



Ecole Nationale Supérieure d'Informatique
et d'Analyse des Systèmes

ROYAUME DU MAROC



MINISTÈRE DE LA FONCTION PUBLIQUE
ET DE LA MODERNISATION DE L'ADMINISTRATION

Application gestion des contacts de MFPMA

Mémoire de stage de fin de 1^{ère} année

Réalisé par : *Mr. keddouch larbi*

Encadré par : *Mr.El Mehdi Sandadi*

Année Universitaire 2016-2017

Remerciement

Tout d'abords, je souhaite remercier le Ministère de la Fonction Publique et de Modernisation de l'Administration pour m'avoir ouvert ses portes et donner l'opportunité de réaliser un stage au Maroc.

Ensuite, je souhaite remercier vivement toutes les personnes qui m'apportent leurs aides tout au long de ce stage à savoir toutes les personnes travaillant aux services de la Division des systèmes d'information internes.

Grâce à leur disponibilité et leur gentillesse à mon égard, je bénéficie de tous les éléments nécessaires au bon déroulement de mon stage.

Je tiens compte à remercier particulièrement à rendre hommage à mon encadrant :

_ Mr El Mehdi Sandadi

Sommaire

<u>Remerciements</u>	1
<u>Liste des figures</u>	4
<u>Introduction</u>	5
<u>Dossier d'initialisation</u>	6
<u>Introduction</u>	6
<u>Status et missions</u>	6
<u>Les directions</u>	6
<u>Organigramme globale du ministère</u>	7
<u>Description du projet</u>	7
<u>Objectifs</u>	8
<u>Technologies et Outils utilisés</u>	8
<u>Méthodologie de travail : la méthode 2TUP</u>	8
<u>Conduite du projet</u>	9
<u>Planification prévisionnelle :Diagramme de GANTT</u>	9
<u>Listes des livrables</u>	10
<u>Conclusion</u>	10
....10	
<u>Dossier d'analyse des besoins</u>	12
<u>Introduction</u>	12
<u>La modélisation et expression des besoins</u>	13
<u>Diagramme technologique</u>	14
<u>Diagramme de cas d'utilisation</u>	14
<u>Administration du système</u>	14
<u>Diagramme globale de la gestion de stock</u>	15
<u>Diagramme détaillé de la gestion de stock</u>	16
<u>Le processus métier</u>	18
<u>Diagramme de séquence</u>	19

Liste des figures

<u>Figure 1 :Organigramme globale du ministère</u>	7
<u>Figure 2 :La méthode 2TUP</u>	9
<u>Figure 3 :Planification prévisionnelle</u>	10
<u>Figure 4 :Diagramme de GANT</u>	11
<u>Figure 5 :Diagramme administration du système</u>	14
<u>Figure 6 : Diagramme globale de la gestion du stock</u>	15
<u>Figure 7 : Diagramme détaillé de la gestion du stock</u>	17
<u>Figure 8 :Le processus métier</u>	18
<u>Figure 9 : Diagramme de séquence</u>	19
<u>Figure 10 : Diagramme de classe</u>	22
<u>Figure 11 : Modèle conceptuel de données</u>	23
<u>Figure 12 : Modèle physique de données</u>	24
<u>Figure 13 : Fenêtre Authentification</u>	28
<u>Figure 14 : Menu de l'application</u>	29
<u>Figure 15 : Fenêtre Responsable des achats</u>	29
<u>Figure 16 : Fenêtre Responsable du stock</u>	30
<u>Figure 17 : Fenêtre Gestion des utilisateurs</u>	31
<u>Figure 18 : Fenêtre Chart</u>	31
<u>Figure 19 : Fenêtre Report</u>	32

Introduction général

Actuellement, les entreprises ont dépassé les méthodes classiques qu'elles utilisent souvent pour la gestion de ses activités, tout en essayant de répondre à des exigences de gestion, comme :

- La grande quantité et la complexité des informations qui doivent être traitées dans un temps de plus en plus court.
- Le besoin de l'amélioration de la qualité de gestion des informations pour qu'elles soient disponibles à tout instant.

Par conséquent, seule une application informatique performante permet de répondre à ces exigences.

Dans cette perspective, se présente l'application de notre projet qui permet de gérer les contacts interne et externe de Ministère de la Fonction Publique et de Modernisation de l'Administration.

Chapitre 1

Présentation du projet

Ce chapitre présentera une description générale de notre sujet « développement d'une application de gestion des contacts de MFPMA » puis présentera notre objectif et notre démarche.

I. Présentation générale du ministère

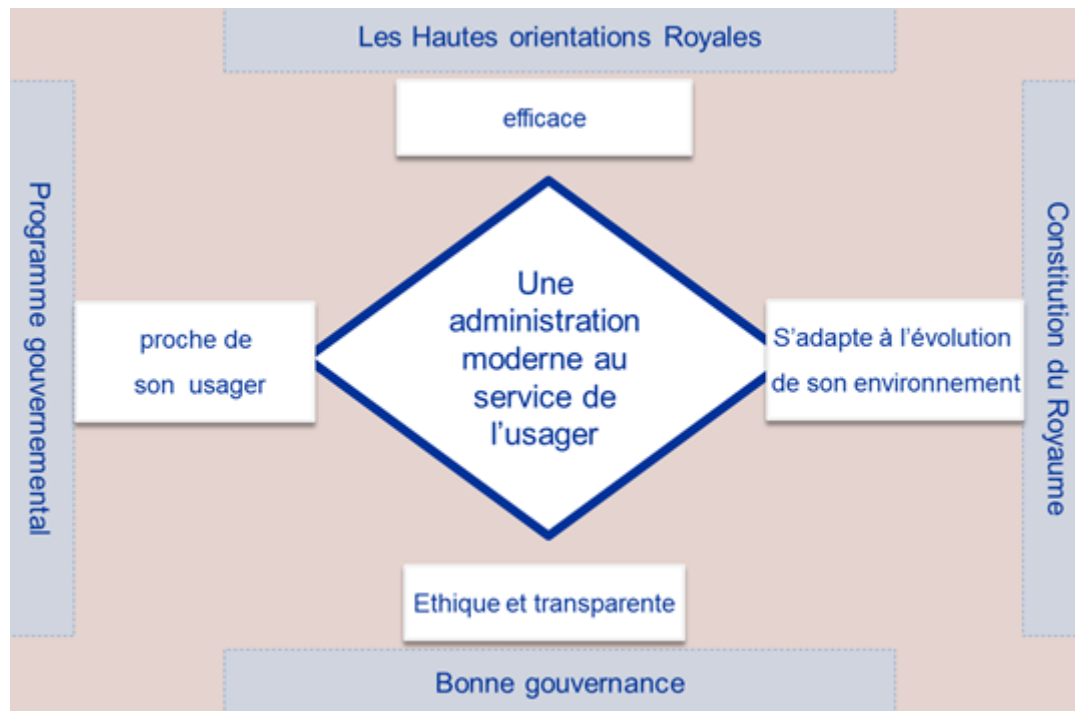
Le Ministère de la Fonction Publique et de Modernisation de l'Administration comme tout autre organisme dispose des contacts internes à savoir les fonctionnaires et externes à savoir les fournisseurs, clients et citoyens.

Quand ce dernier est important, l'utilisation d'un logiciel de gestion des contacts sous forme d'annuaire est hautement recommandée.

En effet, grâce aux différents logiciels de gestion des contacts disponibles, les entreprises peuvent:

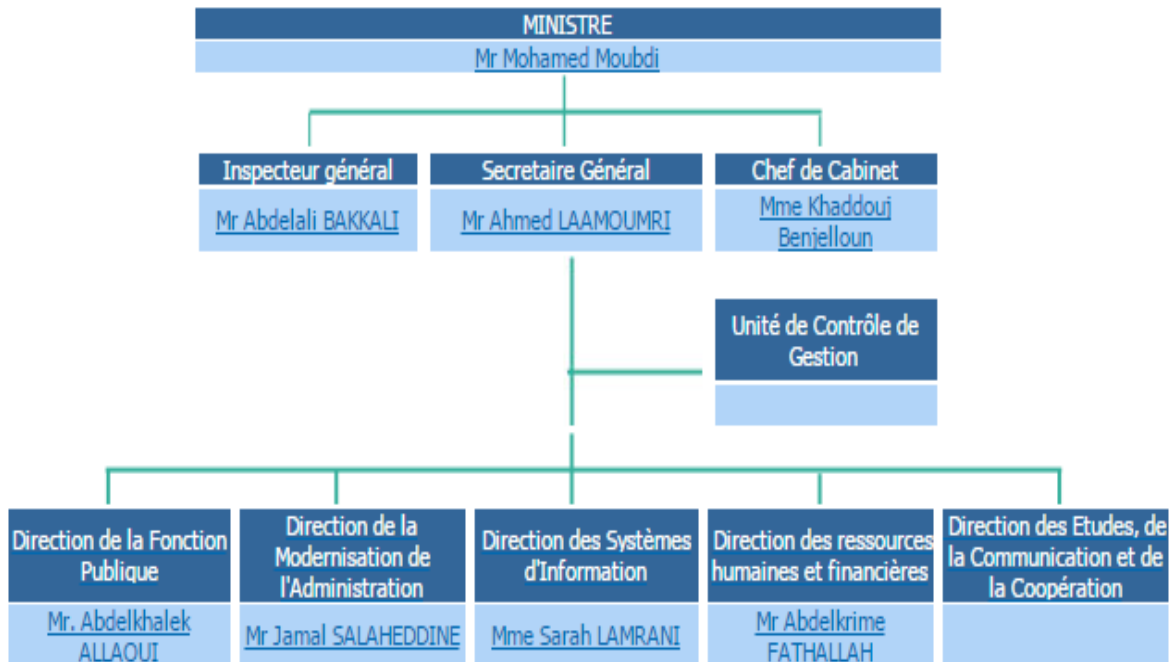
- Gérer informatiquement leurs contacts et plus particulièrement les fonctionnaires et les fournisseurs.
- Informer sur les coordonnées physiques et électroniques des fonctionnaires et certaines services et institutions publics.
- Connaître des informations sur les fonctionnaires en temps réel et obtenir leur historique et leurs orientations.
- Gérer par les services du ministère de la modernisation des secteurs publics.

1- Statuts et missions



- Proposer les mesures et les actions visant à améliorer la gestion des ressources humaines des administrations publiques et à les valoriser;
- Promouvoir la transparence dans la gestion publique ;
- Participer au développement de l'administration numérique ;
- Constituer et gérer les bases de données relatives aux personnels de l'Etat ;
- Contribuer à l'amélioration de la qualité des services publics rendus aux usagers ;
- Réaliser toutes études, enquêtes, et statistiques relatives aux personnels de l'Etat ;
- Concourir à la préparation et à la mise en œuvre de la politique gouvernementale en matière de déconcentration administrative.

2- Organismes et Directions



3- Méthodologie



- **L'appui aux administrations publiques en matière de gestion des ressources humaines et de leurs relations avec l'utilisateur ;**
- **L'ancrage des principes de corrélation entre responsabilité et reddition des comptes ;**
- **Le renforcement de la démocratie participative et de la dimension territoriale du chantier de la modernisation administrative. Et ce, à travers le dialogue et la concertation avec les différents départements ministériels, les acteurs de la société civile et les partenaires sociaux ;**
- **La coopération et la capitalisation sur les expériences réussies aux niveaux national et international ;**
- **La promotion de la gestion par objectifs axée sur les résultats: contractualisation – suivi – évaluation.**

II. Description de projet

1. Annuaire de l'administration

Cet espace est dédiée pour orienter les citoyens vers les services publics à travers la consultation dans l'annuaire de l'Administration qui est composé de :

a- Ministères et institutions

L'administration marocaine est placée sous l'autorité du Gouvernement (Chef du gouvernement, ministres et secrétaires d'Etat). Chaque ministère est composé de plusieurs grandes directions qui se subdivisent en de nombreux divisions et services. Les coordonnées de ces ministères sont constituées des éléments ci-après :

- Adresses physiques ;
 - Adresses postales ;
 - Sites internet et adresses électroniques ;
 - Numéro de téléphone et de fax ;
 - Noms des différents responsables (ministre, secrétaires généraux, directeurs centraux, etc.).
- Vous pouvez accéder à ces coordonnées :
- Soit en choisissant un département (utilisez la liste déroulante).
 - Soit en utilisant le moteur de recherche (tapez le nom de la structure ou le nom des responsables) ou le formulaire de recherche guidée (tapez le nom et prénom du responsable ou la fonction ou le nom de la structure ou la région ou la ville).

b- Services locaux

Vous pouvez accéder aux coordonnées des services locaux (adresses des guichets d'accueil, adresses postales, sites web et adresses électroniques, numéros de téléphone et de fax, noms des responsables) en cliquant sur le formulaire de recherche soit par région ou par ville.

2. Objectifs

- Développer un annuaire qui englobe l'ensemble des contacts interne et externe du MFPMA.
- Automatiser les mises à jour de l'annuaire.
- Généraliser l'accès à l'annuaire.

Le travail consiste à réaliser une application de gestion des contacts. Cette application va être développée à l'aide de « ASP.NET MVC », ce qui permettra sa portabilité et qui va faciliter l'accès aux données sur toutes les plateformes.

Chapitre 2

Analyse et Conception

Dans ce chapitre nous avons présenté la partie conception et modélisation de notre application. En particulier, nous allons rappeler dans un premier temps la méthode de conception utilisée à savoir Merise. Et dans un deuxième temps, nous présenterons les modèles de conception de notre application.

I. *La modélisation et expression des besoins*

Aujourd'hui, l'**Unified Modeling Language (UML)** constitue le standard industriel de modélisation objet. UML est une notation standardisée qui facilite la conception des logiciels, en amont du travail de programmation proprement dit ("codage"). UML se définit comme un langage de modélisation graphique et textuel destiné à comprendre et décrire des besoins, spécifier et documenter des systèmes, esquisser des architectures logicielles, concevoir des solutions et communiquer des points de vue. Il unifie à la fois les notations et les concepts orientés objet.

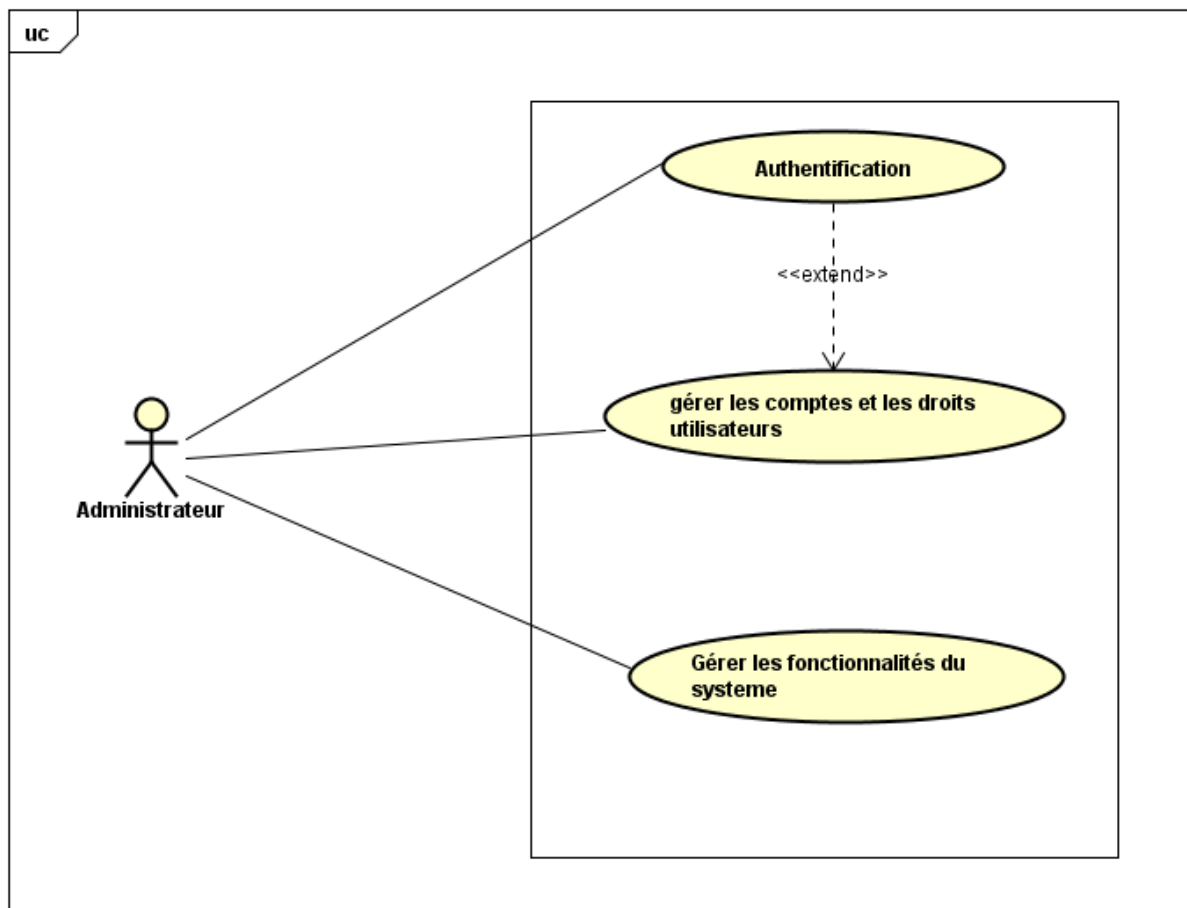
Ils permettent de visualiser et de manipuler les éléments de la modélisation. Les diagrammes définis par UML sont les suivants :

- **Les diagrammes d'activités** : représentation du comportement d'une opération en terme d'action ;
- **Les diagrammes des cas d'utilisation** : représentation des fonctions du système du point de vue de l'utilisateur ;
- **Les diagrammes de classes** : représentation de la structure statique en terme de classes et de relations ;
- **Les diagrammes de déploiement** : représentation de déploiement des composants et les dispositifs matériels ;
- **Les diagrammes d'états-transitions** : représentation du comportement d'une classe en terme d'état ;
- **Les diagrammes d'objet** : représentation des objets et de leurs relations, correspond à un diagramme de collaboration simplifié, sans représentation des envois de message ;
- **Les diagrammes de séquence** : représentation temporelle des objets et leurs interactions.

1. Diagramme de cas d'utilisation

Un diagramme de cas d'utilisation capture le comportement d'un système, d'un sous-système, d'une classe ou d'un composant tel qu'un utilisateur extérieur le voit. Il scinde la fonctionnalité du système en unités cohérentes, les cas d'utilisation, ayant un sens pour les acteurs. Les cas d'utilisation permettent d'exprimer le besoin des utilisateurs d'un système, ils sont donc une vision orientée utilisateur de ce besoin au contraire d'une vision informatique.

L'administration complète du système est assurée par l'administrateur. Celui-ci se charge de contrôler l'accès des utilisateurs et de gérer l'ensemble des fonctionnalités du système. Ceci est modélisé par le diagramme de cas d'utilisation suivant :

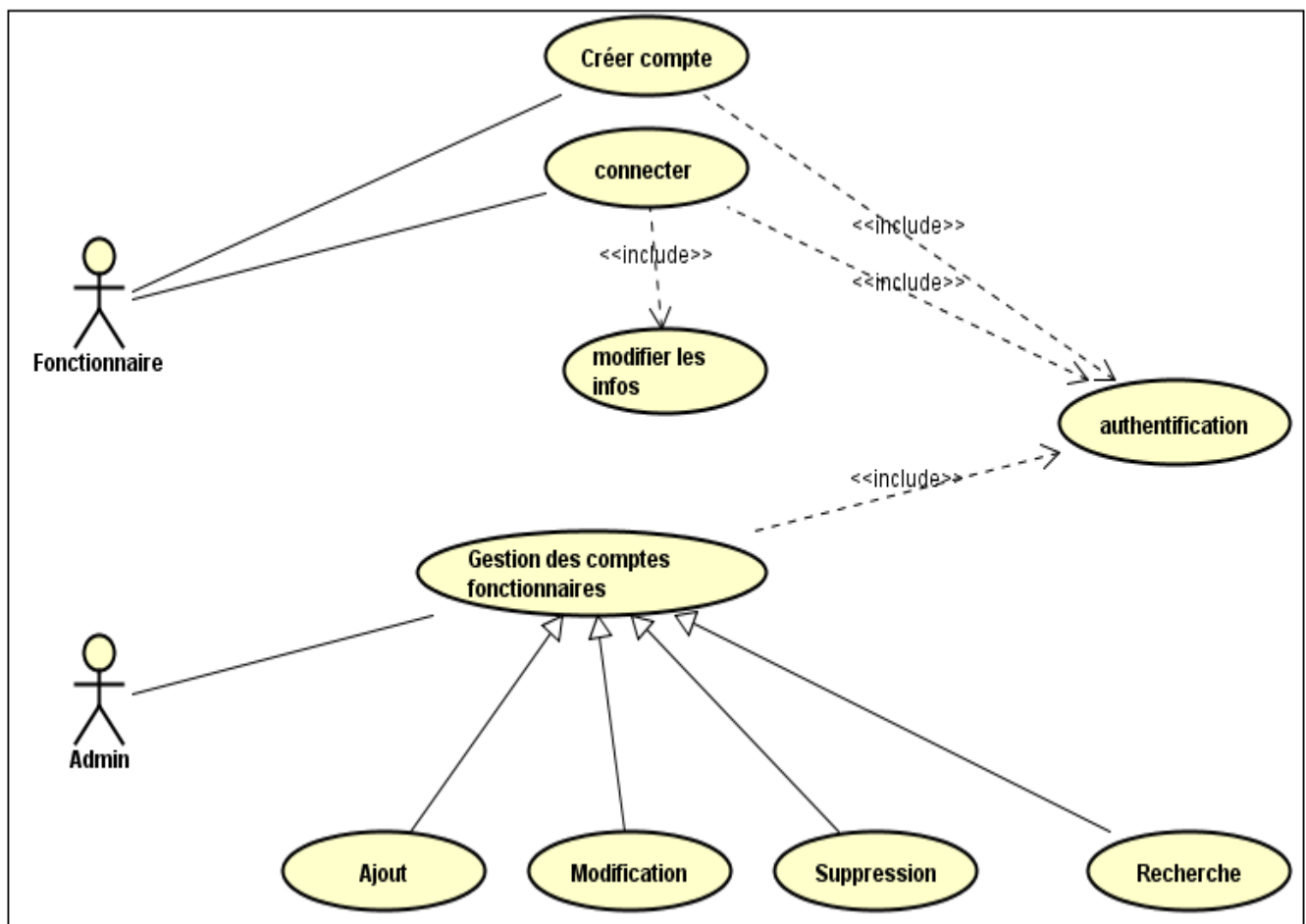


Authentification : avant de se connecter au système, chaque utilisateur doit être identifié par un adresse électronique et un mot de passe afin d'avoir les permissions d'accès au système.

Gérer les comptes et les droits des utilisateurs : ce cas permet à chaque utilisateur du système d'avoir son compte qui comporte l'ensemble de ses informations et des processus métiers dont lesquels il intervient. Comme, il attribue les droits d'accès à l'utilisateur afin de garantir la fiabilité du système.

Gérer les fonctionnalités du système : ce cas permet à l'administrateur de gérer la totalité des fonctions métiers du système, comme il peut paramétrer des nouvelles fonctionnalités en cas de besoin.

2. Diagramme globale de la gestion des contacts



3. Diagramme de séquence

Les **diagrammes de séquences** sont la représentation graphique des interactions entre les acteurs et le système selon un ordre chronologique dans la formulation Unified Modeling Language.

II. Conception de l'application

Le Dossier de Conception est l'un des étapes fondamentales qui précède l'implémentation de l'application en question et plus précisément l'analyse des besoins. Il correspond à une analyse approfondie du cahier des charges qui conduit à des choix de structure du logiciel (classes, interface graphique, persistance). Il présente :

- l'architecture fonctionnelle idéale (indépendante de la solution physique) déduite des exigences ;
- l'architecture physique composée des constituants et de leurs liens ;
- l'architecture fonctionnelle allouée sur les constituants.

Ce qui nous conduit à réaliser trois modèles de conception que nous allons détaillé au long de ce document qui sont :

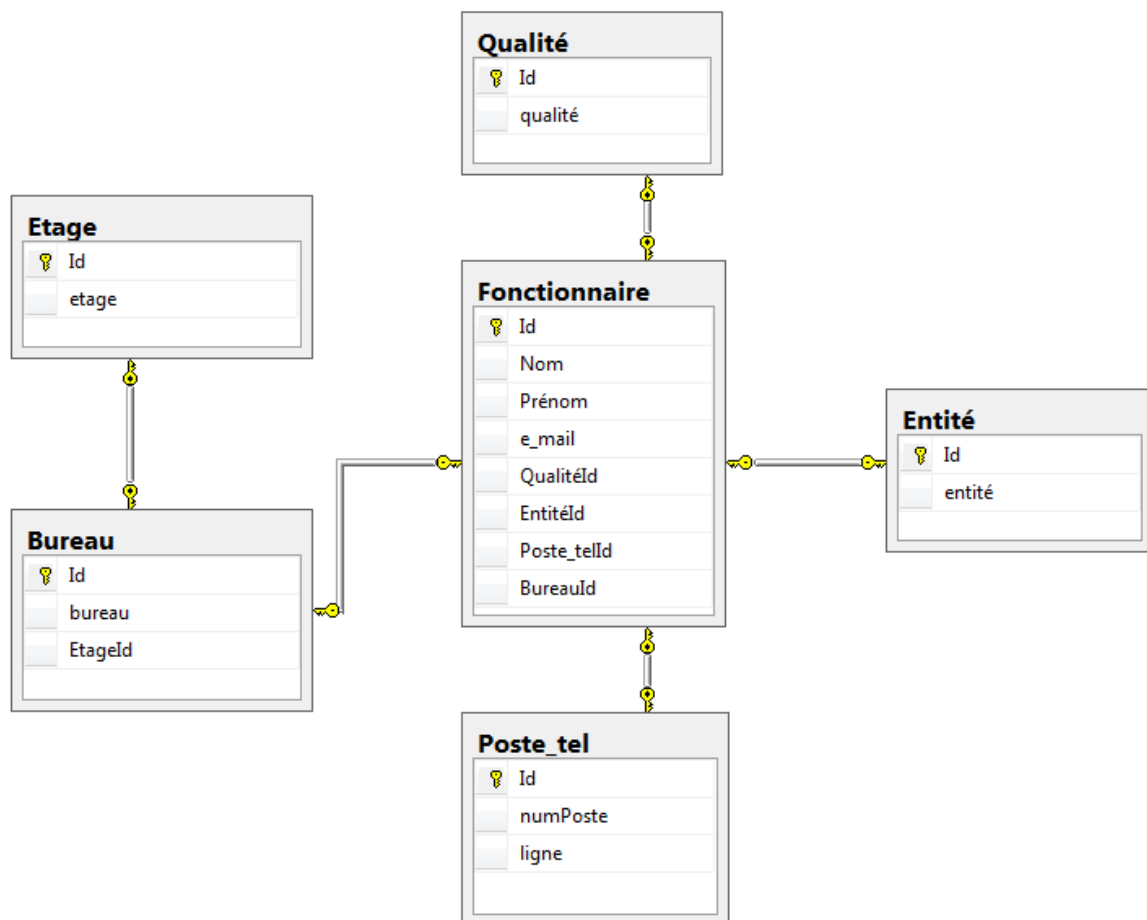
- le diagramme de classe
- le modèle conceptuel de données
- le modèle physique de données

1. Diagramme de classe

Le diagramme de classes est considéré comme le plus important de la modélisation orientée objet, il est le seul obligatoire lors d'une telle modélisation. Il décrit les structures d'objets et d'informations utilisées par notre application, à la fois en interne et dans la communication avec ses utilisateurs. Il décrit les informations sans faire référence à une implémentation particulière. Ses classes et relations vont être implémentées comme es tables de bases de données.

2. Modèle conceptuel de données

Le Modèle Conceptuel des Données permet d'écrire les données qui seront utilisées par le système d'information. Il permet de représenter la structure du système d'information, du point de vue des données, et définit également les dépendances ou relations entre ces différentes données. Il s'agit donc d'une représentation des données, facilement compréhensible, permettant de décrire le système d'information à l'aide d'entités.



3. Conclusion

En informatique, l'étape de l'analyse est souvent victime de préjugé. Nous tenterons ici de rectifier la situation ! L'analyse est malheureusement perçue comme une perte de temps et d'argent... alors que c'est tout le contraire.

Une bonne analyse va permettre de connaître les besoins présents et futurs, et d'établir un bon plan de match. Une maison ne peut être érigée sans discuter du projet, réaliser une série de plans, afin de déterminer tous les détails de la construction. Si un agrandissement est prévu pour le futur, l'architecte en tiendra compte dans le plan pour ne pas empêcher la réalisation d'un rêve. Il en va de même pour un projet informatique.

La phase de conception est la dernière étape avant l'implémentation de notre application, elle est indispensable pour notre projet informatique car elle définit l'ensemble des briques constitutives de l'application et leurs interfaces. La conception est donc une étape à ne pas négliger, elle n'est pas plus importante qu'une autre au niveau contractuel, mais techniquement une conception bâclée plantera un projet de manière sûre et certaine.

Chapitre 3

Outils de développement et réalisation de l'application

La partie réalisation est la partie la plus pesante dans un tel projet, car il constitue la partie de mise en évidence de l'application de notre projet. Dans ce chapitre, on va traiter le processus de la réalisation de cette application de gestion, tout en parlant sur les outils utilisés dans le développement de l'application.

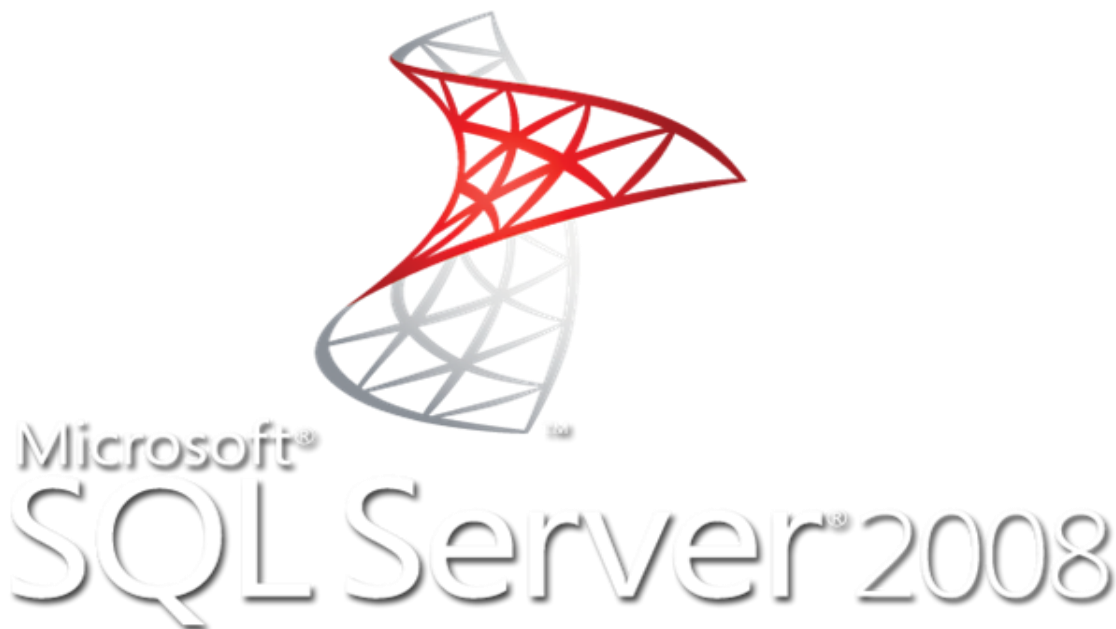
I. Technologies et outils de développement

1. SQL Server Management studio

La solution développée utilise comme système de gestion de base de données (SGBD) Microsoft SQL Server 2014. Comme tout autre (SGBD), il permet le stockage et la gestion des données. Il a pour particularité d'être un SGBD supportant nativement des requêtes SQL qui impliquent plusieurs bases de données. Microsoft SQL Server utilise le langage T-SQL (Transact-SQL) pour ses requêtes. Il s'agit d'une implémentation de SQL prenant en charge les procédures stockées 14 et les déclencheurs 15. Il possède également une composante transactionnelle qui lui permet de préparer des modifications sur les données d'une base et de les valider ou de les annuler de façon atomique. Cela permet de garantir la cohérence et l'intégrité des informations stockées dans la base de données.

a- Définition

Microsoft SQL Server est un système de gestion de base de données (abrégié en SGBD) incorporant entre autres un SGBDR (SGBD relationnel ») développé et commercialisé par la société Microsoft. Il ne fonctionne que sous les OS Windows.



b- Utilité d'une base de données

Une base de données permet de mettre des données à la disposition d'utilisateurs pour une consultation, une saisie ou bien une mise à jour, tout en s'assurant des droits accordés à ces derniers. Cela est d'autant plus utile que les données informatiques sont de plus en plus nombreuses. Une base de données peut être locale, c'est-à-dire utilisable sur une machine par un utilisateur, ou bien répartie, c'est-à-dire que les informations sont stockées sur des machines distantes et accessibles par réseau.

L'avantage majeur de l'utilisation de la base de données est la possibilité de pouvoir être accédées par plusieurs utilisateurs simultanément.

c- La gestion de la base de données

Afin de pouvoir contrôler les données ainsi que les utilisateurs, le besoin d'un système de gestion s'est vite fait ressentir. La gestion de la base de données se fait grâce à un système appelé SGBD (système de gestion de bases de données).

2. Microsoft Visual Studio

Microsoft Visual Studio est une suite de logiciels de développement pour Windows conçue par Microsoft. La dernière version s'appelle **Visual Studio 2015**. Visual Studio est un ensemble complet d'outils de développement permettant de générer des applications web ASP.NET, des services web XML, des applications bureautiques et des applications mobiles. Visual Basic, Visual C++, Visual C# utilisent tous le même environnement de développement intégré (IDE), qui leur permet de partager des outils et facilite la création de solutions faisant appel à plusieurs langages. Par ailleurs, ces langages permettent de mieux tirer parti des fonctionnalités du Framework .NET, qui fournit un accès à des technologies clés simplifiant le développement d'applications web ASP et de services web XML grâce à Visual Web Developer.



Langages supportés par Visual Studio :

- Visual basic
- Visual C++
- Visual C#
- Visual Web Developer
- SQL Server
- C

3. Outils de développement de l'application

a. ASP.NET

ASP.NET est un ensemble de technologies développées par Microsoft permettant la programmation d'applications web dynamiques. Il s'agit du successeur de la technologie Active Server Pages et fait partie de la plateforme .NET de Microsoft (que je décrirais plus loin dans ce rapport). Les pages ASP.NET sont compilées et exécutées par le Common Language Runtime. Cet ensemble de technologies permet de développer des applications web de manière rapide tout en leur fournissant une certaine robustesse. Il existe trois modèles de programmation en ASP.NET :



i. Le modèle ASP.NET WebForms :

est un des trois modèles de programmation pour créer des sites et des applications web en ASP.NET. Il s'agit d'éléments d'interface qui peuvent être positionnés et qui offrent la possibilité à l'utilisateur de modeler lui-même l'apparence des différentes pages de son application web via une interface de type glisser-déposer.

Ce modèle a pour avantage de fournir une vaste palette de contrôle permettant le prototypage rapide d'une application. Cependant, il n'est pas possible de contrôler directement le code HTML généré en sortie.

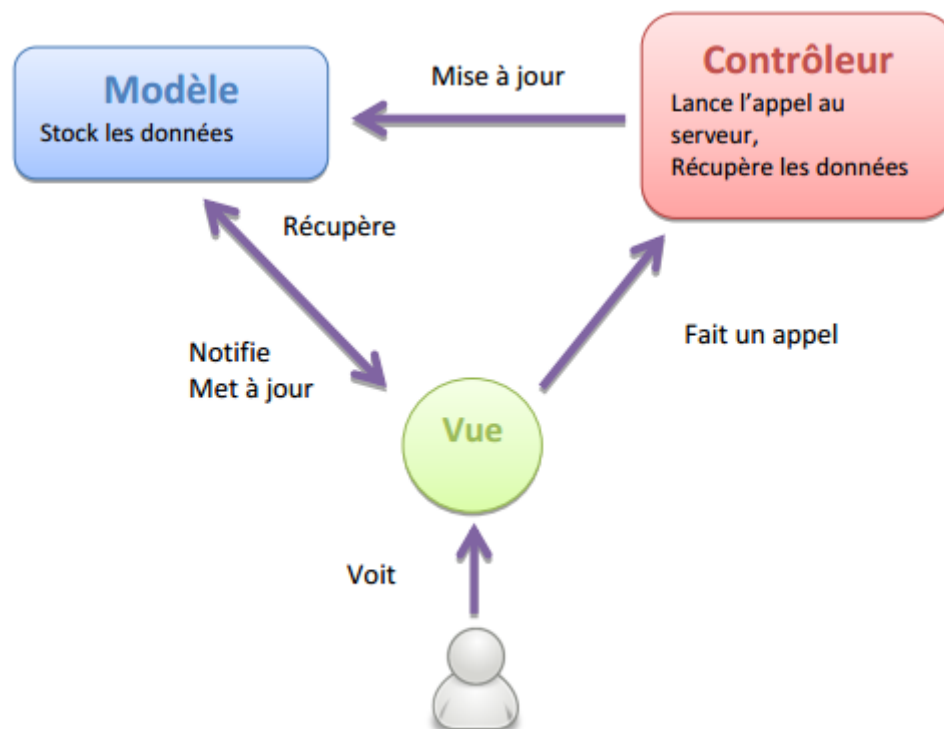
ii. Le modèle ASP.NET MVC :

qui, comme son nom l'indique, permet de créer des applications web basées sur le MVC (Model View Controller) : Le modèle (model) décrit les données manipulées par l'application telles que les interactions avec la base de données et est responsable de leur intégrité. La vue représente la partie de l'application qui dirige l'affichage des données récupérées depuis le modèle. Enfin, le contrôleur est la partie qui s'occupe de la gestion des interactions de l'utilisateur

avec l'application. En général, le contrôleur lit les données affichées depuis la vue, récupère les interactions de l'utilisateur et envoie l'ensemble des données au modèle.

L'avantage principal de ce modèle est cette séparation de la vue, du modèle et du contrôleur rendant le code plus facile à maintenir. De plus, contrairement au modèle WebForms, le développeur a un contrôle total sur le code HTML rendu. En contrepartie, le temps de développement d'une application est plus élevé.

L'architecture MVC (Modèle-Vue-Contrôleur) permet aux développeurs de d'écrire de façon structurée leur code. Le modèle MVC établit le diagramme suivant : L'utilisateur ne voit qu'une vue, qui fait appel à un contrôleur (lui pouvant être utilisé par plusieurs vues) qui fait appel au modèle pour réaliser des traitements de données. Le résultat de ces traitements sera récupéré par la vue qui les affiche à l'utilisateur.



Le modèle MVC permet de séparer les différentes parties qui constituent une application. Cela permet de bien structurer l'application afin de faciliter le développement et la maintenance. Le modèle MVC est constitué des éléments suivant :

- Le modèle : il décrit les données utilisées. C'est lui qui va interagir avec la base de données, permettre des mises à jour, etc.

- La vue : c'est ce avec quoi l'utilisateur va interagir. Le but est de représenter les données que le modèle a renvoyées et de recevoir les actions de l'utilisateur (appui sur un bouton, saisie d'un texte, sélection d'un élément, etc.). Ces événements sont envoyés au contrôleur. La vue ne fait aucun traitement, elle se contente d'afficher les résultats et d'interagir avec l'utilisateur.
- Le contrôleur : il pilote l'application. Il prend en charge la gestion des événements pour mettre à jour la vue ou le modèle. Il reçoit les interactions de l'utilisateur et lance les actions à effectuer. Si une action nécessite un changement des données, le contrôleur demande la modification des données au modèle, et ce dernier notifie la vue que les données ont changé pour qu'elles se mettent à jour. Certains événements de l'utilisateur ne concernent pas les données mais la vue. Dans ce cas, le contrôleur demande à la vue de se modifier.

b. Framework .Net

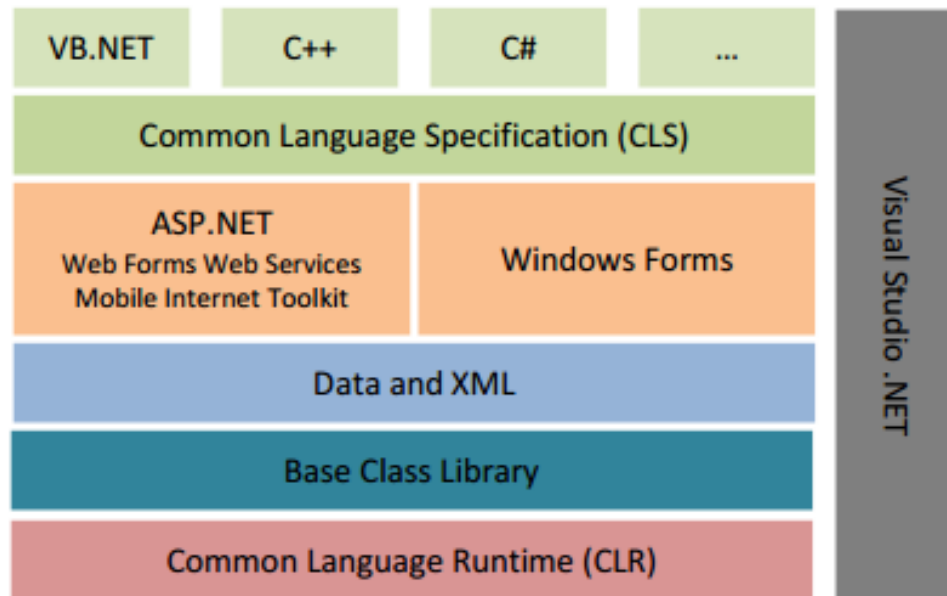
Framework .Net est une plateforme de développement et d'exécution créée par Microsoft destinée au développement de tout type d'applications :

- applications Windows : application client/serveur (Win Form/WPF), service windows
- applications Web (ASP.NET), services web
- applications mobiles

Le Framework est indépendant du langage utilisé pourvu que celui-ci réponde à la CLS (Common Language Specification). La CLS décrit les règles à suivre pour que le langage soit compatible avec le Framework .Net. La plateforme est principalement constituée :

- d'une bibliothèque de classe (Base Class Library) qui comprend un grand nombre de fonctions communes telles que la lecture et l'écriture, l'interface utilisateur, l'accès aux données, la connexion à une base de données, etc. La bibliothèque est exploitable par tous les langages compatibles
- d'une définition commune des types de bases (CTS : Common Type System, CLS : Common Language Specification)
- d'une machine virtuelle (CLR : Common Language Runtime).

L'environnement de développement utilisé pour la réalisation du projet est Microsoft Visual Studio 2015.



c. Langage C#

Le C# est un langage de programmation orientée objet créé en 2002 par Microsoft. Il a été créé afin que la plateforme .Net possède un langage permettant d'exploiter toutes ces capacités. Au niveau de la syntaxe, le C# se rapproche beaucoup du Java. En tant que langage orienté objet, le C# prend en charge les concepts d'héritage, d'encapsulation et de polymorphisme. Les variables et méthodes sont encapsulées dans des définitions de classes. Une classe peut hériter d'une seule autre classe parente et peut implémenter plusieurs interfaces. Comme je l'ai expliqué avant, la compilation d'un code en C# ne donne pas un code binaire utilisable par la machine directement, contrairement au C ou C++. Le code C# est transformé en code intermédiaire (CIL). Ce code n'est pas exécutable par la machine vu qu'elle ne comprend que le binaire. Le code CIL correspondant au programme que l'on veut distribuer. Sous Windows, ça prend l'apparence d'un .exe. Lorsqu'on exécute un programme CIL, il est lu par la machine virtuelle (CLR) qui compile le code en un vrai programme machine codé en binaire et exécutable par la machine. La CLR vérifie également la sécurité du code. Le code C# est donc exécuté sous le contrôle de la machine virtuelle. On parle de code managé.



1- HTML & CSS

Le HTML (HyperText Markup Language) est un format de présentation de données permettant de créer des pages web pouvant être lues dans des navigateurs. C'est un langage de description de données, et non un langage de programmation. Il est figé c'est-à-dire qu'une fois le document chargé dans le navigateur, il ne répond à aucune action de l'utilisateur sur le contenu de la page.

CSS est un langage de style qui définit la présentation des documents HTML. Par exemple, CSS couvre les polices, les couleurs, les marges, les lignes, la hauteur, la largeur, les images d'arrière-plan, les positionnements évolués et bien d'autres choses.

CSS offre plus d'options et se montre plus précis et sophistiqué pour la présentation des sites Web.

CSS est pris en charge par tous les navigateurs actuels. HTML sert à structurer le contenu alors que

CSS sert à formater un contenu structuré. CSS offre aux concepteurs Web des possibilités de présentations sophistiquées, gérées par tous les navigateurs. En même temps, séparer le style de présentation des documents de leur contenu rend leur entretien beaucoup plus facile.



II. Réalisation de l'application

Nous avons pensé dans la réalisation de notre application de faire une page d'accueil dans laquelle on va mettre des informations et des actualités concernant la société.

1- Page d'accueil



