir à réécrire l'application dans son intégralité ni à modifier la conception de votre logiciel, facilitant ainsi sa mise à l'échelle ou sa maintenance. Dans le même temps, en termes de sécurité, vous pouvez stocker des informations sensibles ou confidentielles dans le niveau logique, en les éloignant du niveau présentation et en les rendant ainsi plus sécurisées.

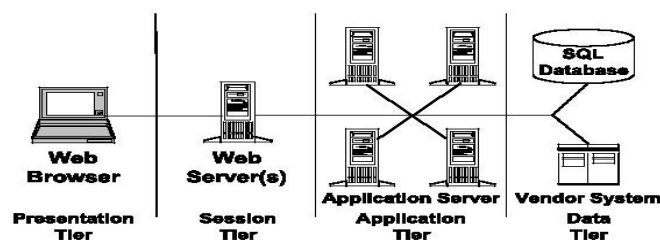


Figure 22 : Présentation de l'architecture N-tiers

## ❖ CHOIX ET JUSTIFICATION

Notre choix porte sur l'architecture **3-Tiers**.

### Pourquoi ?

L'objectif premier d'un système d'information quel qu'il soit est de permettre à plusieurs utilisateurs d'accéder aux mêmes informations. Pour cela, il faut donc regrouper les informations utilisées. En terme technique, cela se traduit par la centralisation des données au sein d'une base de données. L'évolution des systèmes d'information s'est donc basée sur une

meilleure subdivision entre les tâches à réaliser pour permettre l'exploitation de ces données par les utilisateurs finaux. Ceci permet de structurer plus efficacement les informations ce qui entraîne à la fois une meilleure organisation et une meilleure efficacité technique. Cette subdivision a été facilitée par l'avènement des technologies orientées objets qui s'appliquent aussi bien au modèle client-serveur qu'au modèle Internet. Ces technologies permettent une séparation entre les différents composants du système. Il devient alors possible de réaliser de nouvelles architectures permettant la mise à disposition des informations sous différentes formes tout en diminuant les temps de développement. Ces technologies permettent également de faire collaborer une grande diversité de systèmes. On parle alors d'architecture distribuée. Il est ainsi possible de présenter des données en provenance d'un mainframe mélangées à des données en provenance d'un SGBDR, le tout étant affiché dans un browser sur la même page HTML.

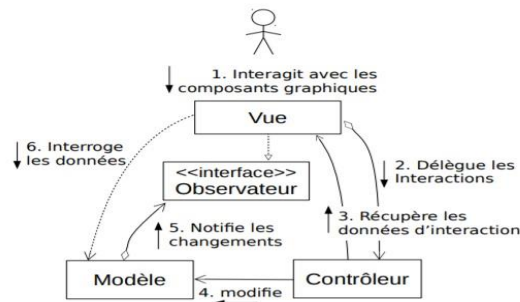
### SECTION 4 : MVC (Modèle-Vue-Contrôleur)

Le patron MVC est le premier à avoir été défini en 1979 par Trygve Reenskaug pour les applications SmallTalk. Dans ce patron, la répartition des responsabilités est faite entre trois parties.

- Le **modèle** contient la logique métier et l'état courant de l'interface durant le cycle dialogue avec l'utilisateur. Il peut être aussi simple qu'une valeur entière ou aussi complexe qu'un service nécessitant plusieurs niveaux de traitement avec persistance des données.
- La **vue** porte toute la logique de présentation. Son travail consiste à afficher les données du domaine et à recevoir des interactions de l'utilisateur. La partie vue peut évidemment être composée de plusieurs « fenêtres » graphiques différentes. La vue ne gère pas les interactions avec les utilisateurs mais les délègue à un autre composant : le contrôleur.
- Le **contrôleur** reçoit les interactions de l'utilisateur de la vue et les traite en modifiant le modèle. Le code à l'intérieur du contrôleur correspond essentiellement à du code de liaison (glue code) entre la vue et le modèle.

Une fois le modèle modifié, il est nécessaire d'afficher les données mises à jour vers l'utilisateur. Pour cela, MVC utilise le patron de conception Observateur qui agit entre la vue et le modèle.

La vue s'inscrit auprès du modèle, et reçoit une notification à chaque modification du modèle. Notons que même si le modèle interagit avec la vue, il n'a aucune idée de son existence grâce à l'interface qui joue le rôle de pare-feu.



**Figure 23 : Diagramme de communication du patron MVC**

Ce patron fonctionne donc en cycle. L'utilisateur interagit avec les composants graphiques des vues à sa disposition ce qui se déclenche la création d'événements qui sont envoyés au contrôleur qui leur est associé. Concrètement, ce ne sont pas directement les événements qui sont envoyés au contrôleur, mais cela passe par l'appel de méthodes spécifiques du contrôleur de manière à circonscrire le code graphique à la vue. En particulier, il ne doit pas y avoir d'import de membres de la bibliothèque graphique utilisée. Le contrôleur vérifie la conformité des interactions et déduit les modifications à apporter au modèle qui les intègre selon ses règles métier. Les modifications du modèle sont ensuite signalées à toutes les vues qui se mettent à jour en conséquence.

## CHAPITRE II : DEPLOIEMENT

### SECTION 1 : PRESENTATION DE LA PLATEFORME

#### ✓ L'interface d'authentification

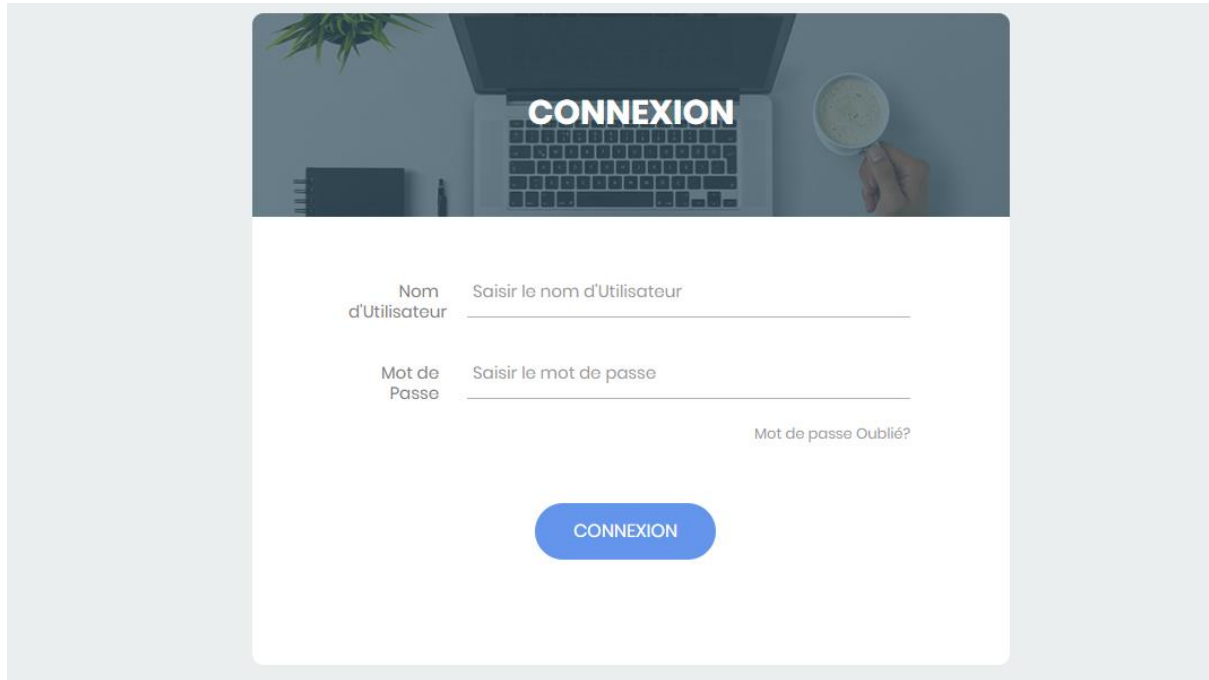


Figure 24 : Authentification

#### ✚ Interface Ajout Utilisateur

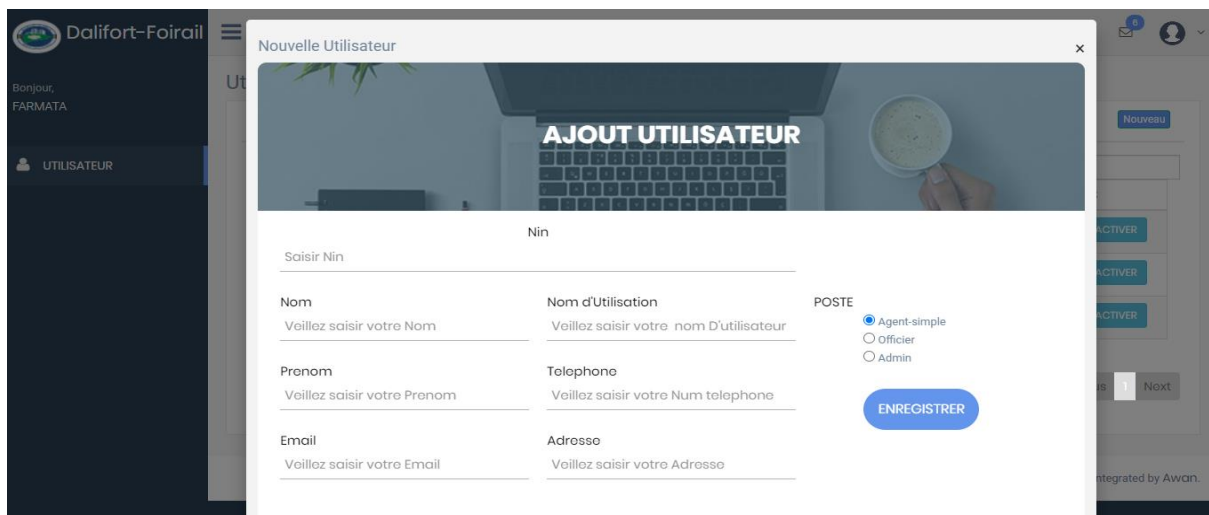


Figure 25 : Ajout Utilisateur

# MISE EN PLACE D'UNE APPLICATION DE GESTION DES ACTES D'ETAT CIVIL

## ✚ Changer mot de passe

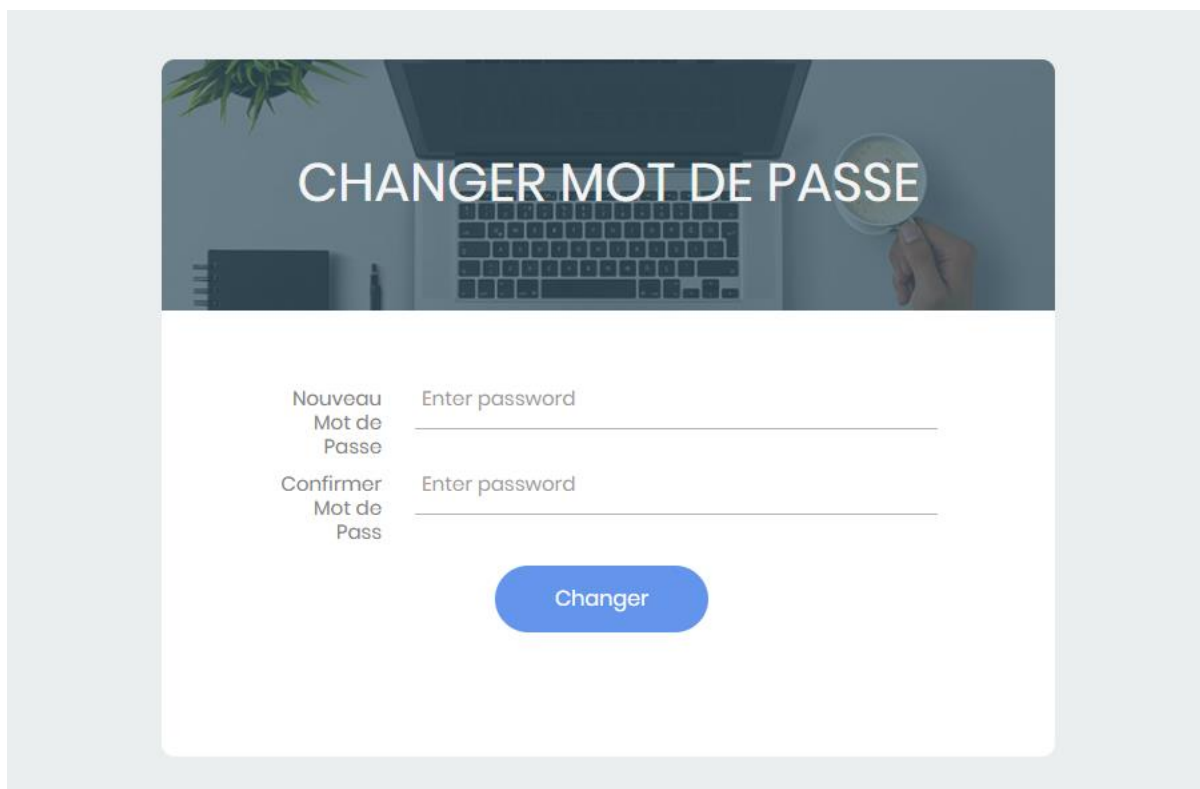


Figure 26 : Changer Mot de Passe

## ✚ Interface d'accueil-officier :

La page d'accueil contient un menu principal composé des tâches suivantes :

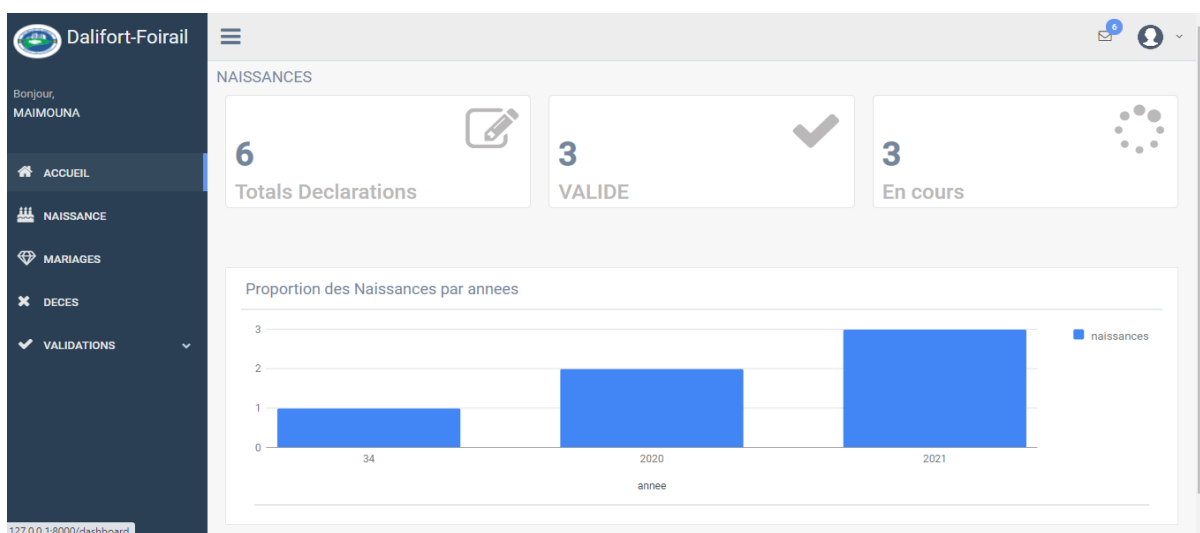


Figure 27 : Accueil Officier

# MISE EN PLACE D'UNE APPLICATION DE GESTION DES ACTES D'ETAT CIVIL

## ✚ Interface d'accueil-Agent :

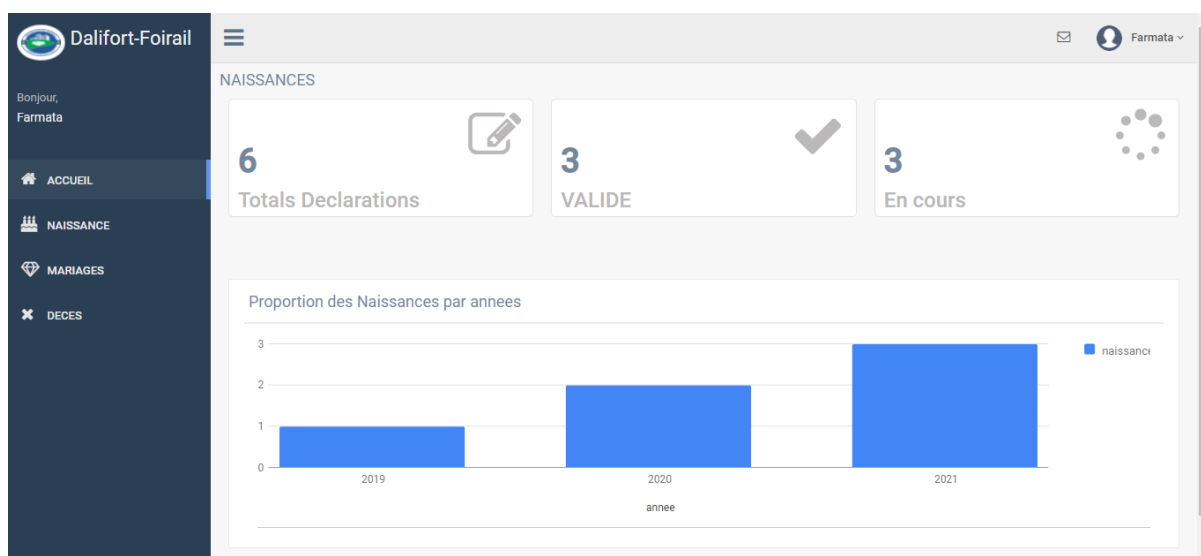


Figure 28 : Accueil Agent

## ✚ Interface d'enregistrement de naissance :

Figure 29 : Enregistrer Naissance

## **SECTION 2 : GESTION DE LA SECURITE**

### **I. DEFINITION ET ROLE**

La sécurité des systèmes d'information (SSI) ou plus simplement sécurité informatique, est l'ensemble des moyens techniques, organisationnels, juridiques et humains nécessaires à la mise en place de moyens visant à empêcher l'utilisation non-autorisée, le mauvais usage, la modification ou le détournement du système d'information. Assurer la sécurité du système d'information est une activité du management du système d'information.

Aujourd'hui, la sécurité est un enjeu majeur pour les entreprises ainsi que pour l'ensemble des acteurs qui l'entourent. Elle n'est plus confinée uniquement au rôle de l'informaticien. Sa finalité sur le long terme est de maintenir la confiance des utilisateurs et des clients. La finalité sur le moyen terme est la cohérence de l'ensemble du système d'information. Sur le court terme, l'objectif est que chacun ait accès aux informations dont il a besoin. La norme traitant des SMSI est l'ISO/CEI 27001 qui insiste sur Disponibilité – Intégrité - Confidentialité.

#### **Pourquoi ?**

Les conséquences d'une mauvaise sécurisation peuvent concerner les organisations, mais aussi la vie privée d'une ou plusieurs personnes, notamment par la diffusion d'informations confidentielles comme leurs coordonnées bancaires, leurs situations patrimoniales, leurs codes confidentiels, etc. De manière générale, la préservation des données relatives aux personnes fait l'objet d'obligations légales régies par la Loi Informatique et Libertés.

Aujourd'hui, il est généralement admis que la sécurité ne peut être garantie à 100 % et requiert donc le plus souvent la mobilisation d'une panoplie de mesures pour réduire les chances de pénétration des systèmes d'information.

#### **Objectif ?**

La sécurité des systèmes d'information vise les objectifs suivants (C.A.I.D.) :

- ✓ **Confidentialité** : seules les personnes autorisées peuvent avoir accès aux informations qui leur sont destinées (notions de droits ou permissions). Tout accès indésirable doit être empêché.

## MISE EN PLACE D'UNE APPLICATION DE GESTION DES ACTES D'ETAT CIVIL

- ✓ **Authentification** : les utilisateurs doivent prouver leur identité par l'usage de code d'accès. Il ne faut pas mélanger identification et authentification. Dans le premier cas, l'utilisateur n'est reconnu que par son identifiant, tandis que dans le deuxième cas, il doit fournir un mot de passe ou un élément que lui-seul connaît. Cela permet de gérer les droits d'accès aux ressources concernées et maintenir la confiance dans les relations d'échange.
- ✓ **Intégrité** : les données doivent être celles que l'on attend, et ne doivent pas être altérées de façon fortuite, illicite ou malveillante. En clair, les éléments considérés doivent être exacts et complets. Cet objectif utilise généralement des méthodes de calculs de checksum ou de hachage.
- ✓ **Disponibilité** : l'accès aux ressources du système d'information doit être permanent et sans faille durant les plages d'utilisation prévues. Les services et ressources sont accessibles rapidement et régulièrement.
- ✓ D'autres aspects peuvent aussi être considérés comme des objectifs de la sécurité des systèmes d'information, tels que :
- ✓ **La traçabilité (ou « preuve »)** : garantie que les accès et tentatives d'accès aux éléments considérés sont tracés et que ces traces sont conservées et exploitables.
- ✓ **La non-répudiation et l'imputation** : aucun utilisateur ne doit pouvoir contester les opérations qu'il a réalisées dans le cadre de ses actions autorisées et aucun tiers ne doit pouvoir s'attribuer les actions d'un autre utilisateur.

Une fois les objectifs de la sécurisation déterminés, les risques pesant sur chacun de ces éléments peuvent être estimés en fonction des menaces. Le niveau global de sécurité des systèmes d'information est défini par le niveau de sécurité du maillon le plus faible. Les précautions et contre-mesures doivent être envisagées en fonction des vulnérabilités propres au contexte auquel le système d'information est censé apporter service et appui.

Il faut pour cela estimer :

- ❖ La **gravité** des conséquences au cas où les risques se réaliseraient ;



## MISE EN PLACE D'UNE APPLICATION DE GESTION DES ACTES D'ETAT CIVIL



- ❖ La **vraisemblance** des risques (ou leur potentialité, ou encore leur probabilité d'occurrence).

Les précautions et contre-mesures doivent être envisagées en fonction des vulnérabilités propres au contexte auquel le système d'information est censé apporter service et appui. C'est ainsi que les mesures suivantes seront adoptées pour ce système :

- ❖ La gestion des rôles et des accès
- ❖ Un algorithme de cryptage pour les mots de passe
- ❖ Une politique de sauvegarde des données

# MISE EN PLACE D'UNE APPLICATION DE GESTION DES ACTES D'ETAT CIVIL

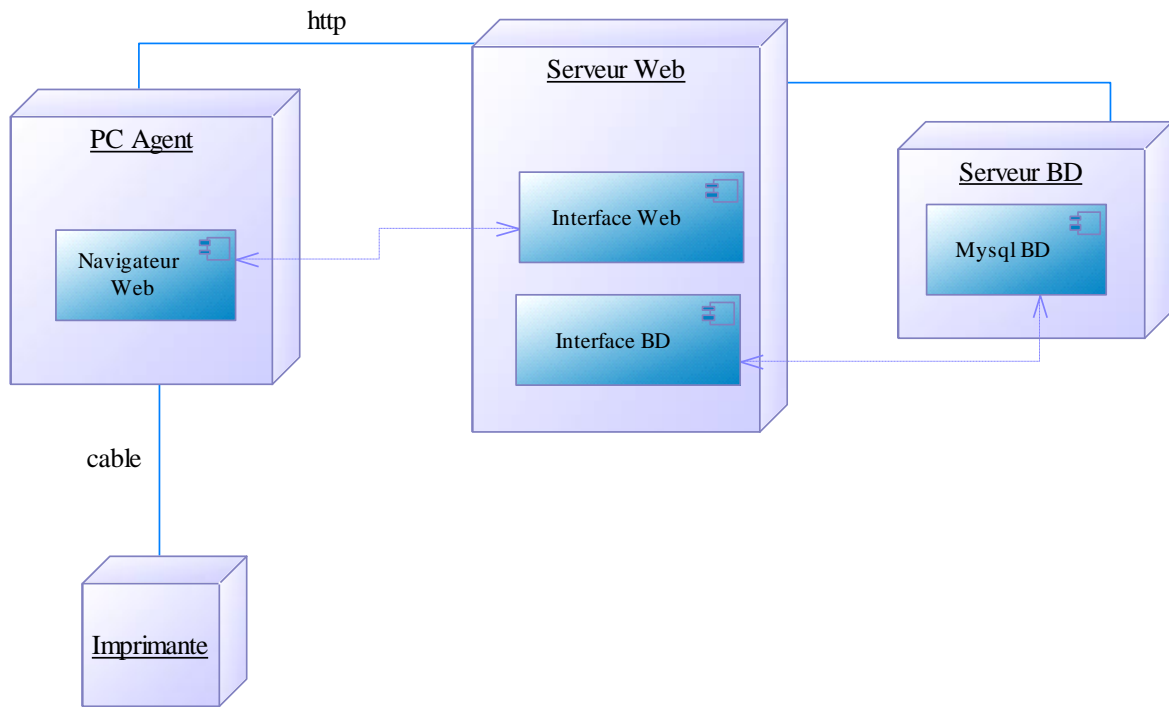
## SECTION 3 : ESTIMATION COUT

Pour le déploiement de cette plateforme, nous aurons besoin d'un ensemble de matériaux à savoir :

**Tableau V : tableau financier**

Besoin	Article	Quantité	Prix Unitaire	Total
Matériel Informatique	Ordinateur HP	3	300.000	900.000
	Imprimant Réseau	1	200.000	200.000
Matériel Electrique	Multiprise	3	5000	15.000
	Régulateur de tension	1	40.000	40.000
Hébergement	Hébergement Serveur	1	60.000	60.000
	Hébergement Application web	1	60.000	60.000
Main d'œuvre	Conception et Déploiement	1	0	0
<b>TOTAL</b>				<b>1.335.000</b>

**SECTION 3 : DIAGRAMME DE DEPLOIEMENT**



**Figure 30 : Diagramme de Déploiement**

# CONCLUSION

## **MISE EN PLACE D'UNE APPLICATION DE GESTION DES ACTES D'ETAT CIVIL**

Au terme de ce rapport, nous pouvons conclure que ce mémoire de fin d'études nous a donné une occasion opportune nous permettant de confronter l'acquis théorique à l'environnement pratique.

En effet, le mémoire nous a permis de prendre certaines responsabilités, par la suite de consolider de plus en plus nos connaissances théoriques et pratiques. C'est là que réside la valeur d'un tel projet de fin d'études qui combine les exigences de la vie professionnelle aux côtés bénéfiques de l'enseignement pratique que nous avons eu à l'UAHB.

Ce travail de conception et de développement d'une plateforme web nous a été aussi bénéfique sur plusieurs plans : il nous a permis de perfectionner nos connaissances acquises en programmation et en conception.

Bien qu'ayant rencontré de nombreuses difficultés, ce projet nous a permis de nous surpasser et aussi d'apprendre à retrouver des solutions ici et là sur internet, des notions que nous n'avons pas reçues forcément en classe, favorisant ainsi l'accroissement de nos connaissances et attisant notre désir d'apprentissage et de perfectionnement.

Au terme de cette étude, nous affirmons l'hypothèse en disant que la gestion numérique est une solution optimale pour la gestion des services offerts par les centres d'états civil. A ceci s'ajoute la connaissance sans ambiguïté des personnes qui s'enregistrent dans une ville, village....

Ainsi en conclusion, il faut retenir qu'il ne s'agit pas seulement d'arriver à ses fins en termes de réalisations mais le plus important reste la valeur des notions acquises tout au long du parcours. C'est en trouvant des solutions à nos problèmes que nous avons pu mener à bien notre projet. Comme on disait « Nulle pierre ne peut être polie sans friction, nul homme ne peut parfaire son expérience sans épreuves ». Dans un futur meilleur, il sera prévu de mettre en place un site web qui nous permettra de faire des demandes en ligne. Enfin nous espérons à travers ce mémoire, avoir permis à des lecteurs d'appréhender l'état nécessaire au développement d'une application web.

## **BIBLIOGRAPHIE**

Code de la famille

RADI (Réseau Africain pour le développement Intégré), Livret d'informations juridiques : L'état Civil, Editeur: Mandarin-Editions, 2009(26 pages).

## **WEBOGRAPHIE**

<https://www.bestcours.com/conception-et-analyse/merise/187-formation-merise-pdf.html>.

<https://www.institut-numerique.org/chapitre-ii-cadre-methodologique-de-letude-50d5c18b14b97>.

[https://fr.wikipedia.org/wiki/C\\_\(langage\)](https://fr.wikipedia.org/wiki/C_(langage)).

<https://openclassrooms.com/fr/courses/1894236-programmez-avec-le-langage-c>.

[https://fr.wikipedia.org/wiki/Java\\_\(langage\)](https://fr.wikipedia.org/wiki/Java_(langage)).

<https://www.geeksforgeeks.org/introduction-to-c-sharp/>.

<https://apprendre-php.com/tutoriels/tutoriel-19-introduction-au-langage-php.html>.

<https://www.geeksforgeeks.org/introduction-to-visual-studio/>.

<https://www.lucidchart.com/pages/fr/tutoriel-sur-les-diagrammes-de-deploiement>.

<http://docplayer.fr/1637318-Introduction-3-iv-comparaison-merise-uml-scrum-14-1approche-fonctionnelle-14-2-schema-entit%C3%A9-association-14-3-methodologie.html>.

<http://projet.eu.org/pedago/sin/term/3-UML.pdf>.



**MISE EN PLACE D'UNE APPLICATION DE GESTION  
DES ACTES D'ETAT CIVIL**