

FACULTE DES SCIENCES ET TECHNIQUES FES

DEPARTEMENT D'INFORMATIQUE



Projet de Fin d'Etudes

Licence Sciences et Techniques Génie Informatique

APPLICATION DE GESTION DE COURRIERS ET BORDEREAUX



Lieu de stage : DIRECTION DE LA METEOROLOGIE NATIONALE

Réalisé par :

BARGHOUTI Yasmine

Encadré par :

Pr. KHARROUBI Jamal

Mr. RAHMI Ali

Soutenu le 19/06/2015 devant le jury composé de :

Pr. KHARROUBI Jamal

Pr. ZARGHILI Arsalan

Pr. LAMRINI Loubna

Remerciements

Je tiens tout d’abord à remercier M. Rahmi Ali de m’avoir accueilli, de son soutien inconditionnel, de sa disponibilité, et de son aide qui aura été inestimable afin de pouvoir achever ce travail dans les meilleures conditions.

Je remercie vivement Pr. Kharroubi Jamal mon encadrant interne pour ses efforts louables lors du suivi de mon projet de fin d’étude, ainsi que pour ses orientations continues et pour le grand soutien qui m’a apporté.

Enfin, je tiens à exprimer ma sincère reconnaissance à tous ceux qui ont contribué de près ou de loin à la réalisation de ce travail.

Dédicace :

*Je dédie ce travail à toute personne qui m'a donné l'esprit d'être formés, et
qui m'a soutenue dans les moments les plus durs.*

À mes parents

Pour leur grande et intarissable affection

À tous mes amis.

Pour leur soutien moral

Et à tous ceux qui me sont chers.

Sommaire :

Remerciements	2
Dédicace :	3
Sommaire :	4
Liste des Figures :	6
Introduction :	8
Chapitre 1: Présentation de La direction de météorologie :	9
I. En général :	10
I. Organigramme :	11
II. Fiche technique :	12
Chapitre 2 : Présentation du projet	13
I. Etude préalable	14
1- Introduction :	14
2- Etude de l'existant :	14
II. Enoncé du problème :	15
III. Solution proposée:	16
Chapitre 3 : Analyse Fonctionnelle	17
I. Methodologie :	18
1- Le langage utilisé : UML.....	18
2- Le modèle utilisé : itératif et incrémentale	18
3- L'architecture Modèle-Vue-Contrôleur :	20
II. MODELISATION	21
1- Les acteurs :	21
2- Les rôles :	21
3- CONCEPTION UML :	22
Chapitre 4 : Présentation de l'application	33
I. Outils de développement	34
1- Framework CodeIgniter et PHP :	34

2-	WAMP SERVER :	36
3-	MyPHPAdmin :	36
4-	Apache :	36
5-	MySQL :	36
6-	DataTable :	36
7-	JavaScript :	37
8-	Ajax :	37
9-	JQuery :	37
10-	CSS :	37
II.	Aperçu de l'application	38
1-	L'authentification :	38
2-	Informations personnelles:	38
3-	Page d'accueil Administrateur :	39
4-	Les divisions (ADMIN):	39
5-	Les utilisateurs (ADMIN):	40
6-	La traçabilité (ADMIN):	40
7-	Ajouter une division (ADMIN):	41
8-	Page d'accueil de la secrétaire :	41
9-	Table des courriers (secrétaire) :	42
10-	Ajouter un courrier (secrétaire) :	42
11-	Synthèse (secrétaire) :	42
12-	Envoyer un e-mail (secrétaire) :	43
13-	Calendrier d'événements (secrétaire) :	43
	Conclusion	45
	Perspectives :	46
	Webographie et bibliographie :	47

Liste des Figures :

Figure 1 : Organigramme.....	11
FIGURE 2 : FICHE TECHNIQUE	12
FIGURE 3 : TABLEAU DE COURRIER ENTRANT	14
FIGURE 4 : TABLEAU DE COURRIER SORTANT	15
FIGURE 5: CYCLE DE VIE DU MODELE ITERATIF INCREMENTALE.....	19
FIGURE 6 : ARCHITECTURE MVC	20
FIGURE 7 : ECHANGE D'INFORMATIONS ENTRE LES ELEMENTS.....	20
FIGURE 8 : LES ACTEURS.....	21
FIGURE 9 : ROLES DES UTILISATEURS	21
FIGURE 10 : USE CASE ADMINISTRATEUR	22
FIGURE 11: USE CASE SECRETAIRE	24
FIGURE 12 : USE CASE VISITEUR.	25
FIGURE 13 : DIAGRAMME DE SEQUENCE/AUTHENTIFICATION.....	26
FIGURE 14 : DIAGRAMME DE SEQUENCE/MODIFICATION	27
FIGURE 15 : DIAGRAMME DE SEQUENCE/AJOUT.....	28
FIGURE 16 : DIAGRAMME DE SEQUENCE/SYNTHESE.....	29
FIGURE 17 : DIAGRAMME DE SEQUENCE/AJOUT DE DIVISION	30
FIGURE 19 : TABLE DE MATIERE DE CODEIGNITER.....	35
FIGURE 20: FENETRE D'AUTHENTIFICATION	38
FIGURE 21: FENETRE D'INFORMATIONS PERSONNELLES	38
FIGURE 22 : PAGE D'ACCUEIL ADMIN	39
FIGURE 23 : LISTE DES DIVISIONS	39



FIGURE 23 : LISTE DES D'UTILISATEURS	40
FIGURE 24 : TABLEAU DE TRAÇABILITE DES ENTRANTS	40
FIGURE 24 : AJOUT DE DIVISION	41
FIGURE 25 : PAGE D'ACCUEIL SECRETAIRE	41
FIGURE 26 : TABLEAU DES COURRIERS ENTRANTS	42
FIGURE 26 : AJOUTER DES COURRIERS ENTRANTS	42
FIGURE 27 : SYNTHESE	43
FIGURE 28 : ENVOIE D'EMAIL.....	43
FIGURE 29 : CALENDRIER	44

Introduction :

Dans le cadre de notre formation au sein de la Faculté des sciences et techniques de Fès, nous avons été amenés à améliorer nos compétences et notre savoir-faire par divers moyens imposés par notre établissements tels que : les mini-projets, les projets de fin d'études ...

Le présent document constitue le fruit de mon travail accompli dans le cadre de stage de fin d'études dont l'objectif est la conception et la réalisation d'une application de gestion électronique des courriers et bordereaux administratifs en utilisant des outils tels que le Framework CODEEIGNITER , WampServer...

Au sein de l'Administration Nationale de météorologie, la secrétaire de direction, poste faisant objet de notre intervention, a une position clé au sein de l'organisme car elle est la pierre angulaire du dispositif administratif. Notre rôle est de mettre à sa disposition une application pouvant assouplir au maximum les tâches qui lui incombent, tout en gardant l'efficacité dont elle fait preuve. La secrétaire est en effet responsable de la réception des courriers entrant et sortant, leur traitement (garder une trace écrite, transfert du bon de liaison à la division adéquate, envoi d'email au chef de division et gestion des rendez-vous), à cela s'ajoute la nécessité de donner un droit de vue au différents éléments de la division sur les traces de gestion des courriers, entrant et sortant. Dans le but de donner une autonomie complète à cette application, un compte administrateur devait être mis en place, afin de gérer les comptes et les divisions. Deux mois de stage ont été nécessaires afin d'accomplir ces différents objectifs.

Durant la période du stage, la première phase était l'étude de l'existant, en établissant un cahier de charges qui définit les besoins réels de l'application, puis chercher une solution optimale qui va respecter tout ce qui a été demandé, ensuite élaborer la phase mise en œuvre de l'application. Et enfin c'est les tests de validation du projet.

Ce rapport comporte quatre chapitres. Le premier chapitre présente le lieu du stage, le second comporte le cahier des charges, l'étude de l'existant ainsi que les solutions proposées. Quant au troisième chapitre, il présente la conception du projet en exposant la structure de notre application. Enfin le quatrième chapitre étale les différents outils et langages utilisés alors que le quatrième chapitre expose la partie dynamique de l'application ainsi que ses différentes interfaces avec l'utilisateur.



Chapitre 1: Présentation de La direction de météorologie :

Dans ce chapitre nous allons présenter le lieu du stage



I. En général :

Cinquante ans après sa création, Maroc Météo a franchi des étapes fondamentales dans son évolution vers un service météorologique de haut niveau, assurant ainsi les multiples missions qui lui sont dévolues en tant que service public d'une part et en tant que prestataire de service au profit des différents secteurs économiques du pays d'autre part.

Sans chercher à être exhaustif, un bref rappel de ces étapes s'impose permettant ainsi de réaliser et de mieux comprendre comment cette évolution a pu se faire.

Dans le domaine de l'amélioration de la qualité de la prévision météorologique, les différents programmes mis en œuvre ont permis des avancées remarquables de la prévision à courte, moyenne et longue échéances.

Le développement du programme d'insémination artificielle Al Ghait, lancé dans un premier temps à titre expérimental sur une zone pilote du pays est actuellement confirmé par son extension sur tout le territoire. L'expertise de nos cadres dans ce domaine a permis le développement d'une coopération Sud-Sud avec plusieurs pays d'Afrique subsaharienne qui ont pu bénéficier de programmes similaires dans leur pays.

La surveillance de la qualité de l'air constitue un excellent exemple de domaine pour lequel la Direction s'est engagée en collaboration avec la Fondation Mohammed VI et la Wilaya du grand Casablanca, afin de mettre à la disposition des collectivités et des décideurs les outils et informations nécessaires à la compréhension et à la gestion des problèmes de pollution de l'air.

Les activités :

Au fur et à mesure que les activités se sont élargies, le besoin d'être encore plus proches des attentes de leurs partenaires s'est ressenti, et une politique de régionalisation de leurs services sur le territoire marocain a vu le jour.

Ainsi, quatre directions météorologiques régionales ont été créées (Centre, Sud, Nord et Nord-Est) afin de permettre l'élaboration et la mise à disposition de prestations plus adaptées à la région.

Le passage de la Direction en Service d'Etat Géré de Manière Autonome (SEGMA) a permis, de mettre en place une politique commerciale et une nouvelle orientation de leurs services visant la mise à disposition de leurs clients et partenaires, de multiples prestations dans les domaines du climat et de la prévision.

Cette nouvelle stratégie d'ouverture vers les attentes de leurs clients et partenaires soit par le biais de conventions soit de manière directe, les incite en permanence à améliorer et adapter leurs prestations aux secteurs économiques qui les intéressent.

Toutes ces activités de service n'ont pu se faire que par la mobilisation d'équipements importants et de haute technologie, en matière d'observation et de télécommunication et par la constante mise à niveau de leurs capacités de calcul avec l'installation entre autres de super calculateurs permettant d'assurer la puissance de calcul nécessaire à l'amélioration de leurs performances.

I. Organigramme :

Direction de la Météorologie Nationale

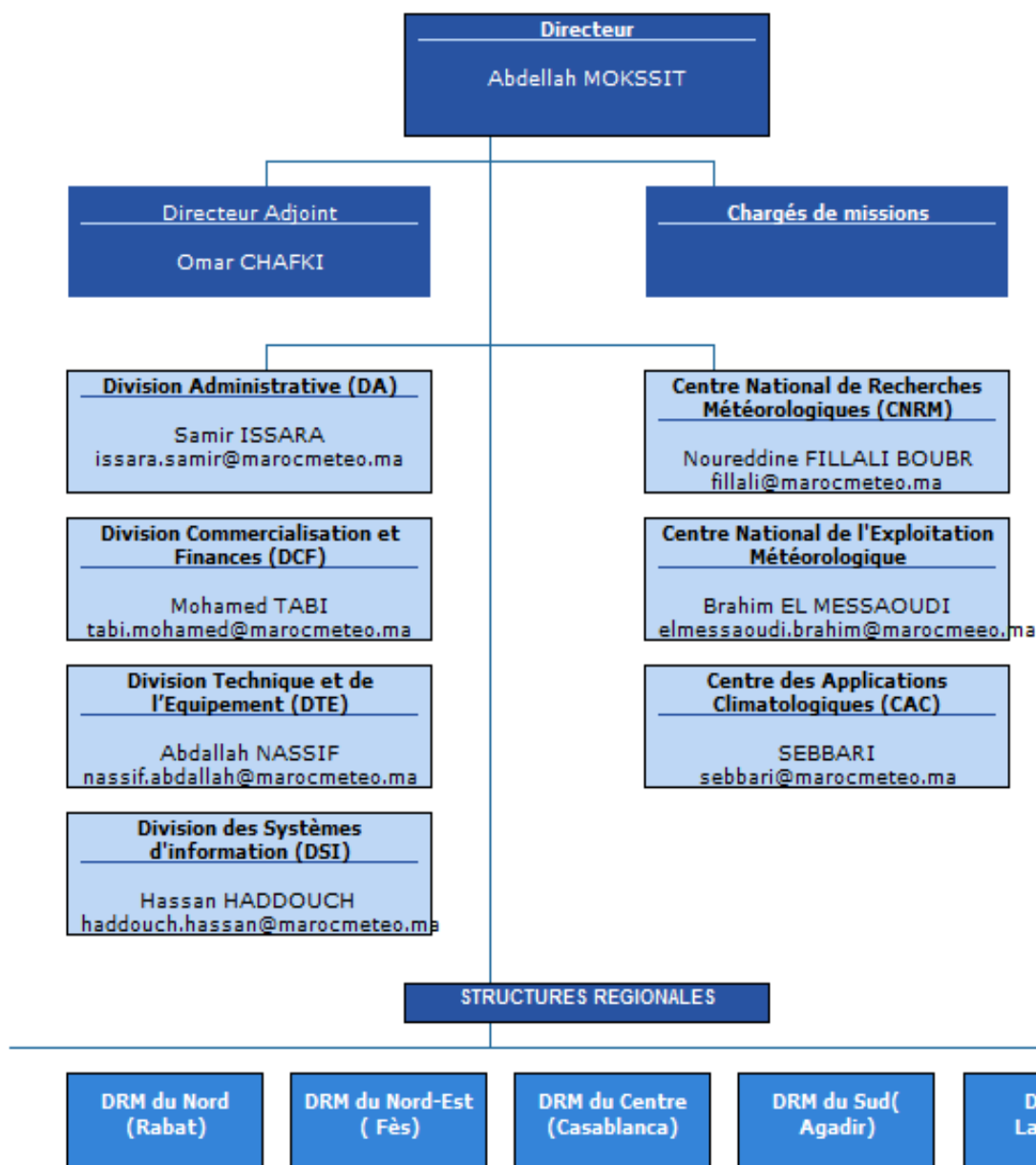


Figure 1 : Organigramme

II. Fiche technique :

Raison sociale	Direction de la météorologie nationale DMN
Date de création	1961
Siège social	B.P :8106 Casablanca oasis
Forme juridique	Administration publique
Téléphone	05 22 65 49 00
Fax	05 22 91 37 97
Internet	www.marocmétéo.ma
Directeur	Mr Abdallah MOKSSIT

FIGURE 2 : FICHE TECHNIQUE



Chapitre 2 : Présentation du projet

*Dans ce chapitre nous allons présenter le
cadre général du projet, les problèmes
affrontés et les études effectuées pour les
résoudre*



I. Etude préalable

1- Introduction :

Dans ce chapitre nous allons présenter l'étude préalable qui doit être élaboré avant d'entamer la mise en place de notre application. Pour cela, nous allons détailler en premier lieu la méthode de travail du client (DMN), puis nous allons citer les besoins essentiels demandés et les fonctionnalités requises par l'utilisateur (secrétaire), ainsi nous arriverons à trouver le problème de la méthode adopté par la société pour en déduire une solution.

2- Etude de l'existant :

Cette étape-là permet de déterminer les points forts et faibles de la méthode de travail adopté par la société, et ça va nous aider aussi à estimer les besoins du client.

Dans le cadre de la bureautique la secrétaire de la division du système d'information prend en charge les courriers et les bordereaux de manière manuelle, ce travail-là se déroule à priori en trois parties différentes;

- Partie de réception
- Partie d'envoi

a. Partie de réception :

- **Etape de traitement**

Sur un registre de grande taille, la secrétaire à l'aide d'un tableau comme celui-là :

Arrivée		Information sur l'expéditeur			L'objet du courrier
Numéro	Date	Référence	Date d'envoi	Origine	

FIGURE 3 : TABLEAU DE COURRIER ENTRANT

Enregistre les courriers entrants.

Après avoir rempli le tableau ci-dessus, la secrétaire imprime une fiche de liaison de cette forme :

Puis elle la joint au courrier pour la remettre au chef de division pour qu'il la remplisse.

- **Etape de dispatching**

Lors du retour du courrier et selon les données envoyées par le chef de division, la secrétaire doit faire ce qu'on appelle une diffusion de l'information selon ce que le chef de division a demandé, là il existe 3 services, dont les chefs doivent recevoir un email de notification qu'elle envoie en utilisant « Outlook ».

b. Partie d'envoi de courrier

Là aussi il s'agit d'un tableau que la secrétaire remplit,

ID	Date	Destination	Objet
----	------	-------------	-------

FIGURE 4 : TABLEAU DE COURRIER SORTANT

- Pour les deux parties précédentes, la secrétaire fait un suivi, en utilisant sa mémoire et des bouts de papier bien sûr, car les courriers ont besoin d'être traité et transmis selon une certaine procédure.
- Dans le cadre de l'entreprise, le personnel lui arrive de consulter le registre des courriers, mais n'a en aucun cas le droit ni la possibilité d'y introduire des modifications.

c. Partie de gestion des RDV

Le travail traditionnel d'un secrétaire est la gestion des rdv et événements, là aussi l'agenda est indispensable.

II. Enoncé du problème :

La secrétaire est d'une grande importance au sein d'une entreprise, elle assure les tâches traditionnelles de secrétariat, organise et encadre le travail administratif et les conférences, envoie des mails et rassemble les informations nécessaires, comme elle est responsable de la gestion du bureau, et elle est chargée de planifier l'agenda du supérieur.

Tout ce travail exige la bonne organisation, l'esprit de synthèse et de collaboration. C'est-à-dire que la mémoire de la personne est parfois incapable de gérer tout cela au même temps, surtout que de nos jours et dans beaucoup d'entreprises ce travail n'a pas évolué, se basant toujours sur le couple stylo/feuille, cela mène des fois à l'oubli ou l'échec d'une tâche qui se traduisent par « incompétence »...

Dans le but d'assurer une bonne fluidité du travail ainsi qu'une efficacité et un taux élevé de bonne production, il nous a été demandé de proposer une solution à la fois simple et concise pouvant aider à améliorer les conditions de travail de la secrétaire qui reste le nœud principal de liaison entre les différents services.

Dans le cadre de la formation suivie au sein de la Faculté des sciences et technique de Fès, la tâche de faire cette amélioration nous a été confiée, afin de résoudre les problèmes suivants :

Problème de gestion de l'information : (bon de liaisons, registres de courriers/bordereaux, dossiers de classification...).

- ❖ Problème de modification des traitements.
- ❖ Problème de gestion des RDV.
- ❖ Problème de classification des registres et dossiers.
- ❖ Problème de sécurité et confidentialité de l'information, l'accès peut être facile pour tout le monde.
- ❖ Problème de mémoire.

- ❖ Problème d’archivage et de recherche d’information.
- ❖ Problème de gestion d’espace.

Nous avons donc proposé la mise en place d’une application informatique afin de mieux gérer le travail.

III. Solution proposée:

Dans le cadre de la résolution des problèmes cités ci-dessus, une application Web va améliorer le travail et la gestion du service d’information au sein de la Direction de la météorologie marocaine.

Tout en utilisant le PHP, l’architecture MVC, et un bon Framework nous arriverons à faire gagner le temps à l’utilisateur de cette interface et lui permettre de faire un saut vers l’évolution du travail manuel fatigant au travail informatisé qui va comporter ce qui suit :

- ❖ Interface de la secrétaire, dans le but de mieux organiser son travail.
- ❖ Interface du visiteur, qui ne fera que voir les tables existantes.
- ❖ Interface de l’administrateur, qui fera la gestion des sessions et services ainsi que l’inspection du travail fait par la secrétaire.
- ❖ L’utilisation sécurisée de l’interface.
- ❖ Facilité d’emploi.
- ❖ La bonne gestion du temps par rapport au rendu.



Chapitre 3 : Analyse Fonctionnelle

Cette partie, est dédiée à la méthodologie de travail, d'analyse et les modèles utilisés dans l'application, ainsi que les diagrammes fait à l'aide de l'UML



I. Méthodologie :

1- Le langage utilisé : UML

Un langage de modélisation graphique à base de pictogramme conçu pour fournir une méthode normalisée pour visualiser la conception d'un système. Il est couramment utilisé en développement et en conception orientée objet.

En fait, avec l'UML, nous pouvons modéliser toutes les étapes du développement d'une application informatique, de sa conception à la mise en route grâce à des diagrammes...

Exemples de diagrammes :

- Le diagramme de « use case », permet de déterminer les différents cas d'utilisations d'un programme informatique.
- Le diagramme de séquence, permet de visualiser le déroulement d'une application dans un contexte donné...
- Le diagramme de classes, duquel on peut générer les tables utilisées dans le projet.
- Etc...

2- Le modèle utilisé : itératif et incrémentale

Nous avons opté pour ce modèle car c'est le plus répandu dans ce domaine, connu pour ses normes universelles.

- Itératif : le processus de développement est appliqué plusieurs fois.
- Incrémental : chaque itération augmente la quantité d'information.

Approche itérative et incrémentale :

- Segmentation du travail.
- Concentration sur les besoins et les risques.
- Les itérations sont des prototypes :
 - Expérimentation et validation des technologies.
 - Planification et évaluation.
- Les prototypes « s'enroulent » autour du noyau de l'architecture.

Le développement incrémental vous donne la possibilité d'améliorer votre processus de développement, ainsi que d'ajuster les exigences à l'évolution de l'environnement.

Le développement itératif vous aide à améliorer la qualité de votre produit. Oui, cela veut dire remanier et oui, vous aurez probablement besoin d'en faire un peu pour obtenir un produit propre.

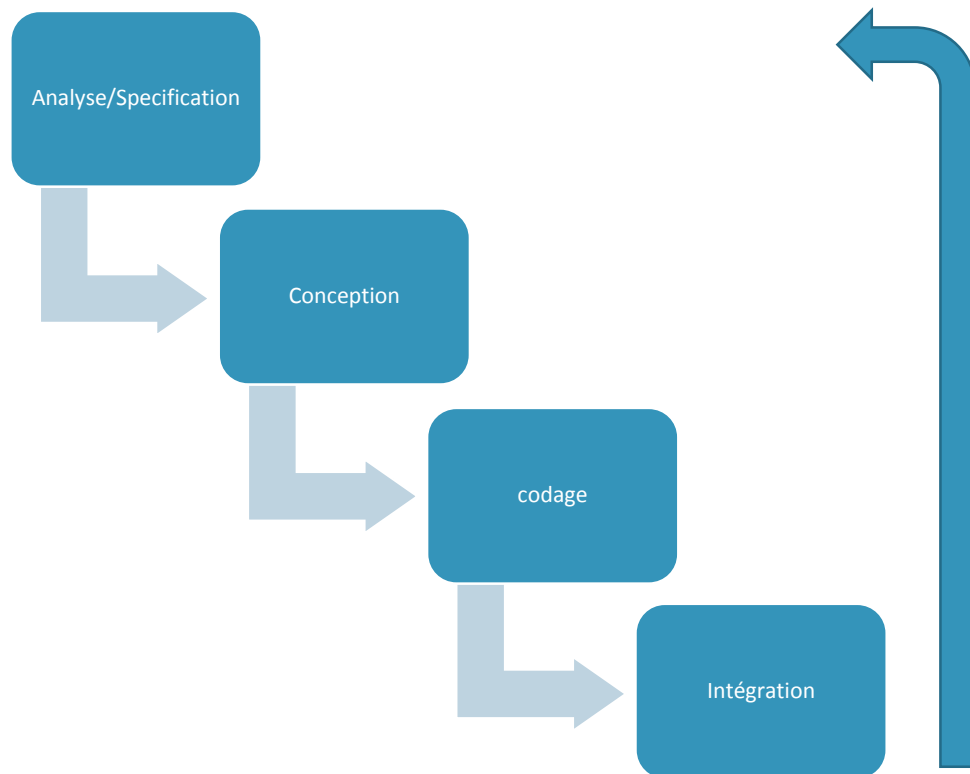


FIGURE 5: CYCLE DE VIE DU MODELE ITERATIF INCREMENTALE

Incréments du projet :

Dans le développement incrémental, nous découpons les tâches en petits morceaux et les planifions pour être développées au fil du temps et intégrées dès qu'elles sont terminées.

Dans ce cadre voici les incréments les plus importants de notre projet :

- ❖ Gestion de sessions.
- ❖ Gestion de courriers entrants.
- ❖ Gestion de courriers sortants.
- ❖ Conception et gestion du calendrier.
- ❖ Gestion d'évènements.
- ❖ Technique d'envoi d'emails.
- ❖ Gestion des contacts.
- ❖ Gestion de synthèse.
- ❖ Gestion des vues.
- ❖ Gestion de la traçabilité.

3- L'architecture Modèle-Vue-Contrôleur :

Le pattern MVC permet de bien organiser son code source. Il va vous aider à savoir quels fichiers créer, mais surtout à définir leur rôle. Le but de MVC est justement de séparer la logique du code en trois parties que l'on retrouve dans des fichiers distincts, comme l'explique la description qui suit.

Modèle : cette partie gère les *données* de votre site. Son rôle est d'aller récupérer les informations « brutes » dans la base de données, de les organiser et de les assembler pour qu'elles puissent ensuite être traitées par le contrôleur. On y trouve donc les requêtes SQL.

Vue : cette partie se concentre sur l'*affichage*. Elle ne fait presque aucun calcul et se contente de récupérer des variables pour savoir ce qu'elle doit afficher. On y trouve essentiellement du code HTML mais aussi quelques boucles et conditions PHP très simples, pour afficher par exemple la liste des messages des forums.

Contrôleur : cette partie gère la logique du code qui prend des *décisions*. C'est en quelque sorte l'intermédiaire entre le modèle et la vue : le contrôleur va demander au modèle les données, les analyser, prendre des décisions et renvoyer le texte à afficher à la vue. Le contrôleur contient exclusivement du PHP. C'est notamment lui qui détermine si le visiteur a le droit de voir la page ou non (gestion des droits d'accès).



FIGURE 6 : ARCHITECTURE MVC

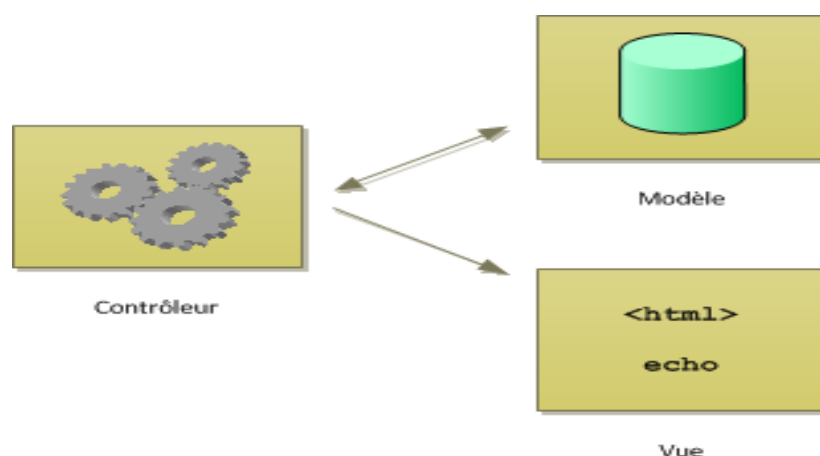


FIGURE 7 : ECHANGE D'INFORMATIONS ENTRE LES ELEMENTS

II. MODELISATION

Dans cette partie du rapport nous allons procéder à la modélisation du projet, c'est-à-dire tout ce qu'il s'agit d'acteurs, leurs rôles et les messages échangés entre les acteurs et le programme.

1- Les acteurs :



LA SECRETAIRE



L'ADMINISTRATEUR



LE VISITEUR

FIGURE 8 : LES ACTEURS

2- Les rôles :



Consulter les courrier entrants/sortants.
Ajouter/Modifier/Supprimer les courriers entrants et sortants .
Faire le dispatching.
Gérer le suivi des courriers.
Gérer les RDV.
Imprimer la synthèse du jour.



Gérer les comptes.
Gérer les groupes et les sessions.
Gérer les divisions.
Consulter les tables de la base de données.



Voir les tables .

FIGURE 9 : ROLES DES UTILISATEURS

3- CONCEPTION UML :

a. Diagrammes de cas d'utilisations :

Le cas d'utilisation est un moyen de représenter les différentes possibilités d'utiliser un système. Il exprime toujours une suite d'interactions entre un acteur et l'application. Ainsi qu'il permet de recenser les grandes fonctionnalités d'un système.

Toute personne voulant avoir des informations sur une enseigne peut accéder à cette application, par conséquent, un seul acteur « utilisateur » est pris en compte.

L'ADMINISTRATEUR :

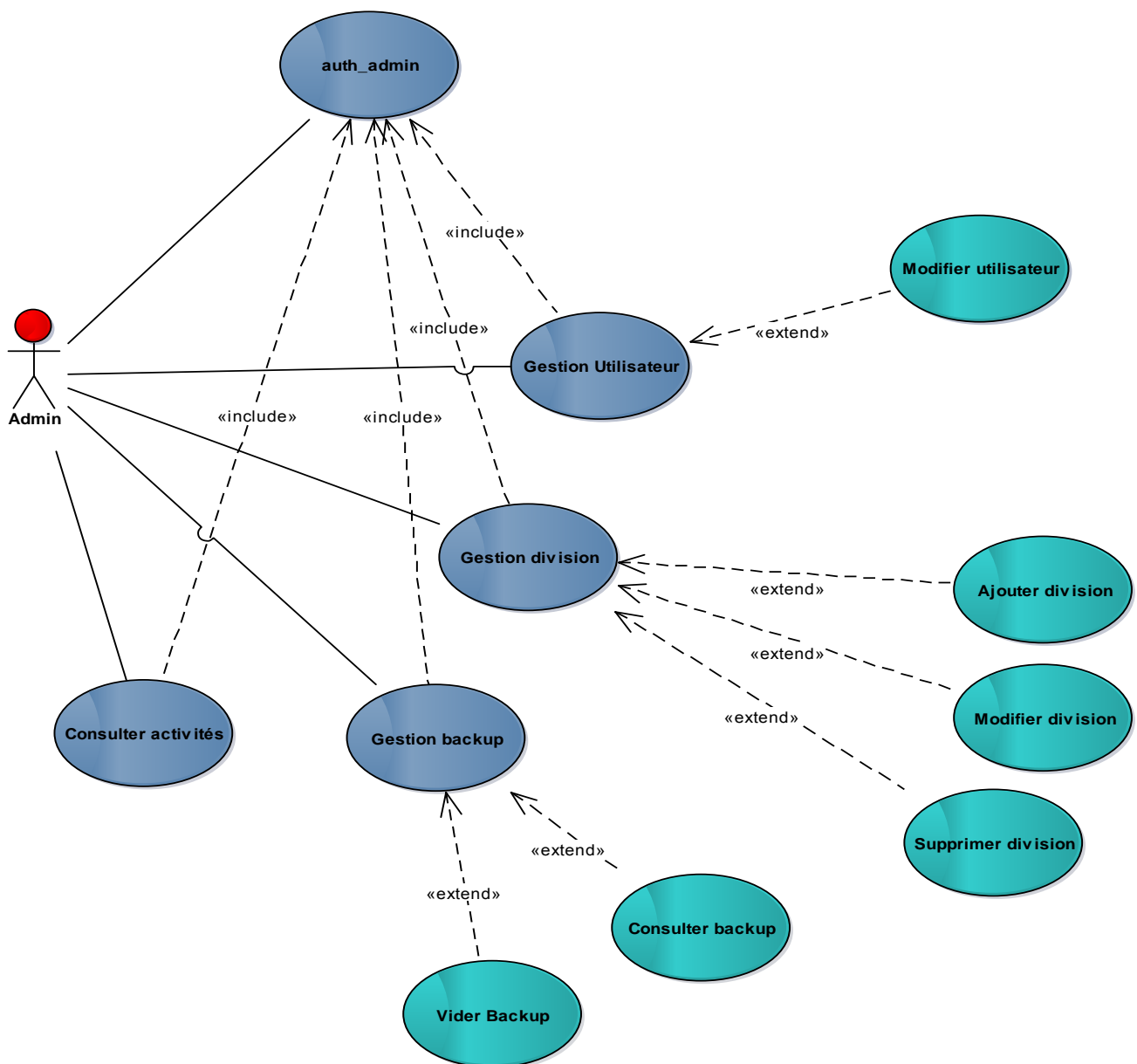


FIGURE 10 : USE CASE ADMINISTRATEUR

Description :

L'administrateur de l'interface est celui qui gère dans un premier lieu les sessions, et il est le seul à faire ce travail.

Donc, après son authentification il peut :

- ❖ Gérer les divisions : Chaque division est une session qui a une secrétaire et un visiteur, l'administrateur peut ajouter, modifier ou supprimer une division.
- ❖ Gérer les utilisateurs : un utilisateur est soit une secrétaire soit un visiteur, donc la modification concerne le nom d'utilisateur, le mot de passe et les informations personnelles. Dans ce cadre aussi l'administrateur peut modifier ses propres informations.
- ❖ Gérer le backup : le backup concerne tout ce qu'une secrétaire peut faire sur son interface, si elle ajouter, modifie ou supprime, en cas de faute ou de problème le backup est fait afin de garder une trace. Donc l'administrateur après avoir vérifié peut tout vider.
- ❖ Consulter les activités : c'est voir les tables sans pouvoir y faire une action quelconque.

LA SECRETAIRE :

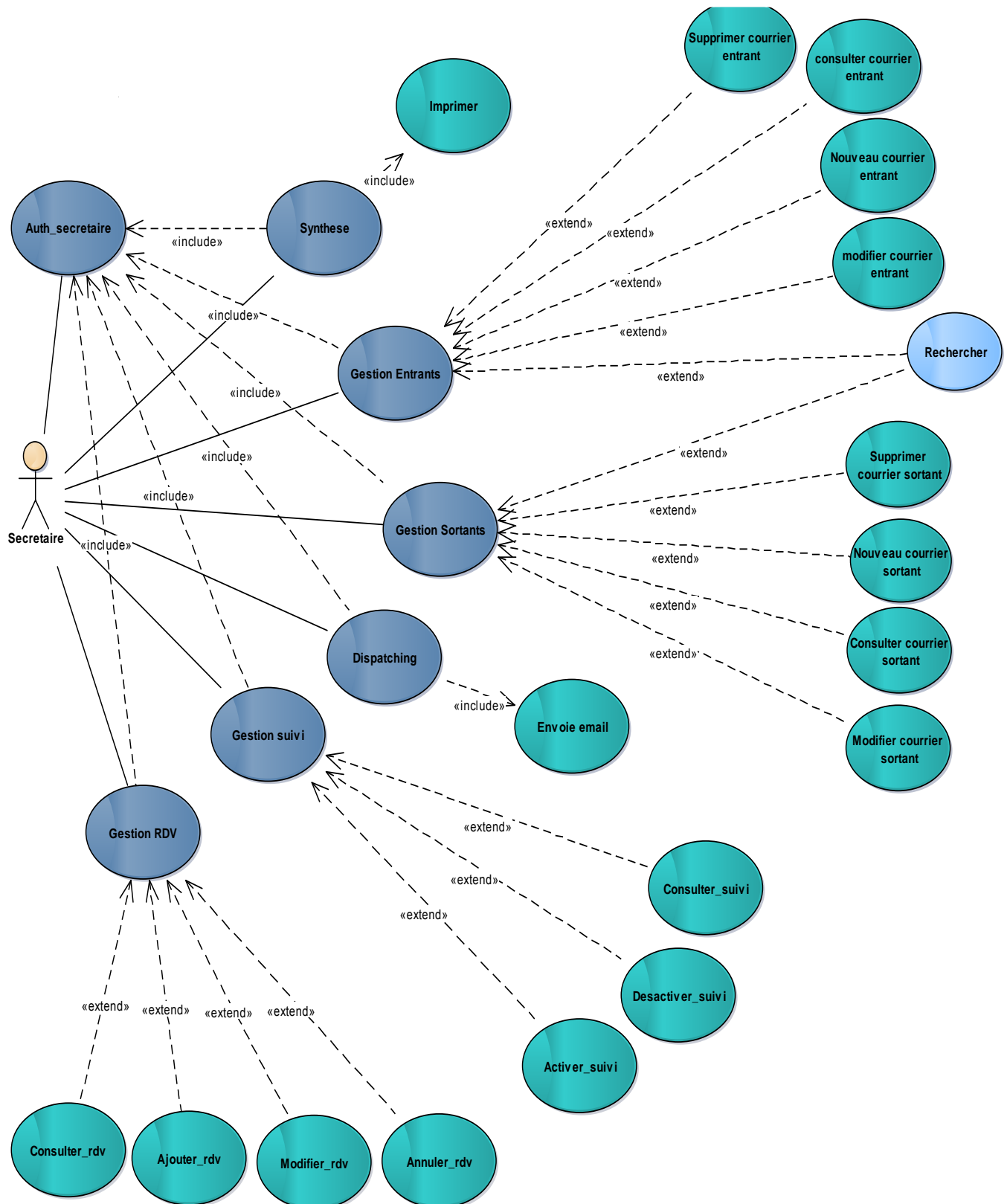


FIGURE 11: USE CASE SECRETAIRE

DESCRIPTION:

La secrétaire est l'utilisateur primaire de l'application, elle peut gérer les courriers entrant/sortants et les rendez-vous à travers sa propre interface.

- ❖ Gérer les courriers entrants/sortants : tout ce qu'il s'agit d'ajout, modification, suppression d'un courrier correspondant à la division à laquelle elle appartient.
- ❖ Gérer les suivis : Chaque courrier à une certaine méthodologie de suivi, l'interface l'aide à se rappeler des courriers qu'elle a mis en suivi.
- ❖ Gérer le calendrier : là elle note ce qu'elle a durant ses journées, et elle peut bénéficier d'un rappel par rapport aux RDV et événements.
- ❖ Le dispatching : c'est l'envoi d'email de rappel aux chefs de services de sa division.

LE VISITEUR :

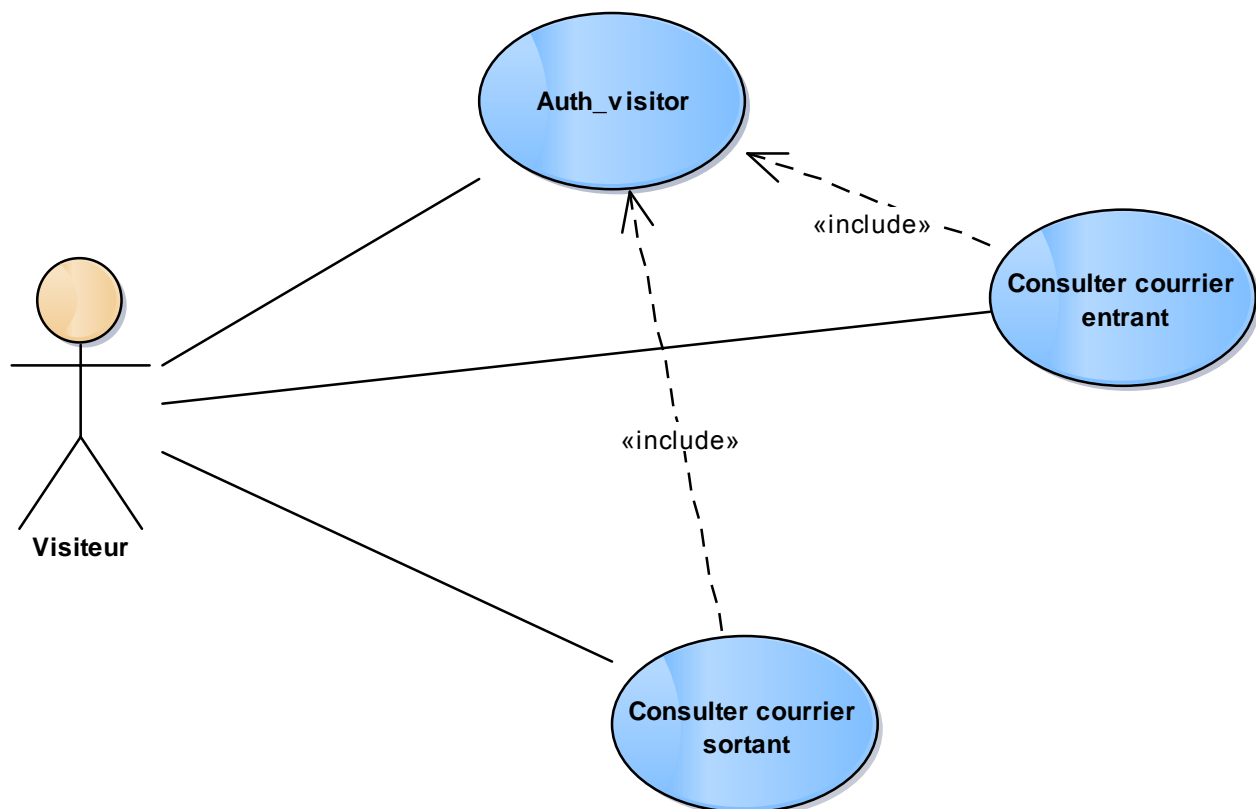


FIGURE 12 : USE CASE VISITEUR.

Description :

- ❖ Voir les tables de la division à laquelle il appartient.

b. Diagrammes de séquences :

Le diagramme de séquences permet de cacher les interactions d'objets dans le cadre d'un scénario d'un Diagramme de cas d'utilisation Dans un souci de simplification, on représente l'acteur principal à gauche du diagramme, et les acteurs secondaires éventuels à droite du système. Le but étant de décrire comment se déroulent les actions entre les acteurs ou objets.

La dimension verticale du diagramme représente le temps, permettant de visualiser l'enchaînement des actions dans le temps, et de spécifier la naissance et la mort d'objets. Les périodes d'activité des objets sont symbolisées par des rectangles, et ces objets dialoguent par le biais de messages.

Dans ce cadre, nous allons présenter quelques exemples de diagrammes de séquences utilisés dans ce projet :

L'authentification :

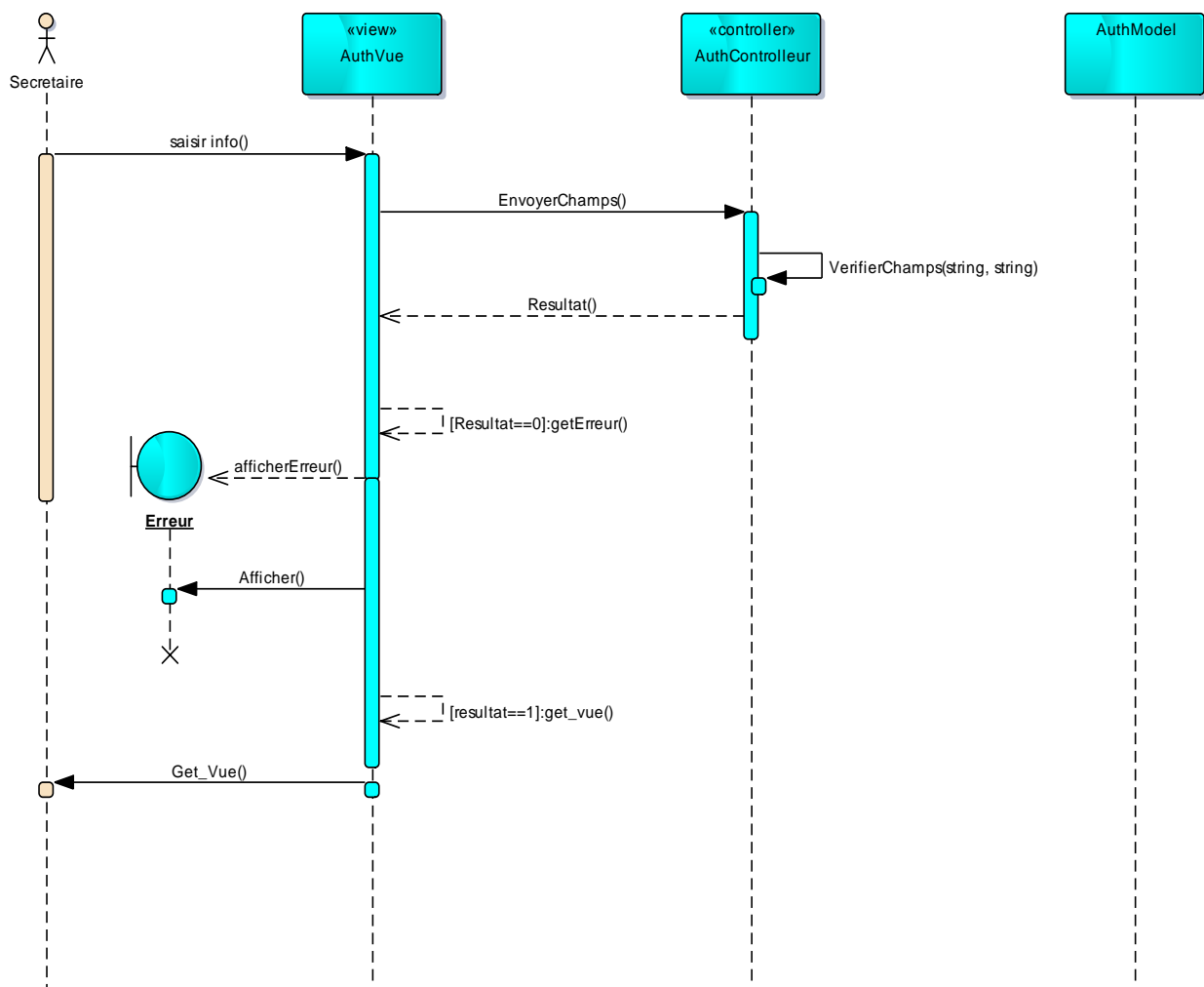


FIGURE 13 : DIAGRAMME DE SEQUENCE/AUTHENTIFICATION

La modification :

Cette fonctionnalité est utilisée par la secrétaire et l'administrateur.

En cas de :

- ❖ Modification de courrier entrant/sortant
- ❖ Modification de Contacte/adresse email
- ❖ Modification de division
- ❖ Modification de compte
- ❖ Etc...

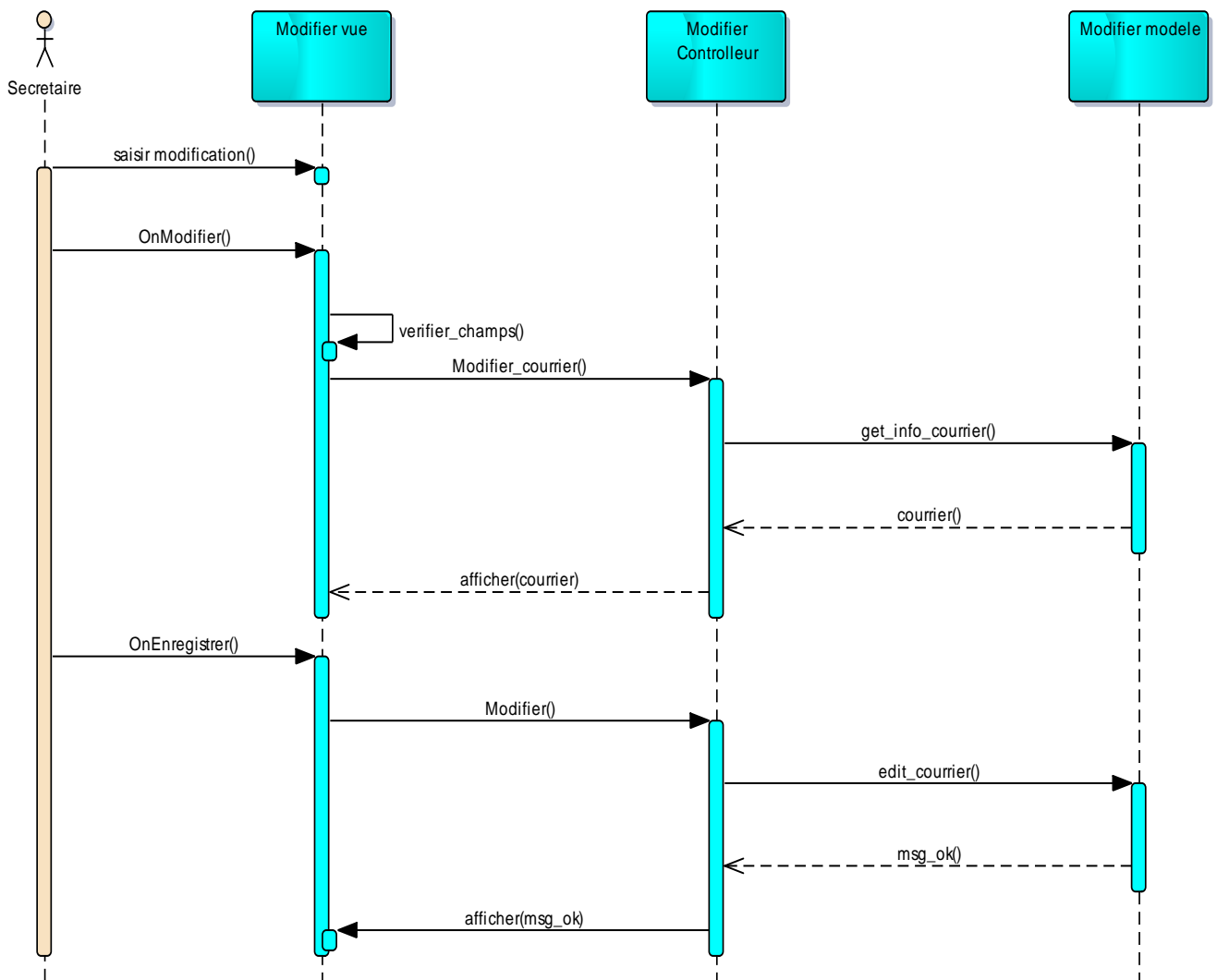


FIGURE 14 : DIAGRAMME DE SEQUENCE/MODIFICATION

L'Ajout à la base de données :

Cette fonctionnalité est utilisée par la secrétaire.

En cas de :

- ❖ Ajout de courrier entrant/sortant
- ❖ Ajout de Contacte/adresse email
- ❖ Ajout d'email envoyé
- ❖ Etc...

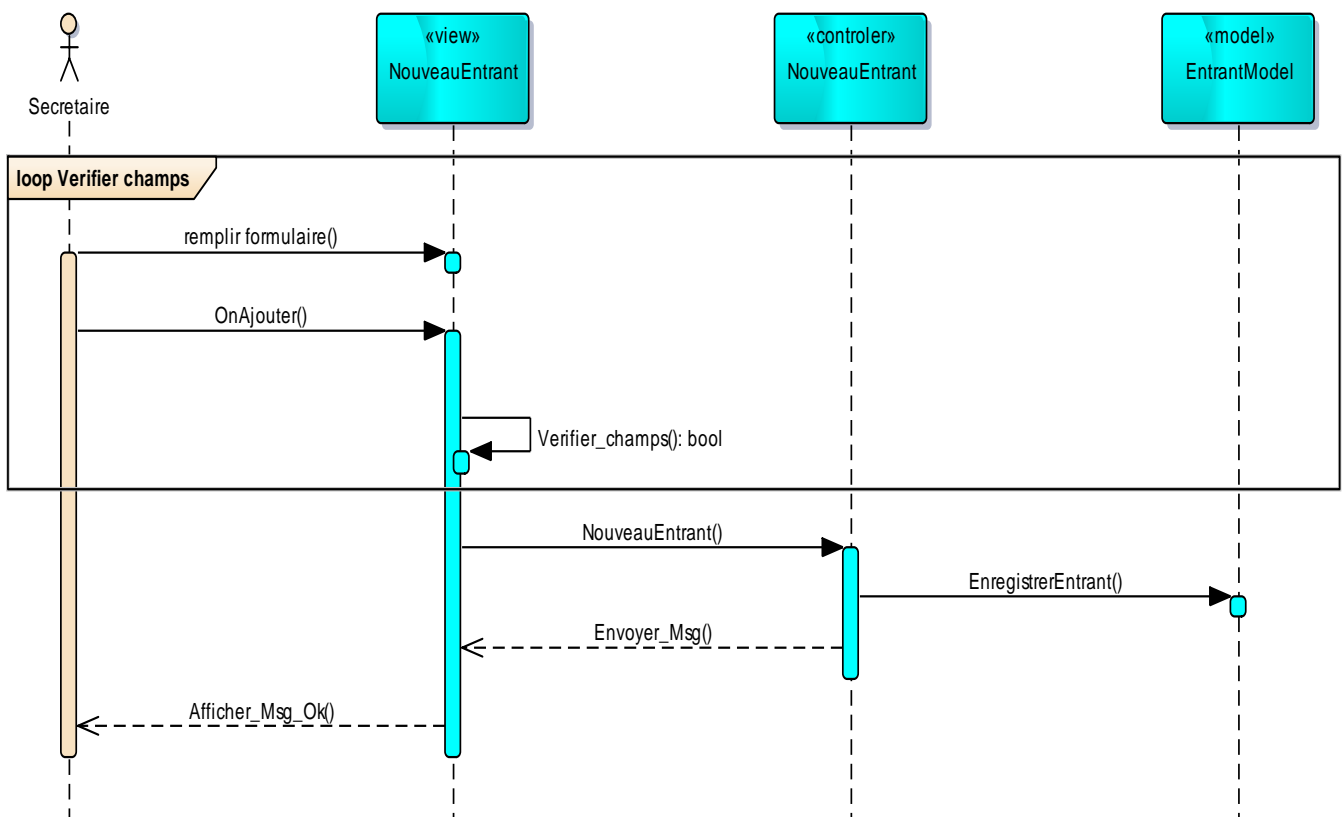


FIGURE 15 : DIAGRAMME DE SEQUENCE/AJOUT

La Synthèse :

Utilisée uniquement par la secrétaire, afin de garder une trace écrite des courriers traités selon une date précise.

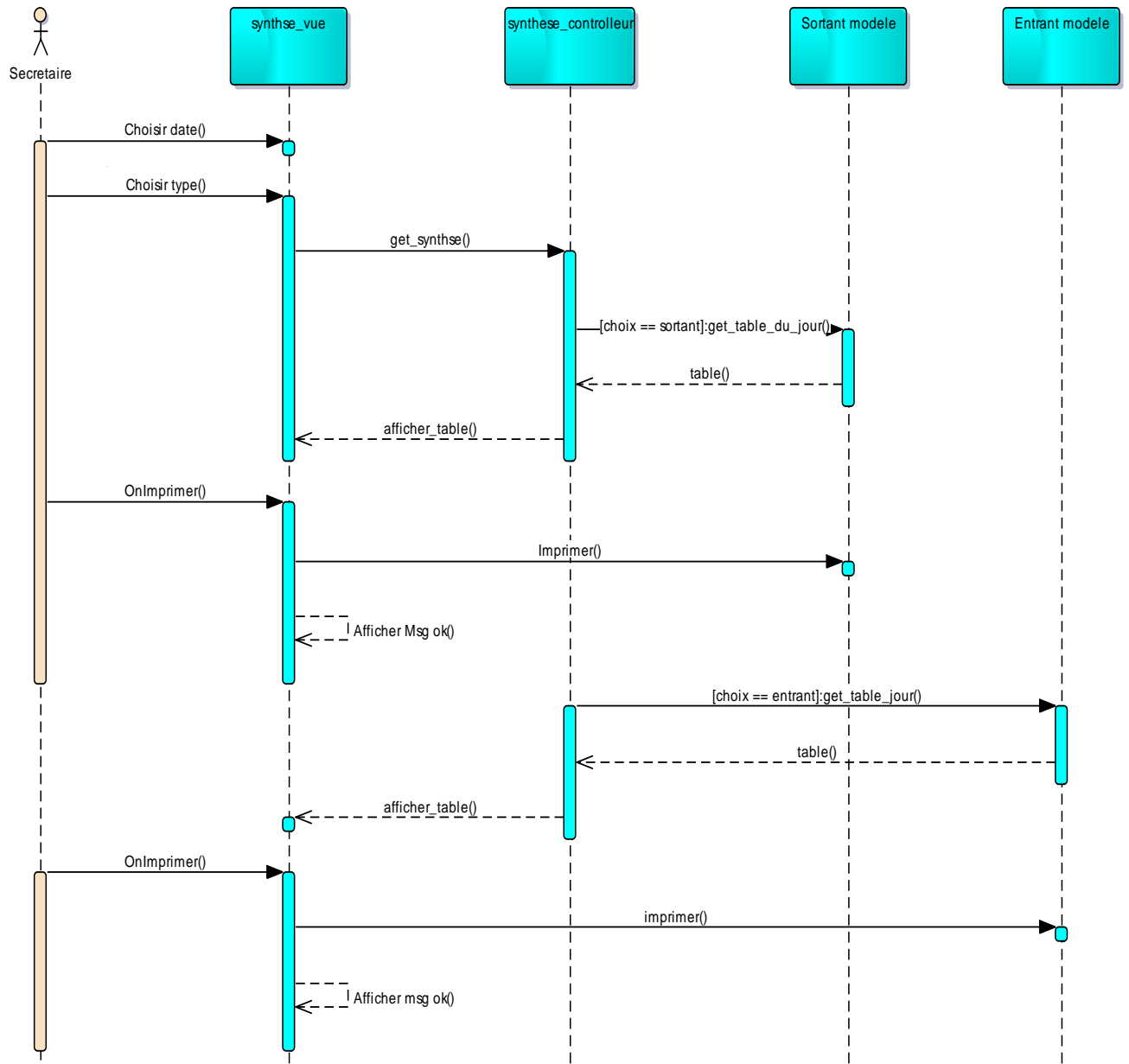


FIGURE 16 : DIAGRAMME DE SEQUENCE/SYNTHESE

Ajout de division :

Uniquement pour l'administrateur, ce qu'il lui permet d'ajouter de nouvelles sessions dites divisions, chacune avec une secrétaire et un visiteur dont les noms d'utilisateurs sont unique.

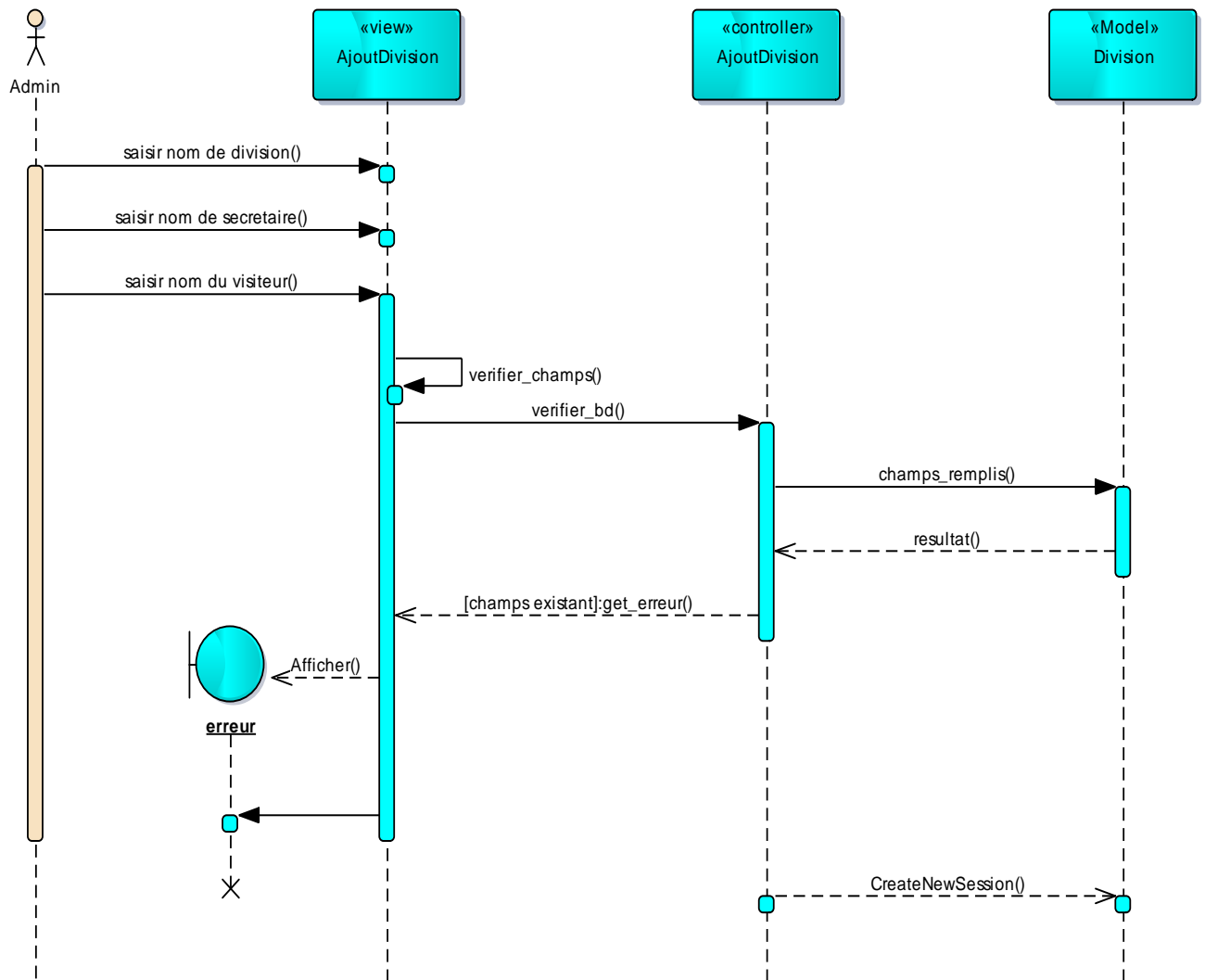
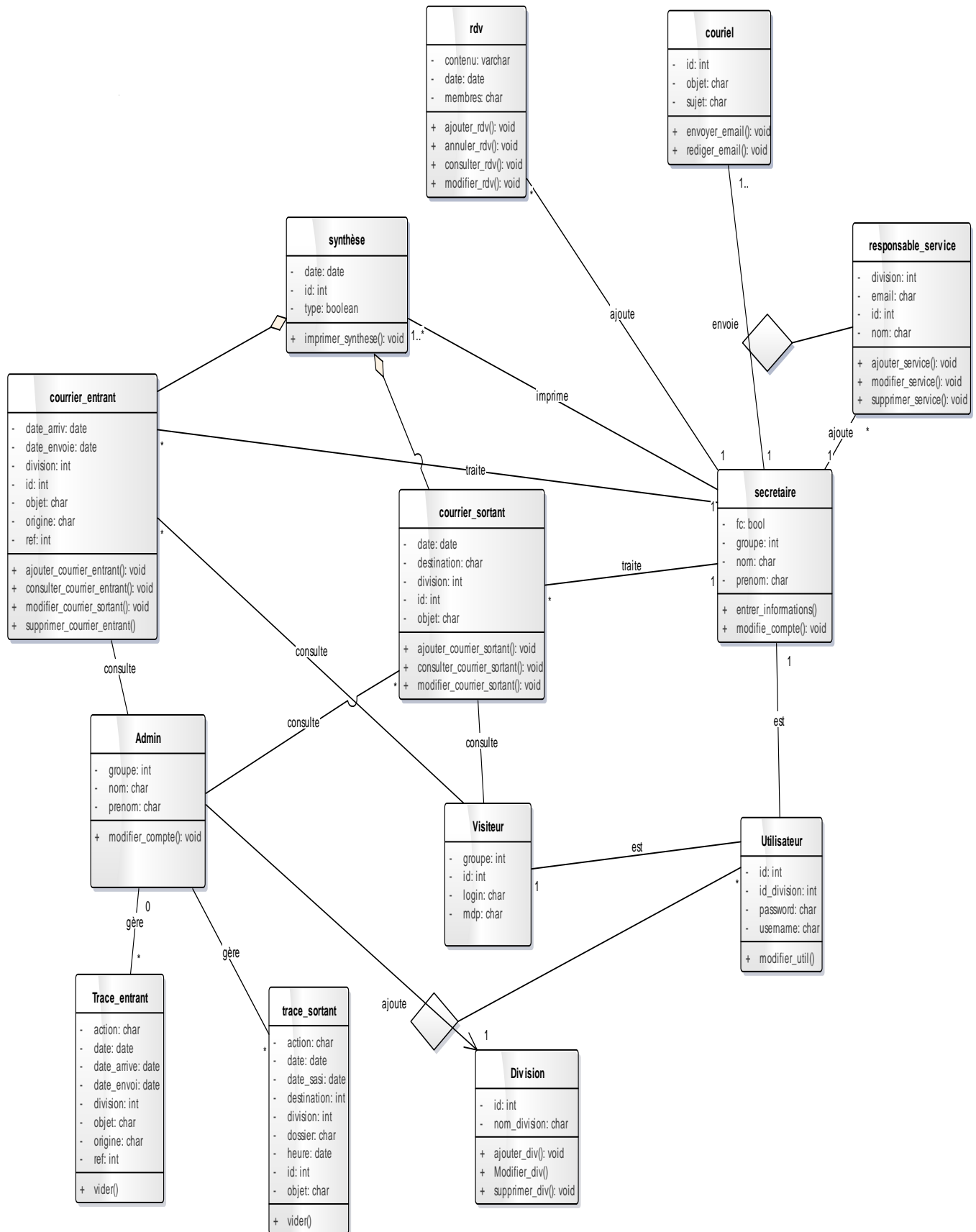


FIGURE 17 : DIAGRAMME DE SEQUENCE/AJOUT DE DIVISION

*NB :

Les diagrammes ci-dessus ne sont que des exemples, non la totalité des diagrammes réalisés dans le projet.

c. Diagramme de classe



Description :

C'est le point central dans le développement orienté objet. Il représente la structure statique du système sous forme de classes et de leurs relations.

Les classes constituent la base pour la génération de code et des schémas de bases de données.

L'application a trois types d'utilisateurs, l'administrateur qui est unique, la secrétaire et le visiteur.

- Chaque utilisateur a une vue propre à lui.
- La secrétaire gère les courriers entrants/sortants, peut ajouter supprimer et modifier selon le besoin, ces actions sont stockés dans une autre table de traçabilité, que seul l'administrateur peut voir ou vider.
- Le dispatching est une opération d'envoi d'email que la secrétaire effectue afin de distribuer une information aux chefs de services de sa division, qu'elle a stocké auparavant.
- La gestion des rendez-vous est l'opération de garder un rappel dans l'application qui aide la secrétaire à se rappeler des événements qu'elle a enregistré.
- La synthèse est l'opération d'imprimer une trace écrite sur papier des tables de courriers entrants/sortants selon le jour.
- L'administrateur est celui qui crée les sessions.
- Chaque division possède une secrétaire et un visiteur propre à elle.
- Lors de la première connexion, la secrétaire doit changer le mot de passe de sa session.
- L'administrateur a la possibilité de changer les mots de passe des utilisateurs en cas d'oubli, ou de problèmes.
- Les vues diffèrent selon le groupe auquel appartient un utilisateur, secrétaire=1 visiteur=2 et admin=0.



Chapitre 4 : Présentation de l'application

Dans ce chapitre nous allons expliquer comment l'application a été réalisée, en mettant à disposition des captures écran ainsi que des figures accompagnées de commentaires et descriptions permettant d'illustrer le travail qui a été accomplis durant la période de stage au sein de la Direction Nationale de la Météorologie



I. Outils de développement

Loin de la faculté et des enseignants, le stage permet à l'étudiant de s'auto-former et forger un nouvel esprit d'analyse qui lui donne l'opportunité de régler les problèmes en toute indépendance, et n'ayant que les connaissances et les techniques d'apprentissage acquises au cours du cursus universitaire comme bagage permettant d'affronter le défis du monde réel dont on a un aperçu à travers le stage.

D'où, le besoin nous poussa à apprendre de nouvelles techniques d'usage du PHP et là il s'agit de **FRAMEWORK** ;

1- Framework CodeIgniter et PHP :



CodeIgniter est un Framework libre écrit en PHP. IL suit le motif de conception MVC. La documentation de CodeIgniter est complète. La communauté du Framework est très active ce qui permet de trouver de l'aide très rapidement. De plus, les membres de la communauté de CodeIgniter ont développé de nombreuses bibliothèques réutilisables.

CodeIgniter facilite beaucoup l'utilisation du PHP grâce à ses bibliothèques, et sa structure favorise la simplicité du code ; c'est-à-dire que les vues, modèles et contrôleurs sont séparés les uns les autres.

La raison pour laquelle nous avons choisi ce framework c'est parce qu'il est le plus adapté au genre de notre application.

- ❖ Une réalisation rapide.
- ❖ Documentation importante.
- ❖ Travail organisé.
- ❖ L'utilisation du PHP.

Ci-dessous on peut trouver la table de matière de notre framework, qui a été très bénéfique durant la réalisation du projet, comme dans le cas de :

- ❖ Formulaires
- ❖ Calendriers
- ❖ Tableaux
- ❖ JavaScript
- ❖ Etc...

<u>Basic Info</u>	<u>General Topics</u>	<u>Class Reference</u>	<u>Driver Reference</u>
<u>Server Requirements</u>	<u>CodeIgniter URLs</u>	<u>Benchmarking Class</u>	<u>Caching Class</u>
<u>License Agreement</u>	<u>Controllers</u>	<u>Calendar Class</u>	<u>Database Class</u>
<u>Change Log</u>	<u>Reserved Names</u>	<u>Cart Class</u>	<u>JavaScript Class</u>
<u>Credits</u>	<u>Views</u>	<u>Config Class</u>	<u>Helper Reference</u>
<u>Installation</u>	<u>Models</u>	<u>Email Class</u>	<u>Array Helper</u>
<u>Downloading CodeIgniter</u>	<u>Helpers</u>	<u>Encryption Class</u>	<u>CAPTCHA Helper</u>
<u>Installation Instructions</u>	<u>Using CodeIgniter Libraries</u>	<u>File Uploading Class</u>	<u>Cookie Helper</u>
<u>Upgrading from a Previous Version</u>	<u>Creating Your Own Libraries</u>	<u>Form Validation Class</u>	<u>Date Helper</u>
<u>Troubleshooting</u>	<u>Using CodeIgniter Drivers</u>	<u>FTP Class</u>	<u>Directory Helper</u>
<u>Introduction</u>	<u>Creating Your Own Drivers</u>	<u>HTML Table Class</u>	<u>Download Helper</u>
<u>Getting Started</u>	<u>Creating Core Classes</u>	<u>Image Manipulation Class</u>	<u>Email Helper</u>
<u>CodeIgniter at a Glance</u>	<u>Hooks - Extending the Core</u>	<u>Input Class</u>	<u>File Helper</u>
<u>CodeIgniter Cheat sheets</u>	<u>Auto-loading Resources</u>	<u>JavaScript Class</u>	<u>Form Helper</u>
<u>Supported Features</u>	<u>Common Functions</u>	<u>Loader Class</u>	<u>HTML Helper</u>
<u>Application Flow Chart</u>	<u>URI Routing</u>	<u>Language Class</u>	<u>Inflector Helper</u>
<u>Model-View-Controller</u>	<u>Error Handling</u>	<u>Migration Class</u>	<u>Language Helper</u>
<u>Architectural Goals</u>	<u>Caching</u>	<u>Output Class</u>	<u>Number Helper</u>
<u>Tutorial</u>	<u>Profiling Your Application</u>	<u>Pagination Class</u>	<u>Path Helper</u>
<u>Introduction</u>	<u>Running via the CLI</u>	<u>Security Class</u>	<u>Security Helper</u>
<u>Static pages</u>	<u>Managing Applications</u>	<u>Session Class</u>	<u>Smiley Helper</u>
<u>News section</u>	<u>Handling Multiple Environments</u>	<u>Trackback Class</u>	<u>String Helper</u>
<u>Create news items</u>	<u>Alternative PHP Syntax</u>	<u>Template Parser Class</u>	<u>Text Helper</u>
<u>Conclusion</u>	<u>Security</u>	<u>Typography Class</u>	<u>Typography Helper</u>
	<u>PHP Style Guide</u>	<u>Unit Testing Class</u>	<u>URL Helper</u>
	<u>Writing Documentation</u>	<u>URI Class</u>	<u>XML Helper</u>
		<u>User Agent Class</u>	
		<u>XML-RPC Class</u>	
		<u>Zip Encoding Class</u>	

FIGURE 19 : TABLE DE MATIERE DE CODEIGNITER

Hypertext Preprocessor, plus connu sous son sigle **PHP** est un langage de programmation libre principalement utilisé pour produire des pages Web dynamiques via un serveur HTTP, mais pouvant également fonctionner comme n'importe quel langage interprété de façon locale. PHP est un langage impératif orienté-objet.

2- WAMP SERVER :



WampServer est une plate-forme de développement Web sous Windows pour des applications Web dynamiques à l'aide du serveur Apache2, du langage de scripts PHP et d'une base de données MySQL.

3- MyPHPAdmin :



C'est une application Web de gestion pour des systèmes de gestion de base de données MySQL réalisée en PHP.

4- Apache :



Apache est un serveur http crée et maintenu au sein de la fondation Apache. C'est le serveur http populaire du World Wide Web. Il est distribué selon les termes de la licence Apache.

5- MySQL :



MySQL est un système de gestion de bases de données relationnelles (SGBDR). Il est distribué sous une double licence GPL et propriétaire. Il fait partie des logiciels de gestion de base de données les plus utilisés au monde¹, autant par le grand public (applications web principalement) que par des professionnels, en concurrence avec Oracle, Informix et Microsoft SQL Server.

Son nom vient du prénom de la fille du Co créateur Michael Widenius, MySQL fait allusion au Structured Query Language, le langage de requête utilisé.

6- DataTable :

C'est un plug-in pour la bibliothèque jQuery. C'est un outil très flexible, basé sur les principes de l'amélioration progressive, et les contrôles d'interactions avancées pour une table HTML. Lorsque vous initialiser l'objet jQuery.dataTable, informations sur la table sont lues directement sur la page HTML. Parmi ses avantages, c'est un logiciel libre (open source), facilite la pagination, utilise une recherche multicritère, supporte plusieurs sources de données : DOM, JavaScript, Ajax et server-side processing.

7- JavaScript :

C'est un langage de programmation de scripts principalement utilisé dans les pages web interactives mais aussi côté serveur. C'est un langage orienté objet à prototype.

8- Ajax :

C'est une architecture informatique basée essentiellement sur le JavaScript, permet d'accéder de manière asynchrone avec les actions de l'utilisateur à la base de données et ce en utilisant en plus du JavaScript, une classe XMLHttpRequest, qui comporte des méthodes permettant de communiquer avec le serveur, offrant ainsi à l'utilisateur une réponse rapide et instantanée.

9- JQuery :

C'est une bibliothèque JavaScript libre qui porte sur l'interaction entre JavaScript (comprenant Ajax) et HTML, et a pour but de simplifier des commandes communes de JavaScript. La première version date de janvier 2006.

10- CSS :



Le concept de feuille de style est présent dès l'origine du World Wide Web : le premier navigateur web (« WorldWideWeb », renommé par la suite « Nexus ») permet de mettre en forme les documents à l'aide de ce qui serait aujourd'hui considéré comme une « feuille de style utilisateur »². De même, les navigateurs Viola en 1992 et Harmony en 1993 recourent à un mécanisme similaire permettant de déterminer le rendu des polices de caractères, des couleurs ou de l'alignement du texte.

II. Aperçu de l'application

1- L'authentification :

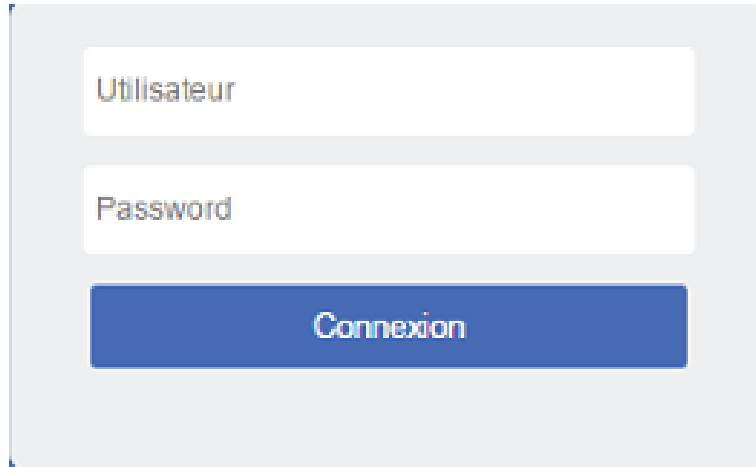


FIGURE 20: FENETRE D'AUTHENTIFICATION

Cette fenêtre permet l'authentification des acteurs, si le mot de passe est correct l'application les redirige vers leurs vues associées. Sinon recharge la page d'authentification.

2- Informations personnelles:

Si la secrétaire se connecte pour la première fois, l'application lui demande d'entrer ses informations personnelles et un nouveau mot de passe de session :

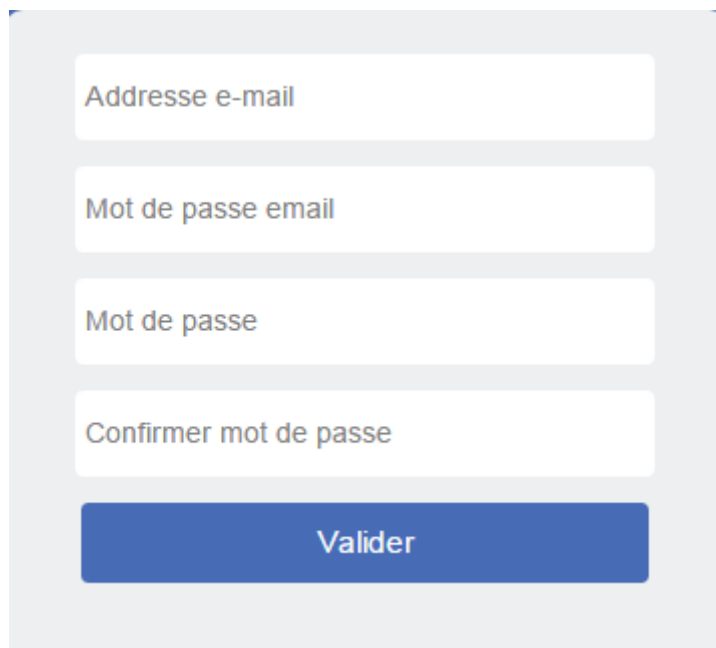


FIGURE 21: FENETRE D'INFORMATIONS PERSONNELLES

3- Page d'accueil Administrateur :

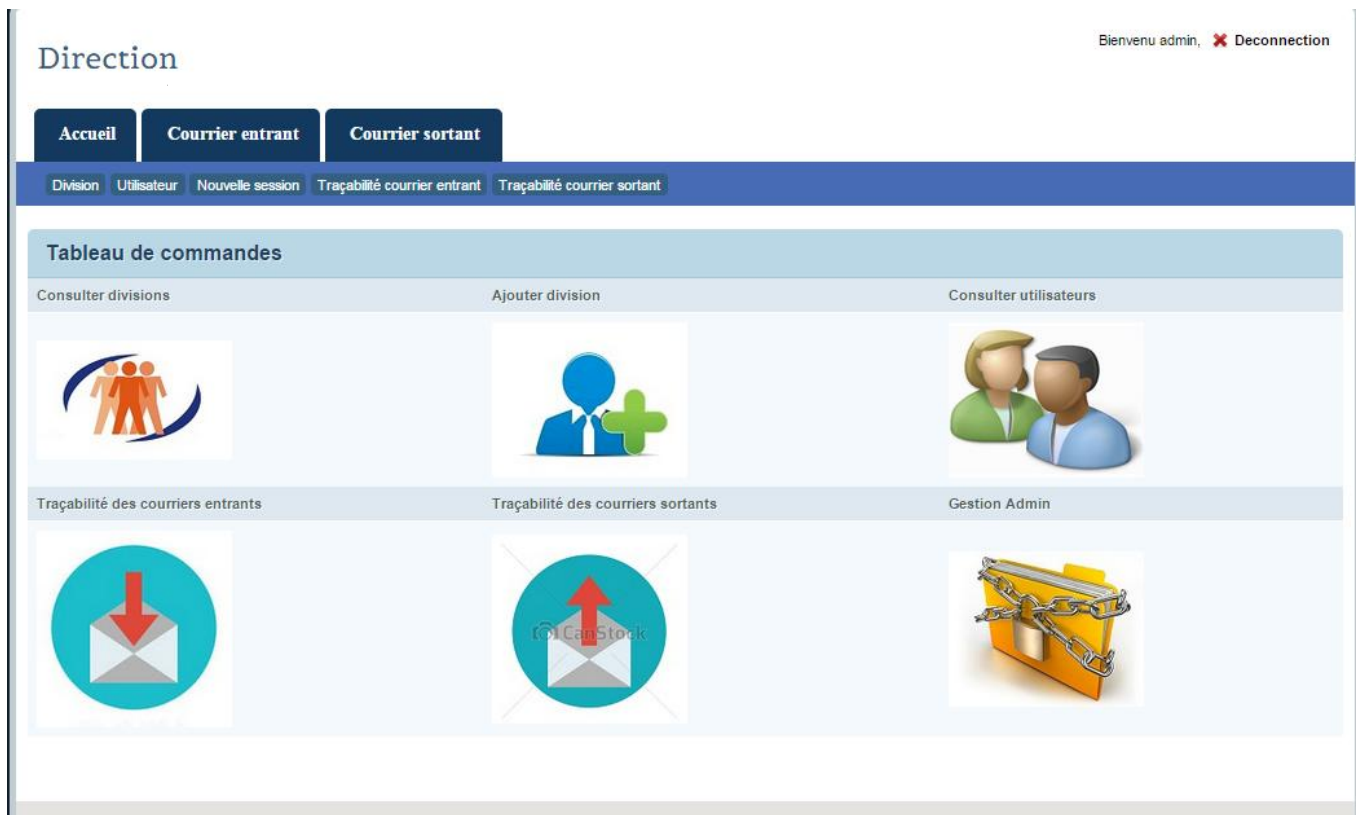


FIGURE 22 : PAGE D'ACCUEIL ADMIN

Comporte les fonctionnalités essentielles de l'administrateur.

4- Les divisions (ADMIN):

Liste des divisions		
Voir 10 entrées	Chercher:	
ID	Nom de division	Actions
1	Systeme information	 
15	Direction technique	 
ID	Nom de division	Actions
Les entrées de 1 a 2 of 2		Precedent 1 Suivant

FIGURE 23 : LISTE DES DIVISIONS

La liste des divisions de l'application, on peut supprimer et modifier le nom de la division.

5- Les utilisateurs (ADMIN)::





Utilisateurs de l'application				
Voir	10	entrées	Chercher:	
ID	Nom d'utilisateur	Groupe	Division	Reinitialiser le mot de passe
2	secretaire	secretaire	Systeme information	
3	visiteur	visiteur	Systeme information	
24	secretaire1	secretaire	Direction technique	
25	visiteur1	visiteur	Direction technique	
ID	Nom d'utilisateur	Groupe	Division	Reinitialiser le mot de passe
Les entrées de 1 a 4 of 4				
		Précédent	1	Suivant

FIGURE 23 : LISTE DES D'UTILISATEURS

Listes des utilisateurs de l'application, on peut modifier le mot de passe des utilisateurs en cas de problèmes et urgences.

6- La traçabilité (ADMIN):

Trace courrier entrant

Voir10entrées

Chercher:

ID	Date d'envoi	Date de reception	Reference	Origine	Objet	Action	Date d'édition	Heure	Division
20151	2015-06-02	2015-06-04	1	test test	ett4	supprimer	2015-06-16	11:16:34	Systeme information
20151	2015-06-01	2015-06-10	1	CCC	AAA	inserer	2015-06-16	11:19:38	Systeme information
20152	2015-06-11	2015-06-02	2	TEST	TEST	inserer	2015-06-16	11:20:04	Systeme information
20154	2015-06-19	2015-03-17	666666	ekjgb	kdbgk	maj	2015-06-02	19:58:20	Systeme information
20154	2015-06-19	2015-03-17	666666	ekjgb	kdbgk	supprimer	2015-06-16	11:17:00	Systeme information
20155	2015-09-09	2015-03-17	121212	nzdbf	skjgn	inserer	2015-06-02	20:06:20	Systeme information
20156	2015-06-03	2015-04-15	999999	azeezar	hklhg	inserer	2015-06-02	20:35:55	Systeme information
20156	2015-06-03	2015-04-15	999999	azeezar	hklhg	supprimer	2015-06-16	11:18:59	Systeme information
20157	2015-03-17	2015-07-10	4646	kjdbf	kjsdgkh	inserer	2015-06-02	20:38:31	Systeme information
20157	2015-03-17	2015-07-10	4646	kjdbf	kjsdgkh	supprimer	2015-06-16	11:19:04	Systeme information
ID	Date d'envoi	Date de reception	Reference	Origine	Objet	Action	Date d'édition	Heure	Division

Les entrées de 1 a 10 of 12

Précédent

1

2

Suivant

Vider

FIGURE 24 : TABLEAU DE TRAÇABILITE DES ENTRANTS

C'est le tableau montre ce que chaque secrétaire effectue comme action sur les tables des courriers, on peut vider ce tableau en un simple clic.

7- Ajouter une division (ADMIN):

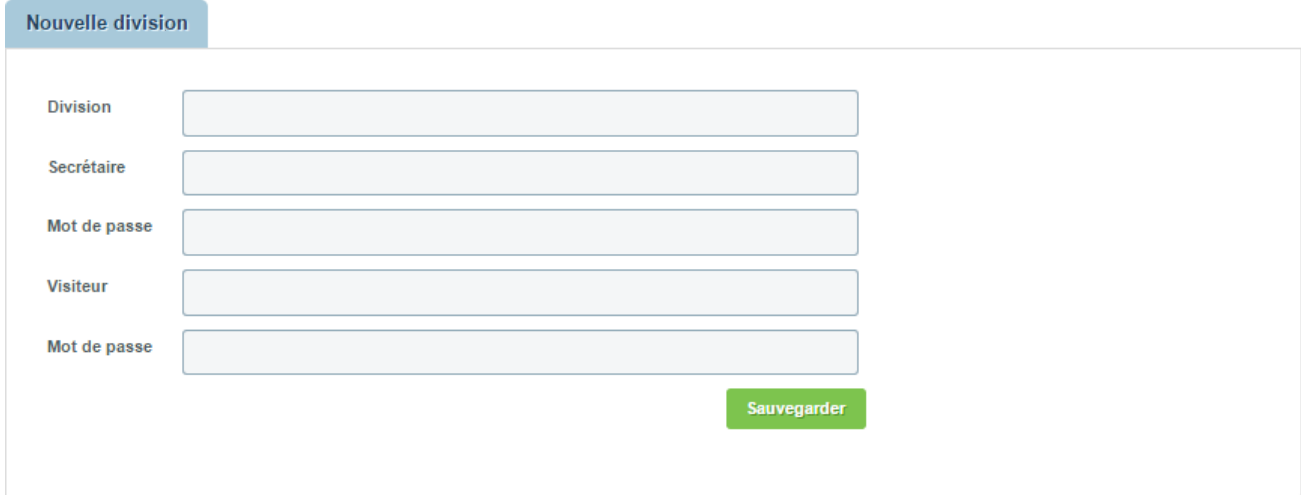


FIGURE 24 : AJOUT DE DIVISION

Ajouter une nouvelle division, avec sa secrétaire et son visiteur.

8- Page d'accueil de la secrétaire :

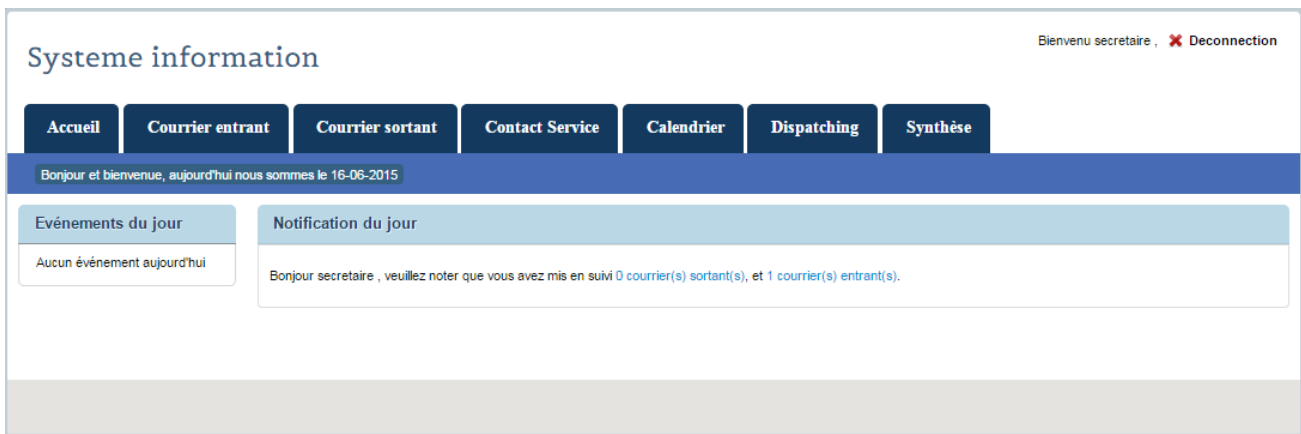


FIGURE 25 : PAGE D'ACCUEIL SECRETAIRE

C'est la page d'accueil qui montre ce que la secrétaire a mis en suivi, et les événements du jour à partir du calendrier. En cliquant sur suivi, le système la redirige vers le tableau des courriers mis en suivi.

9- Table des courriers (secrétaire) :

Liste des courriers entrants								
Voir	10	entrées	Chercher: <input type="text"/>					
ID	Date d'envoi	Date de reception	Reference	Origine	Objet	Date d'édition	Suivi	Action
20151	2015-06-01	2015-06-10	1	CCC	AAA	2015-06-16		
20152	2015-06-11	2015-06-02	2	TEST	TEST	2015-06-16		
ID	Date d'envoi	Date de reception	Reference	Origine	Objet	Date d'édition	Suivi	Action
Les entrées de 1 a 2 of 2							Precedent	1 Suivant
Ajouter								

FIGURE 26 : TABLEAU DES COURRIERS ENTRANTS

C'est l'interface qui lui montre sa base de données, lui donne la main pour faire sa gestion, comme le suivi les actions...

10- Ajouter un courrier (secrétaire) :

Nouveau courrier entrant

Date d'envoi

Date d'arrivée

Référence

Origine

Objet

June 2015

Su	Mo	Tu	We	Th	Fr	Sa
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30				

Sauvegarder

FIGURE 26 : AJOUTER DES COURRIERS ENTRANTS

Donne la possibilité d'ajouter un courrier entrant, à l'aide d'un « datepicker » pour faciliter le stockage.

11- Synthèse (secrétaire) :

Synthèse

Type de courrier

Courrier entrant

Date

June 2015

Afficher

Su	Mo	Tu	We	Th	Fr	Sa
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30				

FIGURE 27 : SYNTHÈSE

Te donne le choix de la date, afin d’avoir une trace écrite des courriers selon la table.

12- Envoyer un e-mail (secrétaire) :

Composez votre e-mail

Destinataire

Sujet

Corps

Envoyer

FIGURE 28 : ENVOIE D’EMAIL

Cette interface permet à la secrétaire d’envoyer un e-mail de dispatching à travers son propre e-mail aux contacts qu’elle a enregistré auparavant sur sa base de données.

13- Calendrier d’événements (secrétaire) :

<<Prec Juin 2015 Suiv>>

Détails des événements du 2 Juin 2015

LUN	MAR	MER	JEU	VEN	SAM	DIM
1	2 1	3	4	5	6	7
8	9	10 1	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30					



17:00:00

Réunion salle 2



+ Ajouter evenement

FIGURE 29 : CALENDRIER

Ce calendrier permet d'enregistrer les événements selon le jour, ces événements sont stockés sur la base de données, et affichés dans le « sidebar » de la session de la secrétaire, pour les rappels.

Conclusion

Ayant effectué le stage au sein de la Direction nationale de la météorologie, le but principal était de satisfaire le maximum de ce qui est demandé dans le cahier de charges.

Il nous a été demandé de mettre en œuvre une application pour gérer les courriers entrants et sortants tout en utilisant un le framework CodeIgniter qui marche en PHP ainsi que la modélisation UML.

Le travail s'est déroulé en trois majeures parties, la première c'était l'étude des nouvelles méthodes, la deuxième comporta l'analyse fonctionnelle quant à la troisième partie elle cerna le codage et la mise en œuvre du projet.

La création de l'application commença par l'instauration de la base de données, fondation du dispositif mis en place, cette base de données devait être capable de gérer les traces écrites des courrier entrant et sortant, la prise de rendez-vous, les traces des actions, les contacts ainsi que la gestion des utilisateurs, et cela en distinguant les élément propre à chaque utilisateur et les utilisateurs appartenant eux même à des divisions et séparés en trois groupes (secrétaire, visiteur et admin). L'étape suivante fut la mise en place d'interfaces basées sur le framework, CodeIgniter, ces interfaces ont été créées pour chaque groupe d'utilisateur précédemment citées.

La direction nationale de la météorologie a bien des atouts, étant le responsable à l'échelon national dans son domaine, étant représentant de l'autorité Marocaine, ils sont un modèle de stabilité, fiabilité, efficacité et organisation.

Ayant effectuée mon stage dans ce cadre, ces valeurs m'ont été transmises et cette période passée au sein de cette organisme m'ont permis de progresser.

Cette formation en alternance m'a été très bénéfique puisqu'elle m'a apporté autant de connaissance théorique que pratique, en effet j'ai eu l'occasion d'appliquer sur le terrain ce que j'ai appris. J'ai également appris à m'adapter au monde du travail.

Notre travail s'est fixé comme objectifs de satisfaire le maximum des besoins du cahier de charge et faciliter les tâches à la secrétaire en premier temps, puis l'administrateur pour assurer la bonne continuation du travail. Ainsi je suis arrivée à achever 98% du travail demandé.

Perspectives :

Vu la courte durée du stage, il nous a été impossible de mettre l'application en service et la tester en temps réel. De là la prochaine étape et mettre en service l'application au niveau du serveur interne de la Direction Nationale de la météorologie, faire une petite formation sur les différentes fonctionnalités de l'application et par la suite faire un suivi afin d'adapter cette dernière aux besoins réels ainsi que corriger les erreurs pouvant survenir au fil de l'utilisation.

L'une des perspectives futures de notre projet est la possibilité d'archiver les différentes données stockées selon les critères de dates ou de division. A cela s'ajoute une nécessité de trouver une adaptation entre l'envoi d'email via CodeIgniter et les différents services d'emails autres que Gmail.

Webographie et bibliographie :

ellislab.com/CodeIgniter

<http://fr.wikipedia.org/wiki/JQuery>

<http://fr.wikipedia.org/wiki/CodeIgniter>

<http://fr.wikipedia.org/wiki/Modèle-vue-contrôleur>

[http://fr.wikipedia.org/wiki/Apache HTTP Server](http://fr.wikipedia.org/wiki/Apache_HTTP_Server)

<http://fr.wikipedia.org/wiki/Aptana>

[http://fr.wikipedia.org/wiki/MySQL Workbench](http://fr.wikipedia.org/wiki/MySQL_Workbench)

<http://fr.wikipedia.org/wiki/PHP>

jquery.com

api.jquery.com/jquery.ajax/

[Cours Ilham Chakir / génie logiciel.](#)

[Cours BENABBOU Abderrahim/ UML](#)

[Cours BEGDOURI Ahlam/ WEB](#)