

**UNIVERSITE SIDI MOHAMED BEN ABDELLAH
FACULTE DES SCIENCES ET TECHNIQUES FES
DEPARTEMENT D'INFORMATIQUE**



Projet de Fin d'Etudes

Licence Sciences et Techniques Génie Informatique

Application gestion de congé

WINCOR
NIXDORF

Lieu de stage :Wincor Nixdorf

Réalisé par :

-Zehouani Fatma

Encadré par :

-Pr.Said Najah

-Mr.Mohamed Douayou

Soutenu le 15/06/2012 devant le jury composé de :

**Pr.Rachid Ben Abbou (Président)
Pr.Khalid Zenkouar (Examinateur)
Pr. Said Najah (Encadrant FST)**

Année Universitaire 2011-2012



Dédicace

Je dédie ce travail

A mes chers parents

Pour toutes les peines endurées, toutes les privations, sacrifices consentis et l'assistance durant toute ma vie estudiantine, pour faire de moi la femme de demain.

A ma sœur et mon beau frère Ghita et Yassine

Pour tant de confiance, d'amour, de patience et d'abnégation

A mes cousins et cousines, oncles et tantes

Pour tant de consolation et de tendresse

A mes meilleurs amies Salma et Zainab

Pour m'avoir comblé de joie et d'assistance durant toute la période de mon stage.



Remerciement

Dans le cadre de la réalisation de notre travail effectué, je tiens vivement à remercier tous ceux qui ont été à l'origine de mon aide et assistance et tous ceux qui ont travaillé derrière les coulisses pour m'aider à effectuer notre travail.

Je pense d'abord aux professeurs de la faculté des sciences et techniques Fès, et plus particulièrement ceux du département du Génie Informatique, artisans de notre formation universitaire.

Je remercie particulièrement *Mr Said Najah* qui malgré ses grandes responsabilités m'a consacré du temps pour m'encadrer.

Je tiens à exprimer ma gratitude à *Mr Simohamed Douayou* qui a bien voulu superviser mon travail. Je les remercie infiniment pour leur patience et leurs précieux conseils.

Je tiens également à remercier les professeurs qui ont eu l'amabilité d'accepter de siéger dans le jury de soutenance et de me faire part de leurs remarques et leurs propositions.

Je voudrai aussi exprimer mes remerciements à tous ceux qui m'ont aidé de près ou de loin à réaliser ce travail.



Sommaire

<i>Introduction générale</i> :.....	7
<i>Chapitre 1:Présentation de l'entreprise</i>	
I. Wincor Nixdorf au niveau international :.....	9
II. Wincor Nixdorf au niveau national :.....	10
Présentation :	10
Fiche technique :	12
Historique :	13
L'organigramme de l'entreprise:.....	14
Les services de la société :.....	15
<i>Chapitre 2:Etude et spécification des besoins</i>	
I. Introduction.....	17
II. Buts et objectifs :.....	17
III. Etude préliminaire :	17
1. Fonctionnalités:.....	17
2.Schéma descriptif de l'application :	18
3.Choix technique :.....	19
<i>Chapitre 3:Modélisation et conception</i>	
I. Introduction.....	25
II. Processus de développement en Y.....	25
III. Diagramme des cas d'utilisation :	27
IV. Modèle statique	31
V. Modèle dynamique.....	35
VI. Diagramme de classe.....	43
<i>Chapitre 4:Réalisation</i>	



I. Introduction.....	46
II. Interface d'application.....	46
<i>Concluions et perspectives.....</i>	<i>50</i>

Liste des figures

<i>Figure 1 : Diagramme de cas d'utilisation pour l'acteur salarié.....</i>	<i>29</i>
<i>Figure 2 : Diagramme de cas d'utilisation pour l'acteur responsable.....</i>	<i>30</i>
<i>Figure 3 : Diagramme de cas d'utilisation pour l'acteur direction R.H.....</i>	<i>31</i>
<i>Figure 4: Diagramme de classe préliminaire pour le cas d'authentification.....</i>	<i>32</i>
<i>Figure 5: Diagramme de classe préliminaire pour le cas de saisie de demande de congé.....</i>	<i>32</i>
<i>Figure 6: Diagramme de classe préliminaire pour le cas de consultation d'état</i>	<i>33</i>
<i>Figure 7: Diagramme de classe préliminaire pour le cas d'envoi de demande</i>	<i>33</i>
<i>Figure 8: Diagramme de classe préliminaire pour le cas de consultation des demandes.....</i>	<i>34</i>
<i>Figure 9: Diagramme de classe préliminaire pour le cas de validation des demandes.....</i>	<i>34</i>
<i>Figure 10: Diagramme de classe préliminaire pour le cas de consultation des listes des employés.....</i>	<i>35</i>
<i>Figure 11: Diagramme de séquence pour l'étape d'authentification.....</i>	<i>36</i>
<i>Figure 12: Diagramme de séquence pour l'étape de saisie des demande</i>	<i>37</i>
<i>Figure 13: Diagramme de séquence pour la consultation d'états de congé</i>	<i>38</i>
<i>Figure 14: Diagramme de séquence pour l'étape d'envoi de demande de congé</i>	<i>39</i>
<i>Figure 15: Diagramme de séquence pour de consultation des demandes.....</i>	<i>40</i>
<i>Figure 16: Diagramme de séquence pour l'envoi de réponse de demande</i>	<i>41</i>
<i>Figure 17: Diagramme de séquence pour l'étape de consultation des employés</i>	<i>42</i>



Introduction générale :

Le Maroc est le troisième pays en Afrique, après l'Afrique du Sud et l'Égypte, dans le domaine des technologies de l'information, en termes de flux économique drainé par le secteur. Ce constat découle de la croissance soutenue réalisée par les sociétés opérant sur ce segment d'activité à fort potentiel.

Les implantations d'entreprise de services informatiques se sont rapidement multipliées. Actuellement, le nombre des sociétés spécialisées au Maroc s'élève à plus de 1.000 entreprises, soit une croissance de l'ordre de 20% en l'espace de cinq ans.

La majorité de ces entreprises sont concentrées auprès de leurs principaux donneurs d'ordre. D'ailleurs, Casablanca, à elle seule, concentre près de 65% de l'offre tandis que Rabat en polarise 20%.

Vu la croissance exponentielle de secteur d'informatique, la concurrence des sociétés intéressées devient de plus en plus rude et acharnée. Dans ce contexte Wincor Nixdorf a la part de lion vu l'investissement et la capacité professionnelle de son personnel qui ne cesse de s'intéresser à la formation continue, pour être en mesure de s'adapter à toutes les innovations.

C'est dans l'optique de mieux cerner le marché d'informatique que j'ai choisi de passer mon stage de fin d'études dans une société de ce domaine. Il s'agit de Wincor Nixdorf qui m'a ouvert ses portes pour réaliser mon projet de fin d'études qui a duré deux mois j'avais comme tâche la réalisation d'une application de gestion de congé pour simplifier la saisie et la validation des demandes

Dans ce rapport, je présenterai en détail les étapes que j'ai suivies pour réaliser l'application. Ce rapport comporte quatre chapitres. Le premier chapitre contiendra la présentation de l'entreprise, le deuxième chapitre je l'ai consacré à l'étude et spécification des besoins, le troisième chapitre à la modélisation et conception, et le dernier chapitre à la réalisation



Chapitre I :

Présentation de l'entreprise

I. Wincor Nixdorf au niveau international :



Wincor Nixdorf est le spécialiste des métiers de la banque et de la distribution depuis plus de 30 ans. Il est ainsi créateur de solutions de services pour les points de vente et le libre-service.

Wincor Nixdorf compte 7300 employés dans le monde pour un chiffre d'affaires de 1,74 milliards d'euros. Il est présent dans plus de 90 pays avec des filiales dans 32 d'entre eux.

Wincor Nixdorf est l'une des rares entreprises à proposer une gamme complète et spécialisée de solutions et de services pour ces secteurs d'activité.

Ses usines sont basées à Paderborn (Allemagne), Singapour, Sao Paulo (brésil).



II. Wincor Nixdorf au niveau national :

Présentation :

Au Maroc, forte de la confiance de ses clients, Wincor Nixdorf a poursuivi sa politique d'investissement à long terme dans les secteurs de la banque de la distribution, de la poste et des grands comptes

S'appuyant sur sa vision à long terme, Wincor Nixdorf a développé depuis une vingtaine d'années, une stratégie clef pour les différents marchés qu'elle adresse :

➤ La banque :

Wincor Nixdorf, leader dans ce secteur, renforce son engagement à long terme envers ses clients et les accompagne dans les défis auxquels ils sont confrontés : optimisation des coûts, amélioration de la qualité de service....

➤ La distribution :

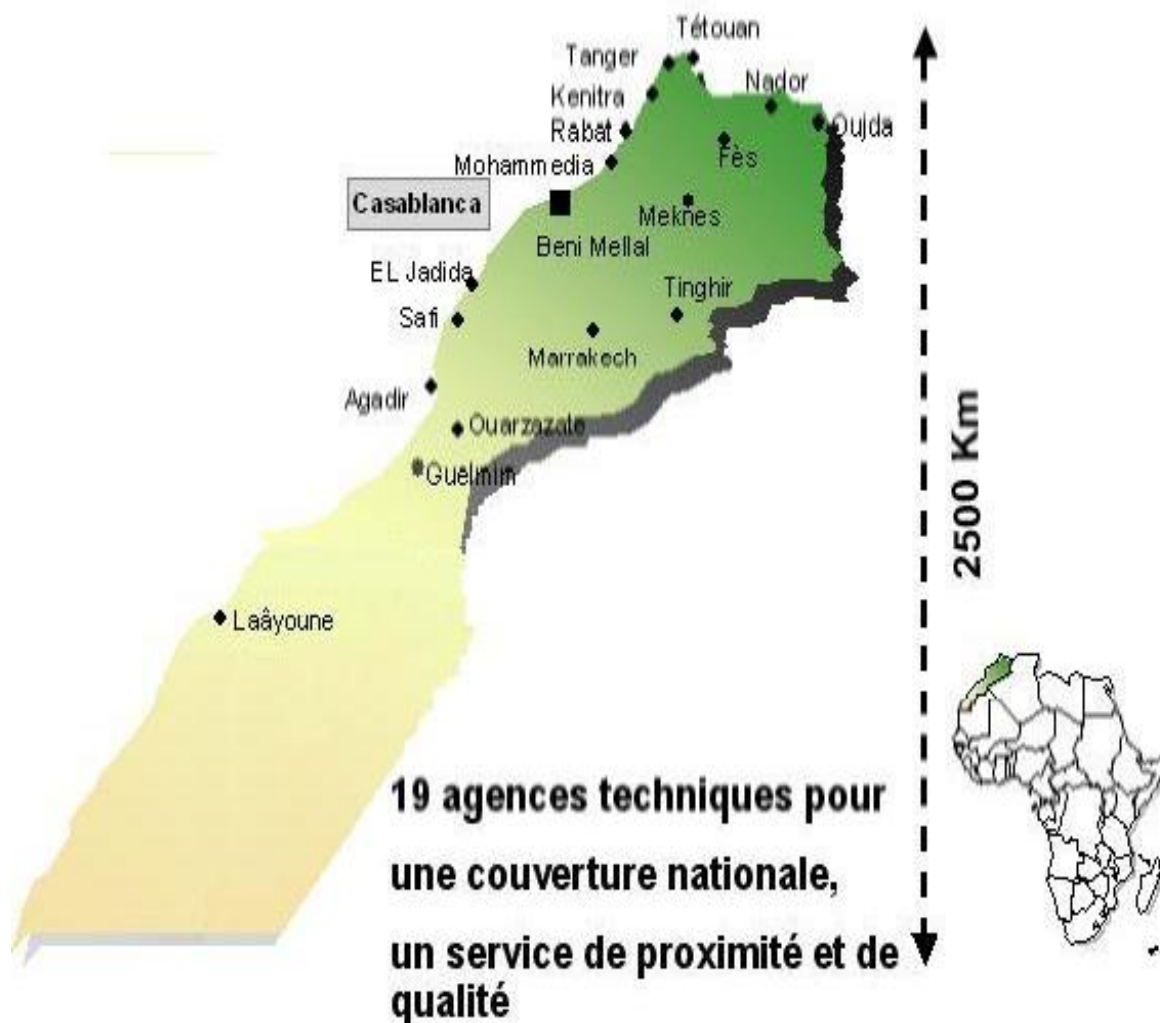
Conforme à sa stratégie pragmatique et bénéficiant du leadership en Europe, Wincor Nixdorf capitalise sur l'expertise internationale et sa stratégie de proximité au Maroc pour adresser ce marché en pleine expansion

➤ Les grands comptes :

Le service public, les compagnies de transport (aérien, ferroviaire), les offres et les opérations télécoms mettent en place de plus en plus performants pour se rapprocher de leurs clients et leur offrir une meilleure qualité de service. Wincor Nixdorf les accompagne dans leur stratégie en leur proposant des produits et solutions adaptés à leur métier.

➤ **Service de proximité :**

Wincor Nixdorf continue sa croissance dans les centres régionaux de maintenance et compte désormais 20 agences dans les grandes villes mais dans les villes comme Laayoune, Ouarzazate, Guelmim, Errachidia , Nador ...





Fiche technique :

- Raison social : Wincor Nixdor
- Date de création : 02/09/1999.
- Expérience : 20 ans d'expérience dans le domaine de technologie de l'information.
- Capital social : 15 millions de DH
- Chiffre d'affaires : 371 millions de DH
- Employés : 100 dont 76 dans la Division Services.
- Siège : Casablanca.
- Centres régionaux de maintenance : 25
- Conseil d'administration :
 - Said Wahbi : Directeur Général.
 - Abdelghani Mounib : Directeur administratif et financier.
 - Eckard Heidloff : President Wincor Nixdorf Group
 - Yassir Abdelilah : Directeur division banking



Historique :

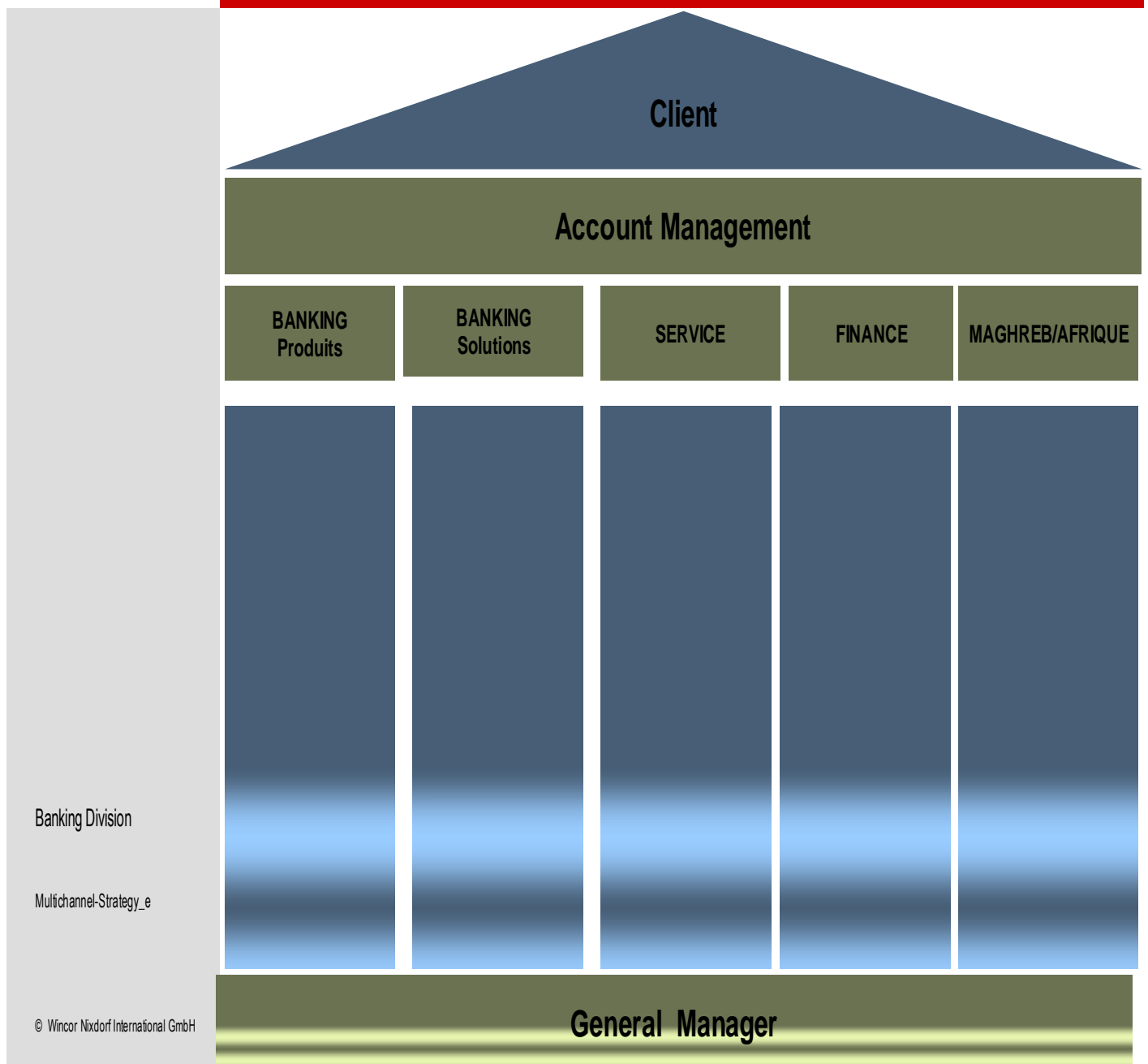
- 1952** Heinz Nixdorf a fondé sa société et l'a amené à devenir, au cours des décennies des suivantes, le quatrième groupe européen.
- 1990** 4 ans après la mort de Heinz Nixdorf computer a été reprise par Siemens donnant naissance à Siemens Nixdorf. Les activités distribution et banque étaient intégrées dans cette organisation.
- 1992-1994** La restructuration et la rationalisation ont été payantes avec un résultat d'exploitation positif.
- 1995-1996** Premières grandes vagues d'expansion. La gamme des produits et services a été étendue et l'internationalisation du groupe est allé de l'avant.
- 1996-1997** Grandes périodes de croissance de diversification.
- 1998** Siemens Nixdorf était créée en tant qu'entité indépendante. Le résultat a été une plus grande expansion.
- 1999** Wincor Nixdorf est sorti du groupe Siemens le 1^{er} octobre 1999 une grande organisation complètement indépendante a été créée avec un grand nombre de filiales dans le monde.
- 2004** Intégration de Nixdorf système bancaire dans le groupe Wincor qui devient Wincor Nixdorf système bancaires.
- 2006** La société continue sa croissance.



L'organigramme de l'entreprise:

Wincor-Nixdorf Maroc est organisée selon l'organigramme suivant :

Wincor Nixdorf Maroc: Organigramme





Les services de la société :

Wincor Nixdorf se compose de plusieurs services homogènes, formant une entité cohérente visant le bien de la société, sa survie, son développement et sa croissance.

Le service juridique : composé d'un responsable, qui a pour mission la prise en charge des actes et formalités diverses.

Le service ressources humaines : composé d'un responsable ressources humaines qui s'occupe du service paie, le recrutement, etc.

Le service financier : composé d'un directeur administratif et financier, d'un responsable du service, de deux comptables qui ont pour mission d'assurer la tenue de la comptabilité, les déclarations fiscales et sociales de la société et tous les travaux de comptabilisation des dossiers entrés et sorties.

Le service commercial : c'est un service qui joue le rôle primordial dans l'organisation de la société, il est composé d'un manager et deux supports commerciaux qui réunissent leurs efforts à la prospection des clients et le développement du chiffre d'affaires, et cela à partir du traitement des appels d'offres en se référant à la mise en œuvre de nouvelles offres de service conformément aux évolutions du marché.

La division Banking : spécialisé dans les solutions et services monétiques des institutions financières, ce service se compose de plusieurs compétences qui viellent sur la réussite des projets banking.

La division service : spécialisé dans la bonne marche des projets des grandes et moyennes surfaces et hôtellerie, il est composé d'un manager et d'un directeur contrôleur budgétaire. Ce service reflète l'image de la société et la qualité du service vis-à-vis du client.



Chapitre II :

Etude et spécification

des besoins

I. Introduction

Wincor Nixdorf souhaite se doter d'un système informatique a fin de permettre aux personnels de gérer leurs demandes de congés à partir d'une interface web et de l'envoyer au responsable qui peut valider ou refuser la demande.

II. But et objectif :

L'objectif principal de l'application est la dématérialisation des demandes de congé.

L'application devra notamment permettre la saisie des demandes et leurs validations par la direction.

Ceci est fait en suivant les étapes suivantes :

- Analyse des besoins
- Elaboration d'un scénario maquette
- Conception et implémentation de la base de données
- Développement de l'application.
- Validation de l'application par des scénarios de tests d'exécution.

III. Etude préliminaire :

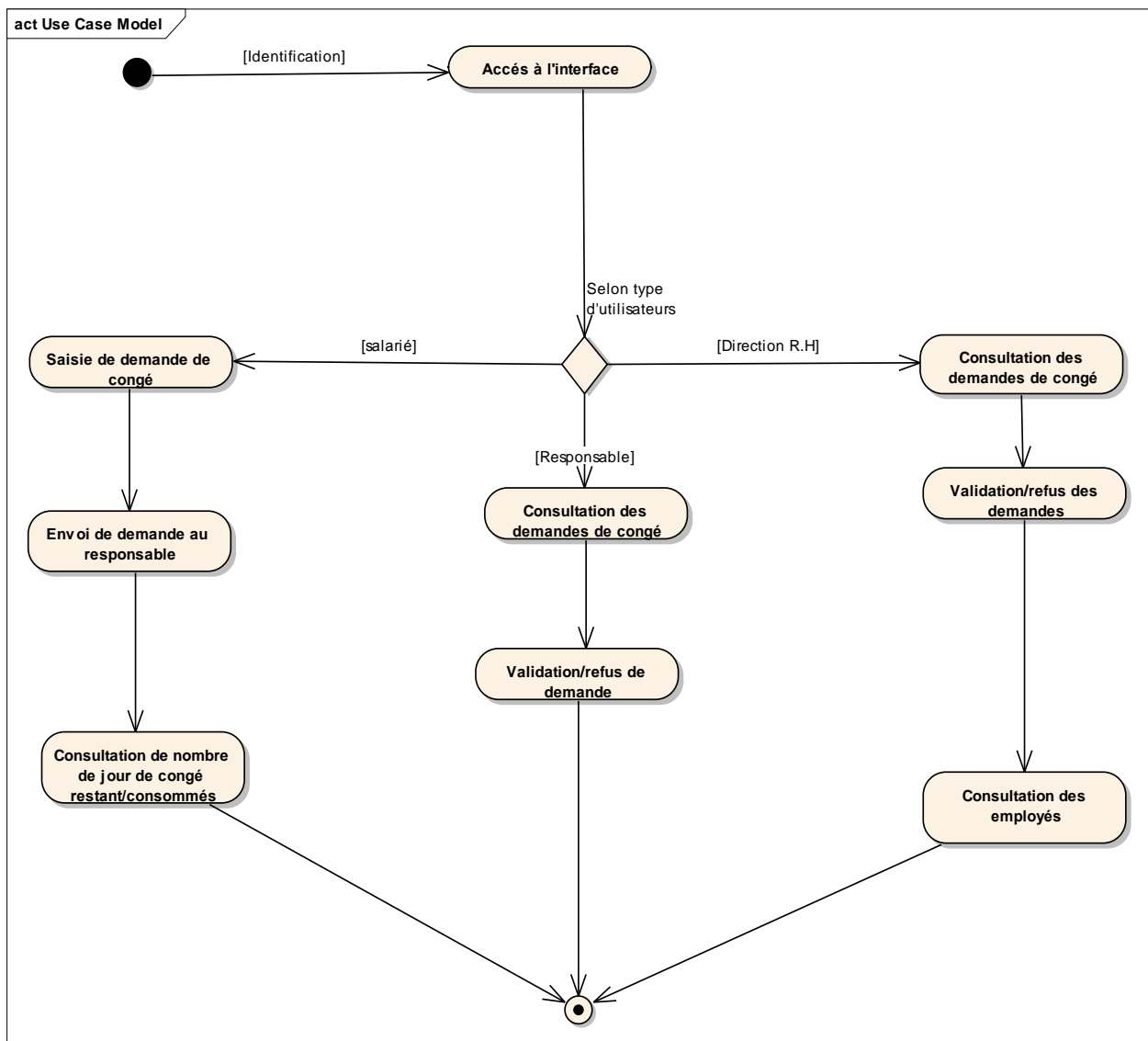
1. Fonctionnalités:

Il s'agit des fonctionnalités du système, ce sont les besoins spécifiant un comportement d'entrée/sortie de l'application. Le système a pour but :

- ✓ Dématérialisation des congés.
- ✓ Optimisation de suivis et validation des congés
- ✓ Simplification de saisie et procès de validation.
- ✓ Paramétrage et administration système.

2. Schéma descriptif de l'application :

Après avoir analysé tout les besoins fonctionnels on peut schématiser notre application par le schéma suivant :



3. Choix technique :

i. Plateforme J2EE :

On parle généralement de «plate-forme J2EE» pour désigner l'ensemble constitué des services (API) offerts et de l'infrastructure d'exécution.

J2EE comprend notamment :

Les spécifications du **serveur d'application**, c'est-à-dire de l'environnement d'exécution : J2EE définit finement les rôles et les interfaces pour les applications ainsi que l'environnement dans lequel elles seront exécutées. Ces recommandations permettent ainsi à des entreprises tierces de développer des serveurs d'application conformes aux spécifications ainsi définies, sans avoir à redévelopper les principaux services.

Des services, au travers d'**API**, c'est-à-dire des extensions Java indépendantes permettant d'offrir en standard un certain nombre de fonctionnalités. *Sun* fournit une implémentation minimale de ces API appelée **J2EE SDK** (*J2EE Software Development Kit*).

Dans la mesure où J2EE s'appuie entièrement sur le Java, il bénéficie des avantages et inconvénients de ce langage, en particulier une bonne portabilité et une maintenabilité du code.

De plus, l'architecture J2EE repose sur des composants distincts, interchangeable et distribués, ce qui signifie notamment :

- ✓ qu'il est simple d'étendre l'architecture ;
- ✓ qu'un système reposant sur J2EE peut posséder des mécanismes de haute-disponibilité, afin de garantir une bonne qualité de service ;
- ✓ que la maintenabilité des applications est facilitée.

Ainsi que l'architecture J2EE permet ainsi de séparer la couche présentation, correspondant à l'interface homme-machine (IHM), la couche métier contenant l'essentiel des traitements de données en se basant dans la mesure du possible sur des API existantes, et enfin la couche de données correspondant aux informations de l'entreprise stockées dans des fichiers, dans des bases de données relationnelles ou XML, dans des annuaires d'entreprise ou encore dans des systèmes d'information complexes

ii. Grails :

Grails est un framework open source de développement agile d'applications web basé sur le langage Groovy et sur le patron de conception Modèle-Vue-Contrôleur .

Grails est effectivement un conteneur dit « léger », c'est-à-dire une infrastructure similaire à un serveur d'application J2EE. Il prend donc en charge la création d'objets et la mise en relation d'objets par l'intermédiaire d'un fichier de configuration qui décrit les objets à fabriquer et les relations de dépendances entre ces objets. Le gros avantage par rapport aux serveurs d'application est qu'avec Grails, vos classes n'ont pas besoin d'implémenter une quelconque interface pour être prises en charge par le Framework (au contraire des serveurs d'applications J2EE). C'est en ce sens que Grails est qualifié de conteneur « léger ». Outre cette espèce de super fabrique d'objets, Grails propose tout un ensemble d'abstractions permettant de gérer entre autres : Pour réaliser tout ceci, Grails s'appuie sur les principes du design pattern et sur la programmation par aspects.

ii. Langage groovy :

Groovy est le nom d'un langage de programmation orienté objet destiné à la plate-forme Java. Il constitue une alternative au langage Java pour cette plate-forme et est inspiré de Python, Ruby. Il est l'objet de la spécification JSR 241.

Groovy utilise une syntaxe très proche de Java, avec des accolades, et est directement compilé, soit à la volée dynamiquement, soit classiquement avec un compilateur en bytecode.

Groovy s'intègre et est entièrement compatible avec la JVM étant donné que le bytecode est le même. Il peut donc

- utiliser les bibliothèques Java,
- être utilisé dans des classes Java.

Le modèle de conception le plus répandu à cette architecture c'est : MVC (modèle, vue, contrôleur).

iii. Modèle MVC :



Le **Modèle-Vue-Contrôleur** est une architecture et une méthode de conception qui organise l'interface homme-machine (IHM) d'une application logicielle. Ce paradigme divise l'IHM en un modèle (modèle de données), une vue (présentation, interface utilisateur) et un contrôleur (logique de contrôle, gestion des événements), chacun ayant un rôle précis dans l'interface, ce qui donne trois parties fondamentales dans l'application finale : le modèle, la vue et le contrôleur.

- **Modèle** : Le modèle représente le comportement de l'application : traitements des données, interactions avec la base de données, etc. Il décrit ou contient les données manipulées par l'application. Il assure la gestion de ces données et garantit leur intégrité. Dans le cas typique d'une base de données, c'est le modèle qui la contient. Le modèle offre des méthodes pour mettre à jour ces données (insertion, suppression, changement de valeur). Il offre aussi des méthodes pour récupérer ces données.
- **Vue** : La vue correspond à l'interface avec laquelle l'utilisateur interagit. Sa première tâche est de présenter les résultats renvoyés par le modèle.

Sa seconde tâche est de recevoir toutes les actions de l'utilisateur (clic de souris, sélection d'une entrée, boutons...). Ces différents événements sont envoyés au contrôleur. La vue n'effectue aucun traitement, elle se contente d'afficher les résultats des traitements effectués par le modèle. Plusieurs vues, partielles ou non, peuvent afficher des informations d'un même modèle.

- **Contrôleur** : Le contrôleur prend en charge la gestion des événements de synchronisation pour mettre à jour la vue ou le modèle et les synchroniser. Il reçoit tous les événements de l'utilisateur et enclenche les actions à effectuer. Si une action nécessite un changement des données, le contrôleur demande la modification des données au modèle et ensuite avertit la vue que les données ont changé pour qu'elle se mette 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. Le contrôleur n'effectue aucun traitement, ne modifie aucune donnée. Il analyse la requête du client et se contente d'appeler le modèle adéquat et de renvoyer la vue correspondant à la demande.

iv. Choix du SGBD :

PostgreSQL est un système de gestion de bases de données relationnel et objet (**SGBDOR**). Ces fonctionnalités sont comparables à celles des ses concurrents propriétaires comme Oracle.

PostgreSQL possède de nombreuses caractéristiques en faisant un **SGBDOR** robuste et puissant digne des **SGBD** commerciaux :

- des interfaces graphiques (X-Window est donc nécessaire) pour gérer les tables
- des bibliothèques pour de nombreux langages (appelés *frontaux*) afin d'accéder aux enregistrements à partir de programmes écrits en :
 - Java (JDBC)
 - langage C/C++
 - Perl
- une API **ODBC** permettant à n'importe quelle application supportant ce type d'interface d'accéder à des bases de données de type **PostgreSQL**.

PostgreSQL fonctionne selon une architecture **client/serveur**, il est ainsi constitué

- d'une partie serveur, c'est-à-dire une application fonctionnant sur la machine hébergeant la base de données (le serveur de bases de données) capable de traiter les requêtes des clients. Il s'agit dans le cas de **PostgreSQL** d'un programme résident en mémoire appelé *postmaster*.
- d'une partie client devant être installée sur toutes les machines nécessitant d'accéder au serveur de base de données (un client peut éventuellement fonctionner sur le serveur lui-même)



Les clients (les machines sur lesquelles le client **PostgreSQL** est installé) peuvent interroger le serveur de bases de données à l'aide de **requêtes SQL**.

PostgreSQL offre :

- La possibilité d'imposer des contraintes à l'insertion des données.
- Les contraintes d'intégrités (garantie d'intégrité des données).
- La possibilité de créer des vues.
- La possibilité d'hériter des tables et des types de données.
- Les procédures stockées (programmation coté serveur).
- Les transactions (plusieurs étapes en une seule opération).
- Les triggers (gestion des événements).

Après avoir étudié le cahier de charge et choisis les outils de développement, je passe à la phase de modélisation et de conception.

Chapitre III :

Modélisation et conception

I. Introduction

La plupart des nouveaux langages sont orientés objet. Le passage de la programmation fonctionnelle à l'orienté objet n'était pas facile. L'un des soucis était d'avoir une idée globale en avance de ce je dois programmer.

L'algorithmique qui était utilisé dans la programmation fonctionnelle lui seul ne pourrait pas suffire. Le besoin d'avoir des méthodes ou langages pour la modélisation des langages orientés objet se faisait sentir. Ainsi plusieurs méthodes ou langages on vu le jour. En occurrence UML qui nous a permis de faire la conception de notre système.

De nos jours UML2 possède treize diagrammes qui sont classés en deux catégories (dynamique et statique).

Pour ce faire j'ai commencé par les diagrammes de cas d'utilisation (Use Case) qui permettent de donner une vue globale de l'application.

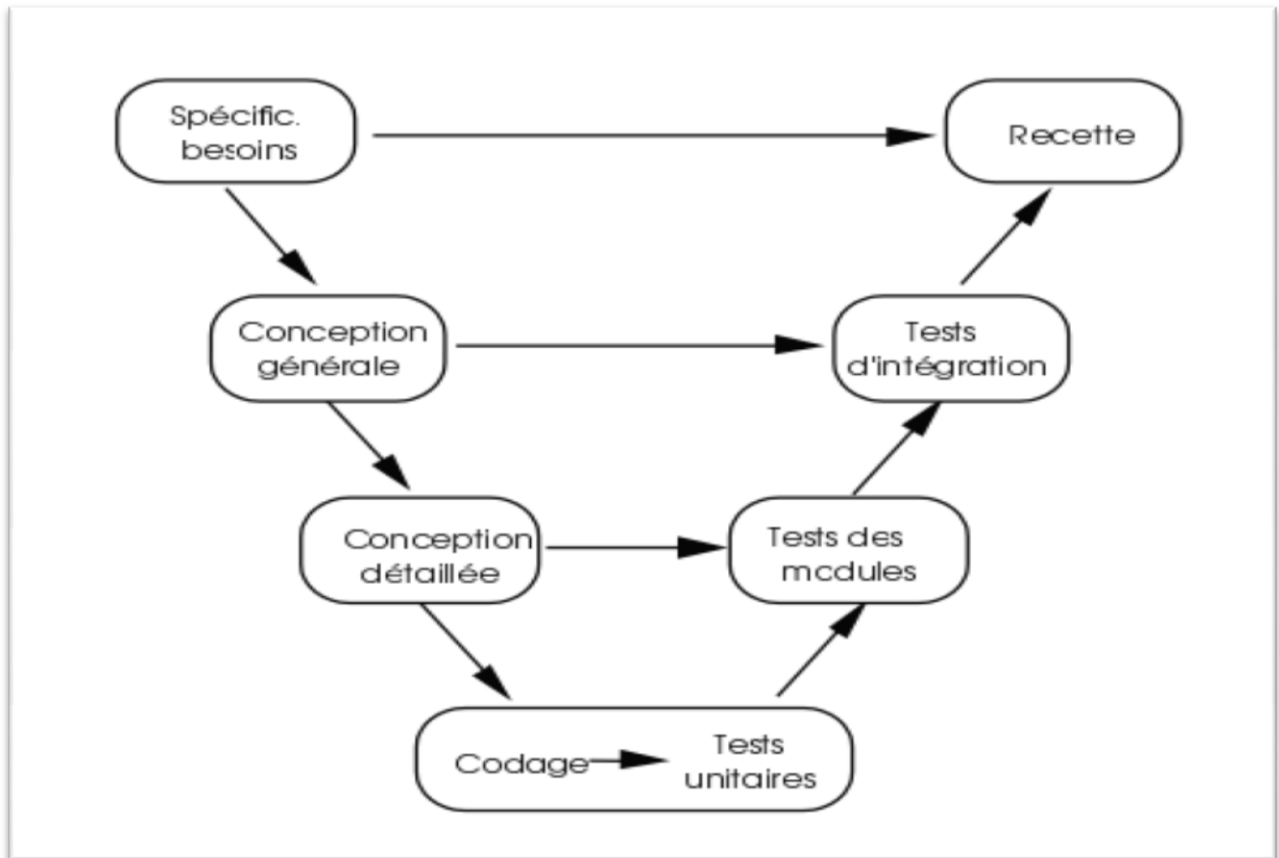
En deuxième lieu je vais présenter la chronologie des opérations par les diagrammes de séquences.

Et finir par les diagrammes statiques qui sont celles des classes et le modèle physique.

II. Processus de développement en Y

Le développement d'un logiciel se fait suivant un cycle appelé le cycle de vie du logiciel.

Schéma représentant le processus de développement d'un logiciel classique :



Ces phases sont échelonnées dans le temps. Une phase se termine par la remise d'un (ou plusieurs) (s) validé(s) conjointement par l'utilisateur et le développeur. Une phase de développement se termine lorsque la revue de cette phase est faite. Une phase ne peut commencer que lorsque la précédente est terminée. A la fin de chaque phase, l'utilisateur et le développeur sont d'accord. La décomposition en phases de développement permet donc le suivi du projet par l'utilisateur

III. Diagramme des cas d'utilisation :

Chaque cas d'utilisation doit faire l'objet d'une définition à priori, qui décrit l'intention de l'acteur lorsqu'il utilise le système et les séquences d'actions principales, qu'il est susceptible d'effectuer. Ces définitions servent à fixer les idées lors de l'identification des cas d'utilisation et n'ont aucun caractère exhaustif.

I. Acteurs du système :

- Le salarié qui saisie la demande de congé.
- Le responsable qui valide ou refuse la demande de congé.
- La direction R.H qui valide aussi les demandes de congé et affiche la liste des employés en congé.

II. Message entre le système et ses acteurs :

Acteur	Messages envoyés au système.	Messages émis par le système.
Salarié	<ul style="list-style-type: none"> ☞ Identification ☞ Saisie des informations concernant la demande de congé. ☞ Envoi de la demande au responsable 	<ul style="list-style-type: none"> ☞ Validation de l'identification ☞ Confirmation de la saisie. ☞ Accusé de réception.
Responsable	<ul style="list-style-type: none"> ☞ Identification ☞ Consultation des demandes de congé 	<ul style="list-style-type: none"> ☞ Validation de l'identification ☞ Affichage des demandes
Direction R.H	<ul style="list-style-type: none"> ☞ Identification. ☞ Consultation des demandes. ☞ Impression de demandes. ☞ Consultation 	<ul style="list-style-type: none"> ☞ Validation de l'indentification. ☞ Affichage des demandes. ☞ Affichage de la liste des employés en congés.

III. Identification des cas d'utilisation :

Cas d'utilisation	Acteurs	Messages reçu/émis
Authentification	Salariés	☞ Emis : Envoi du login et du mot de passe ☞ Reçu : Validation de l'indentification.
	Direction R.H	
	Responsable	
Saisie de demande de congé.	Salariés	☞ Reçu : Confirmation de saisie
Consultation d'états	Salariés	☞ Reçu : Affichage des jours de congés
Envoi de demande de congé.	Salariés	☞ Emis : Demande de congé. ☞ Reçu : Accusé de réception.
Consultation des demandes de congé.	Direction R.H	☞ Reçu: Affichage de demandes
	Responsable	
Réponse aux demandes.	Responsable	☞ Emis : Validation/refus des demandes.
	Direction R.H	
Consultation des employés.	Direction R.H	☞ Reçu : affichage des listes.
Impression des demandes de congé	Direction R.H	☞ Emis : Demande de congé.

IV. Diagramme des cas d'utilisation :

✓ Acteur : Salarié

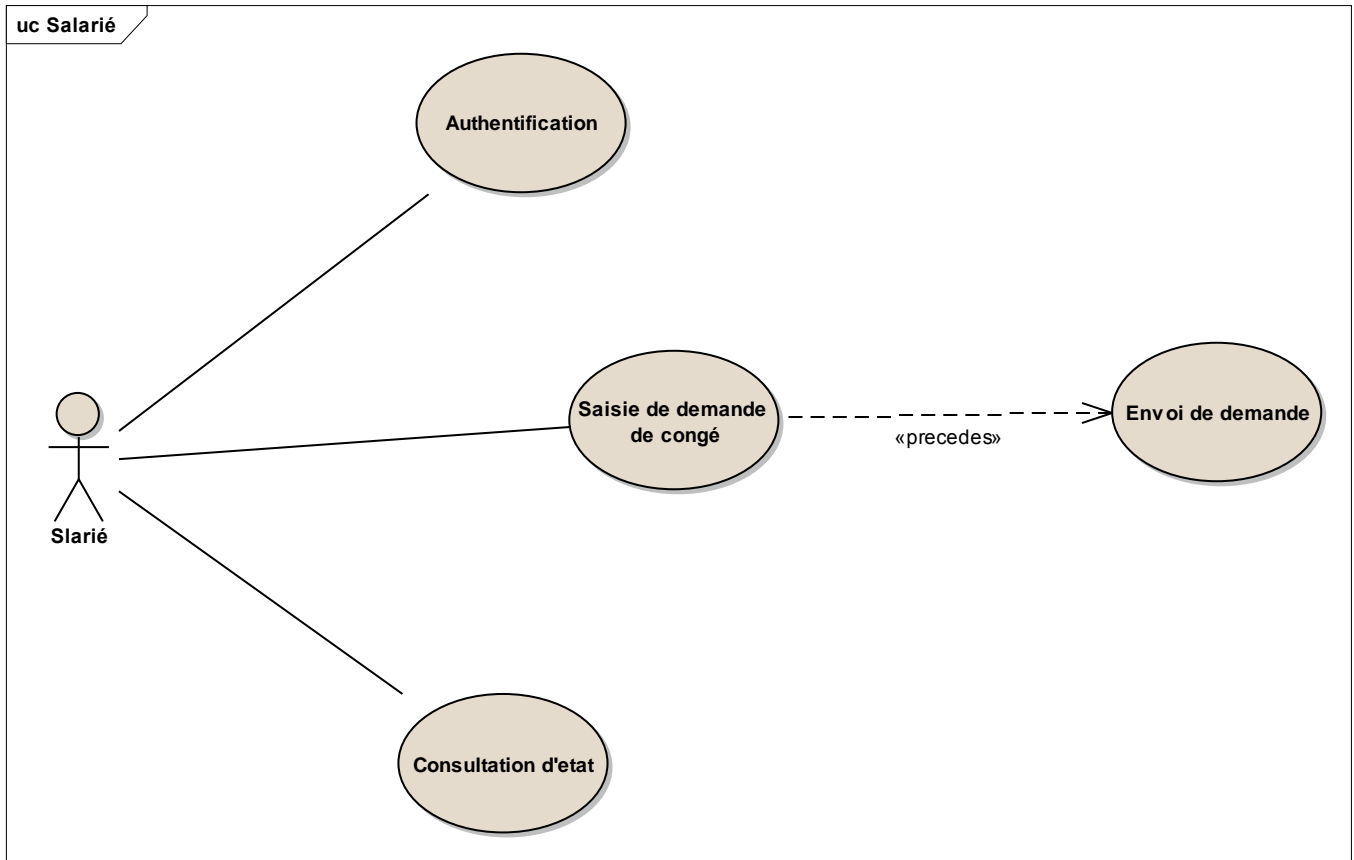


Figure 1 : Diagramme de cas d'utilisation pour l'acteur salarié

✓ Acteur : Responsable

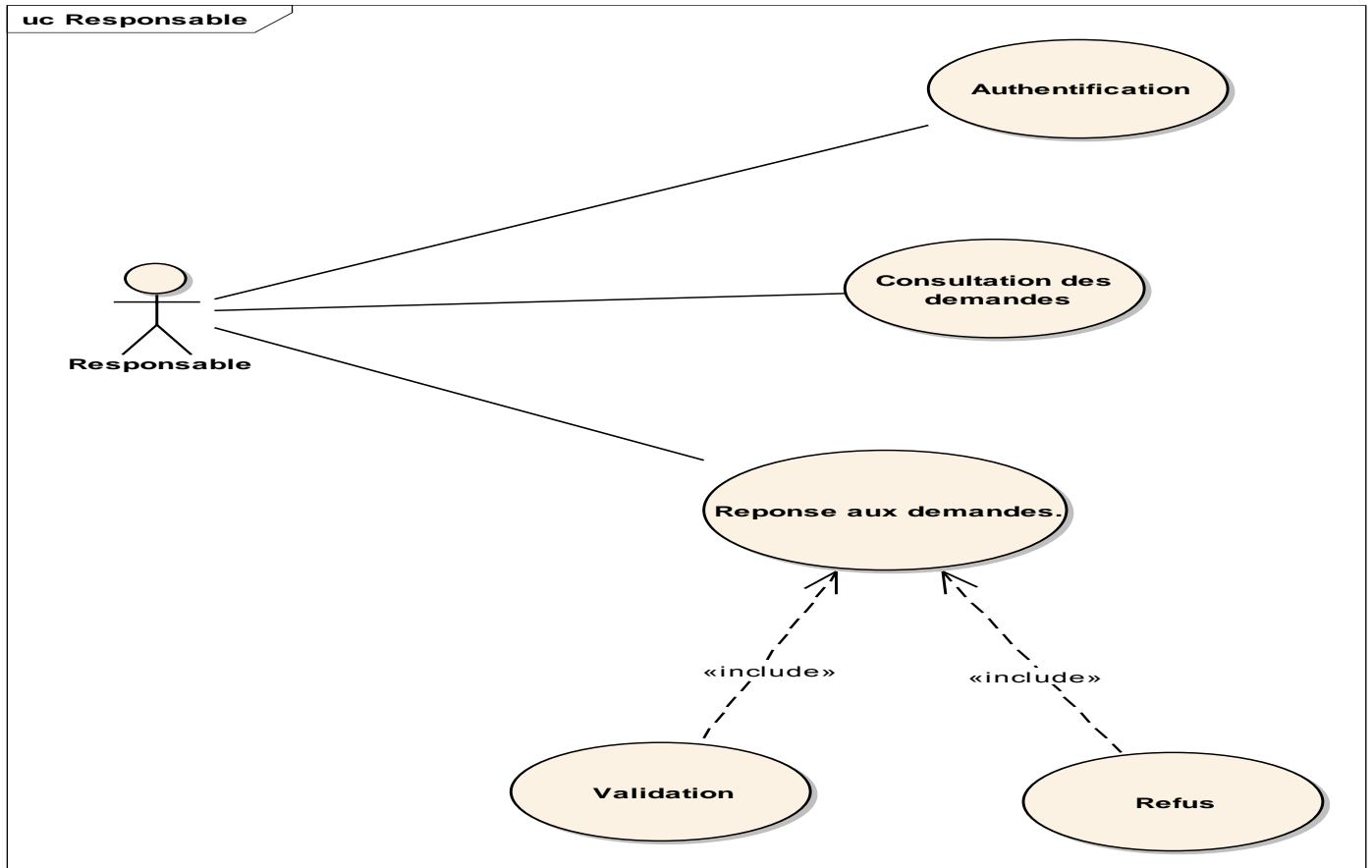


Figure 2 : Diagramme de cas d'utilisation pour l'acteur responsable

✓ Acteur : Direction R.H

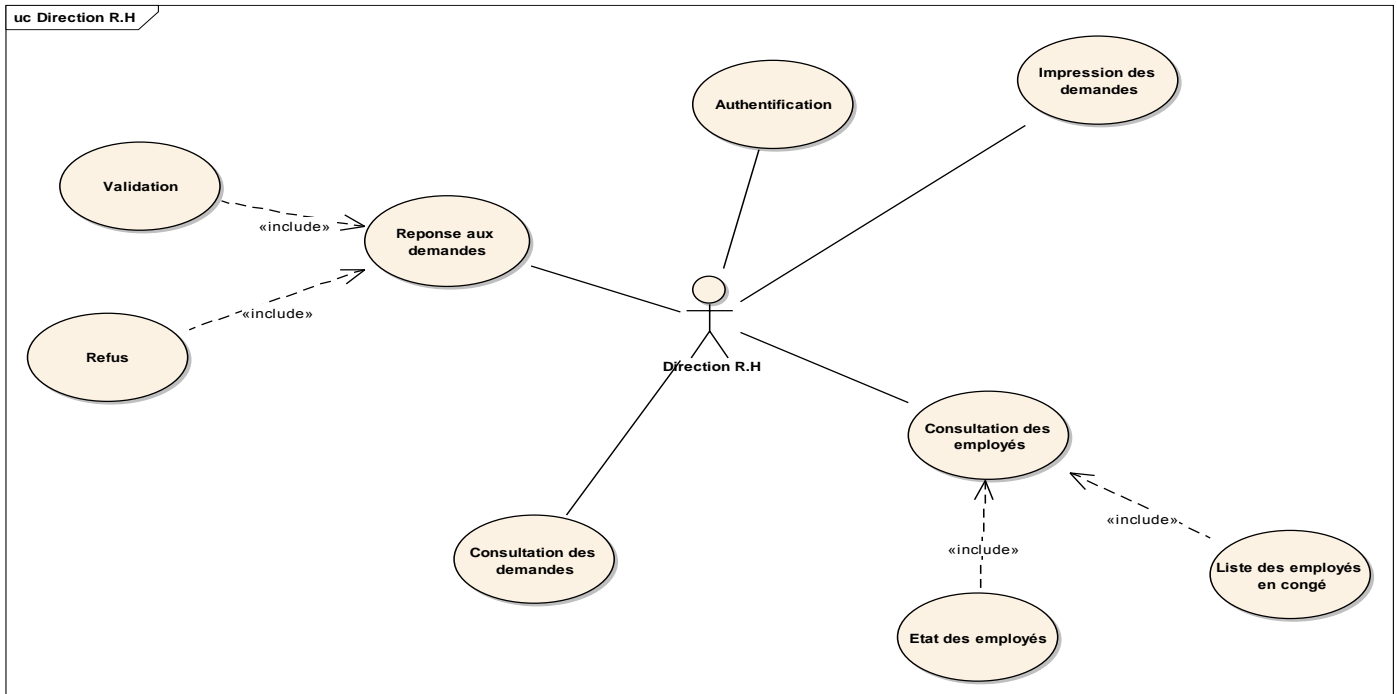


Figure 3 : Diagramme de cas d'utilisation pour l'acteur direction R.H

IV. Modèle statique :

Les classes candidates seront présentées comme suit :
Pour chaque cas d'utilisation on associé un diagramme de classe préliminaire.

⚡ Authentification :

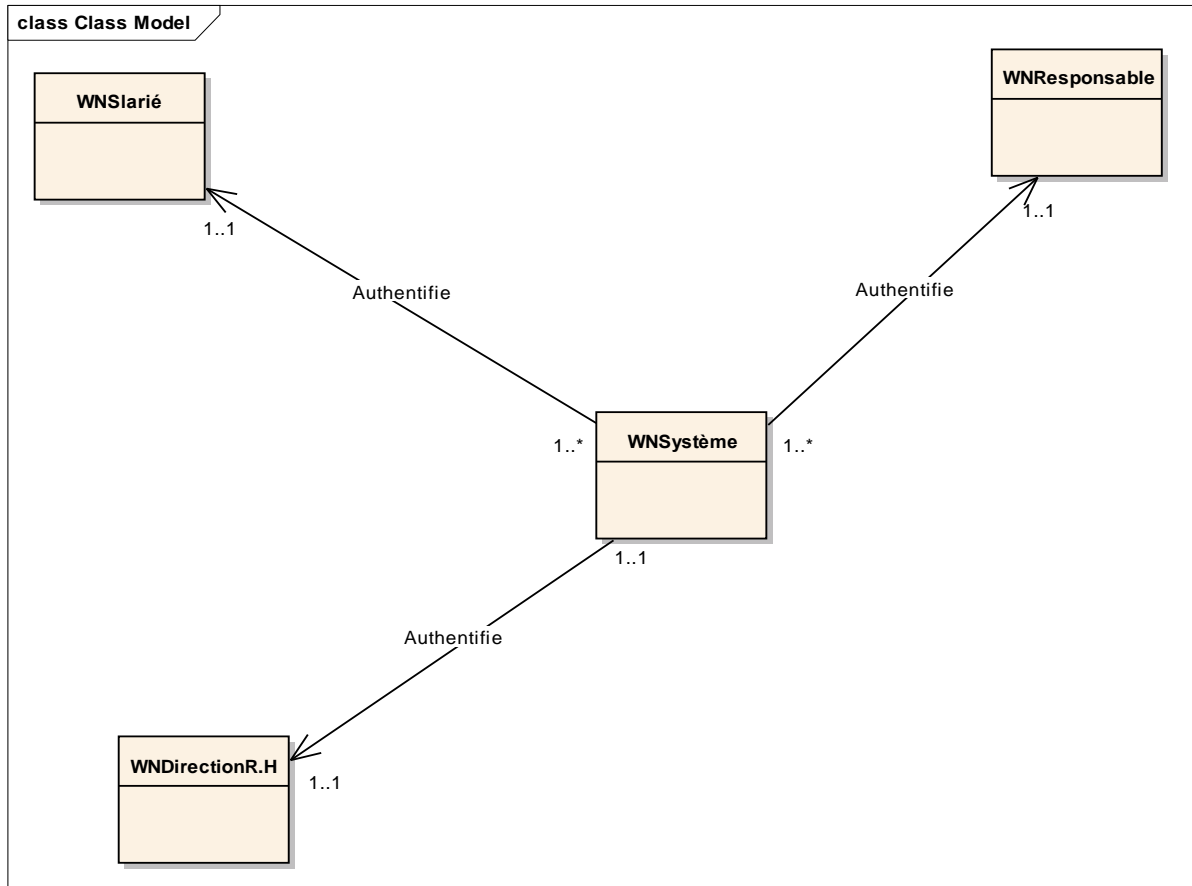


Figure 4: Diagramme de classe préliminaire pour le cas d'authentification

⚡ Saisie de demande de congé :



Figure 5: Diagramme de classe préliminaire pour le cas de saisie de demande de congé

⚡ Consultation d'état :

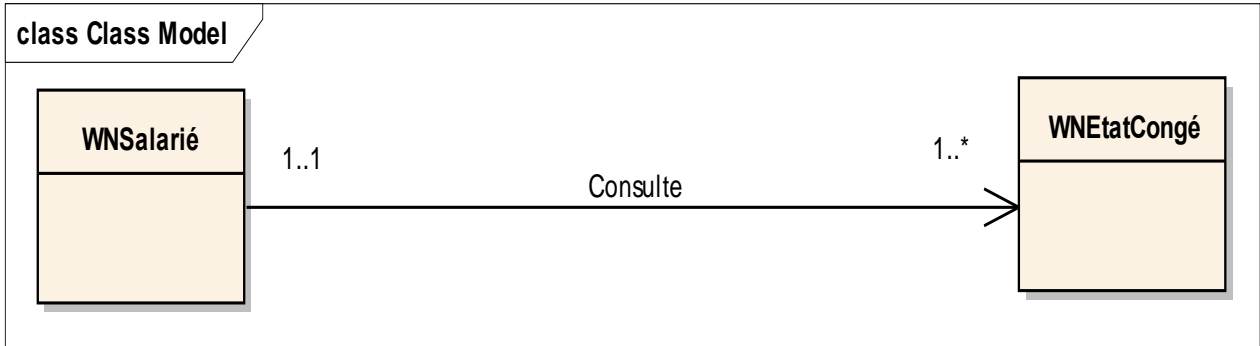


Figure 6: Diagramme de classe préliminaire pour le cas de consultation d'état

⚡ Envoi de demande de congé :

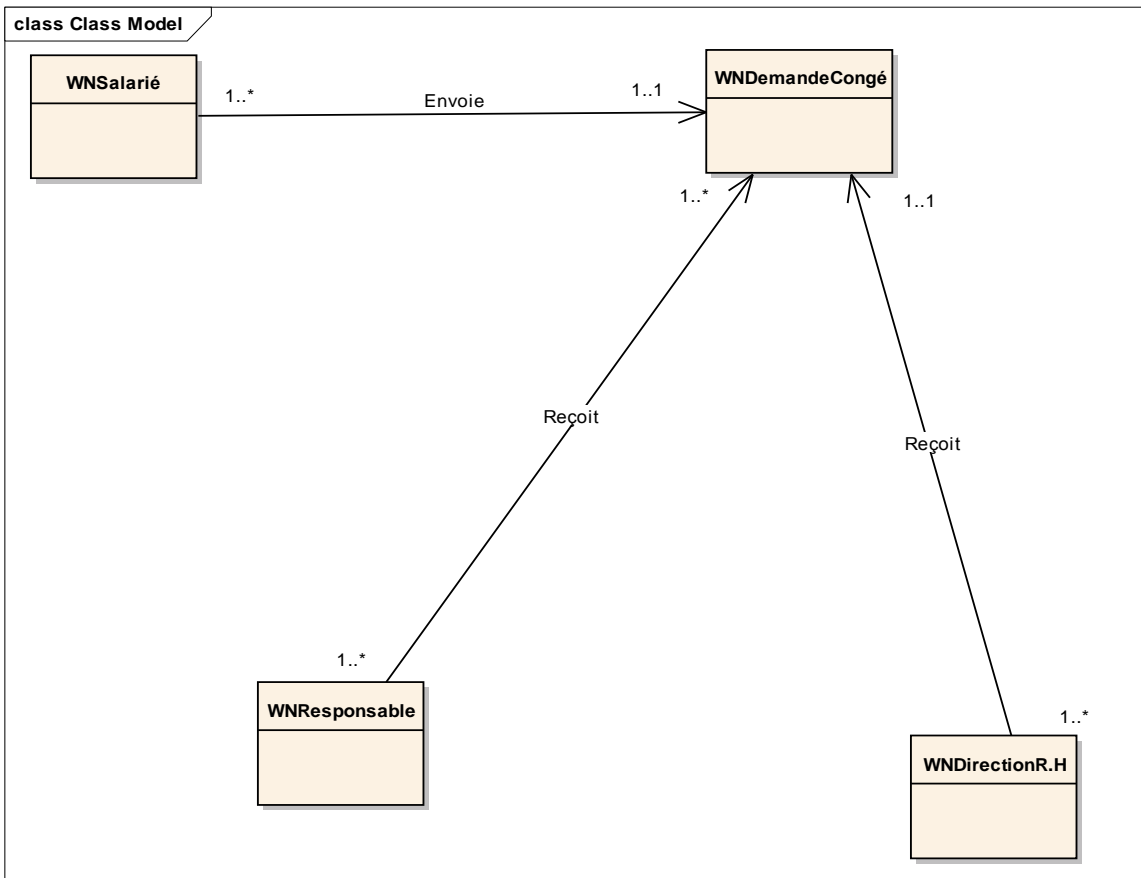


Figure 7: Diagramme de classe préliminaire pour le cas d'envoi de demande

⚡ Consultation des demandes de congé :

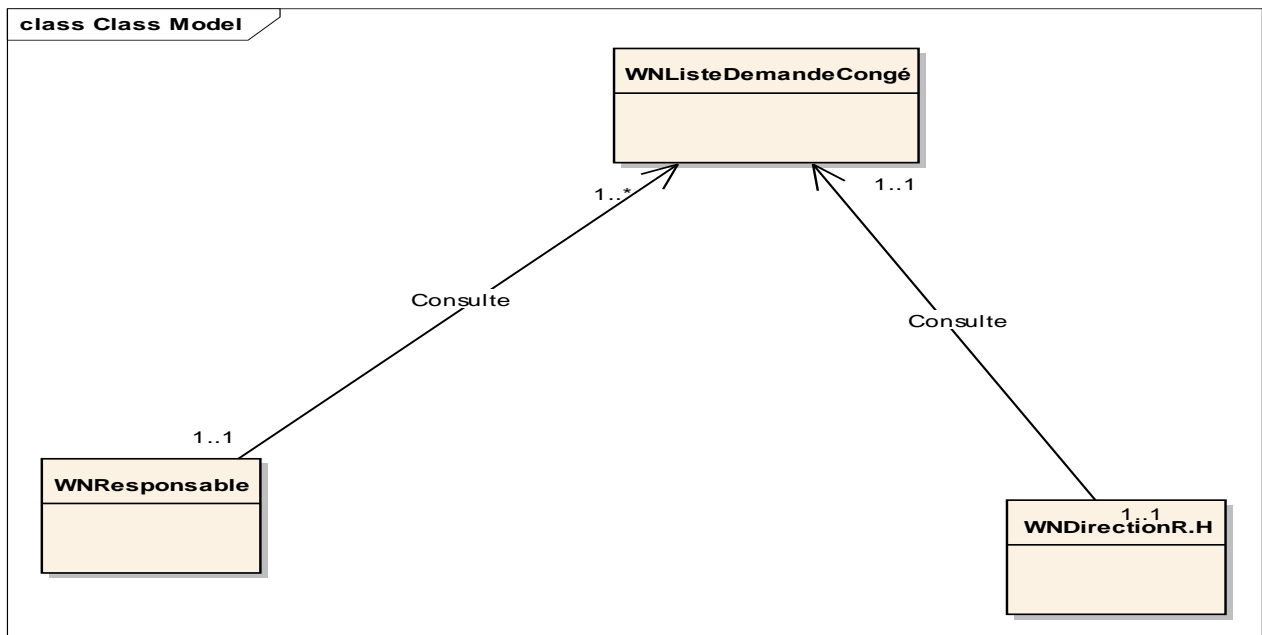


Figure 8: Diagramme de classe préliminaire pour le cas de consultation des demandes

⚡ Validation des demandes :

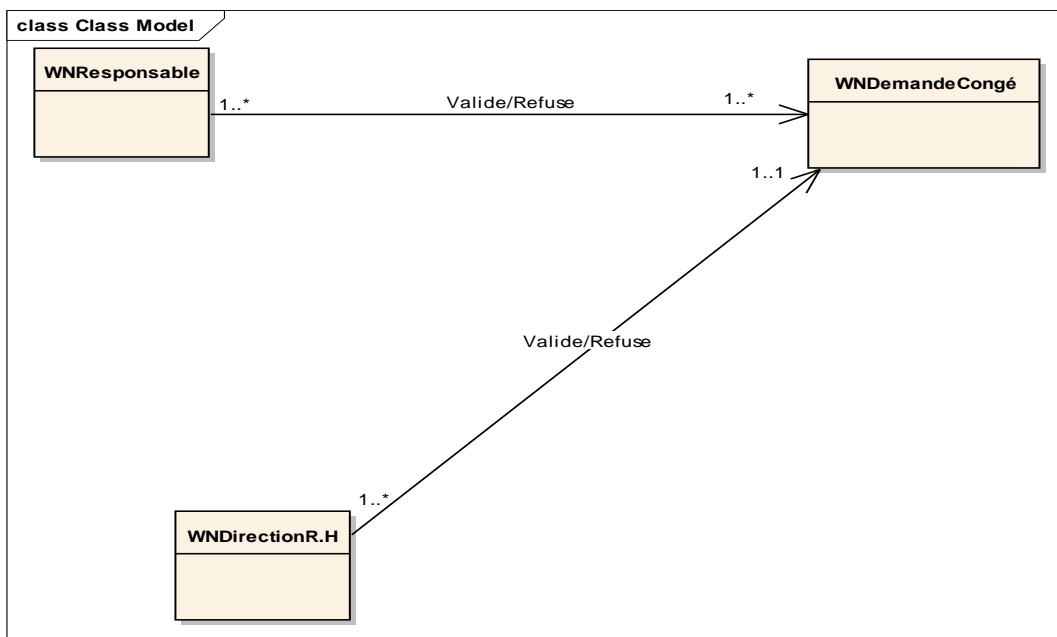


Figure 9: Diagramme de classe préliminaire pour le cas de validation des demandes

▣ Consultation de la liste des employés :

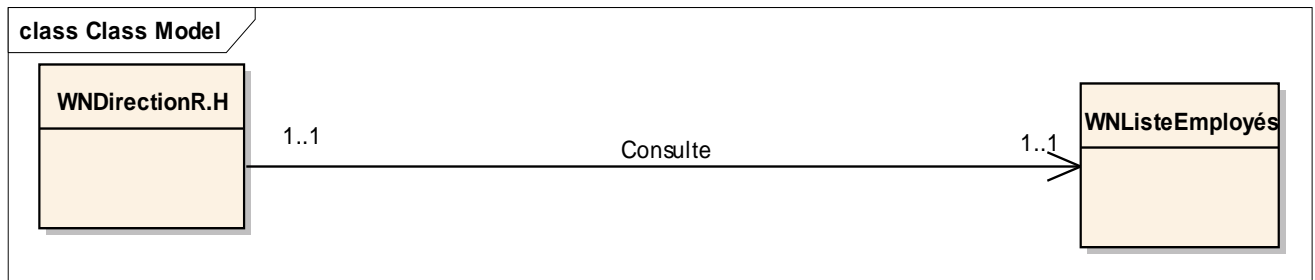


Figure 10: Diagramme de classe préliminaire pour le cas de consultation des listes des employés.

V. Modèle dynamique :

Représentation des scénarios par un diagramme de séquence pour chaque cas d'utilisation.

⚡ Authentification :

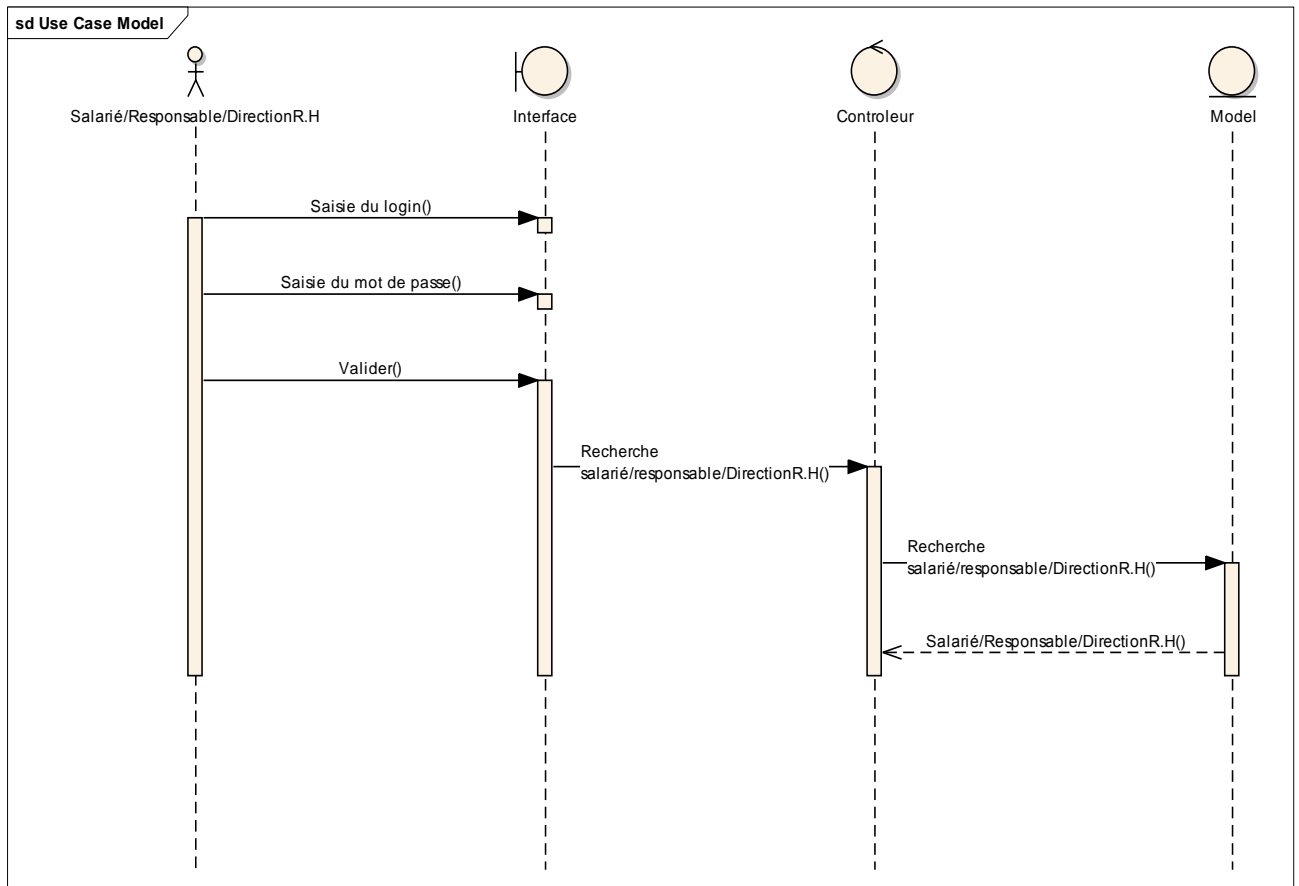


Figure 11: Diagramme de séquence pour l'étape d'authentification

⚡ Saisie de demande de congé :

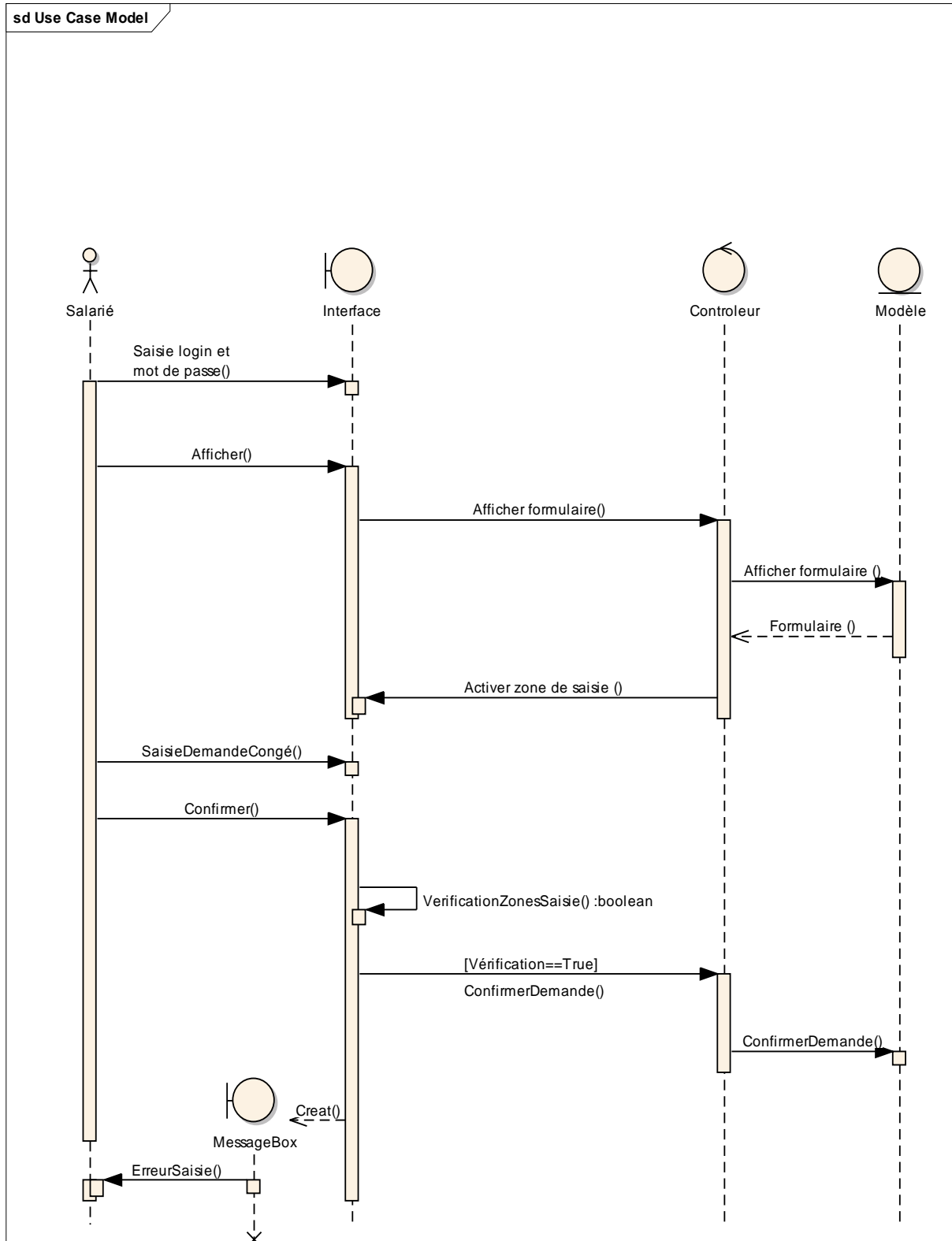


Figure 12: Diagramme de séquence pour l'étape de saisie des demande

Consultation d'état :

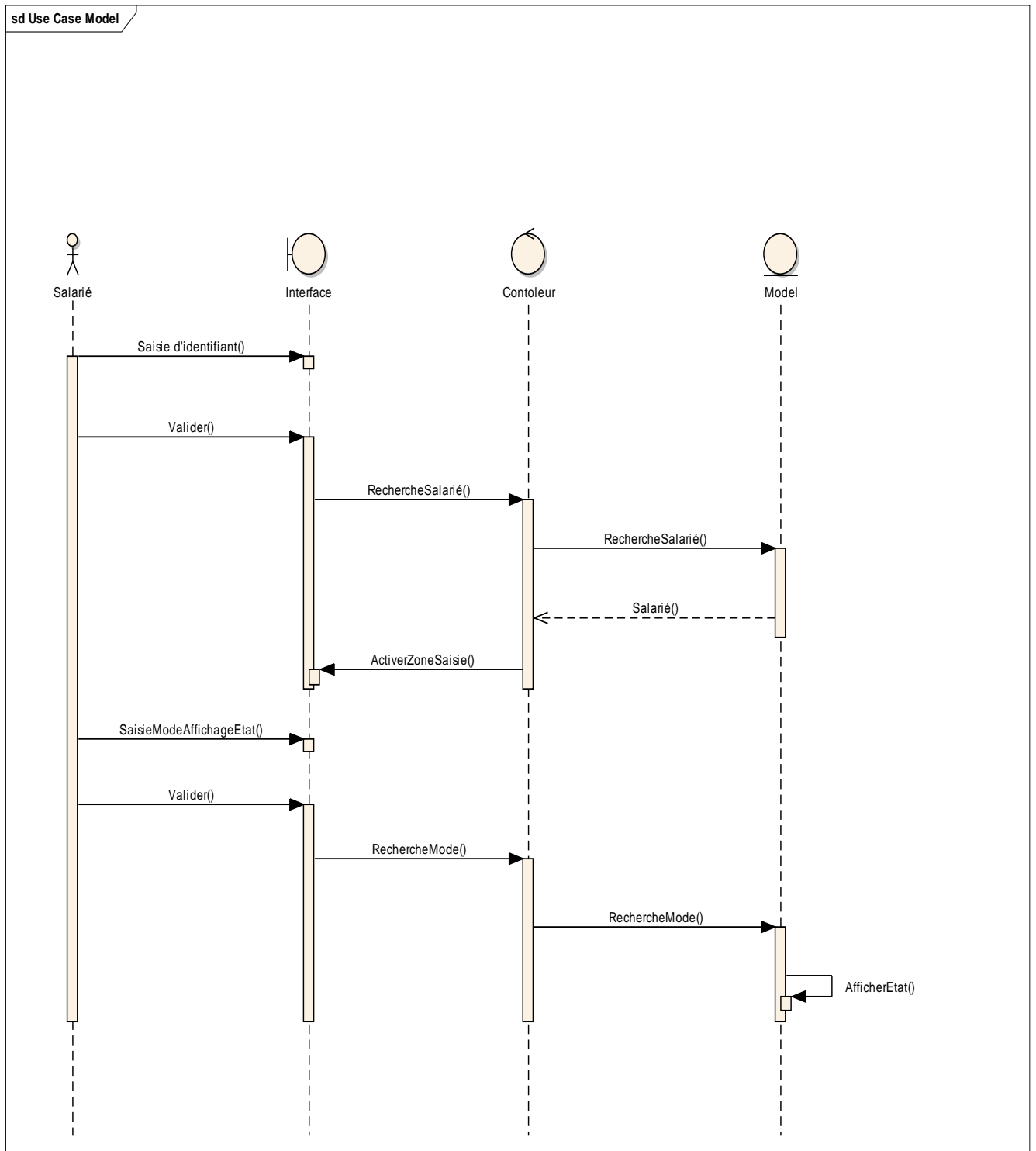


Figure 13: Diagramme de séquence pour la consultation d'états de congé

Envoi demande de congé:

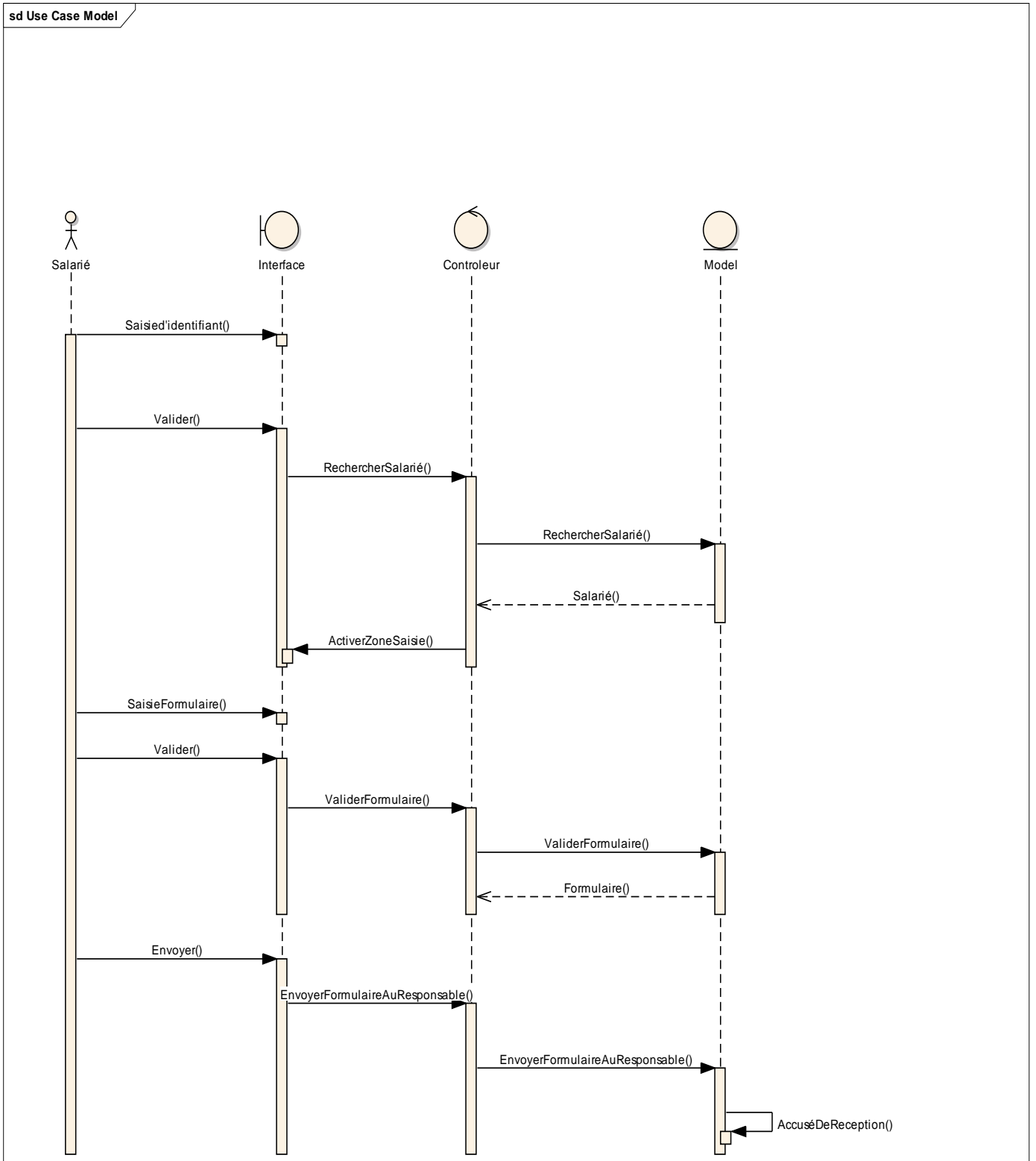


Figure 14: Diagramme de séquence pour l'étape d'envoi de demande de congé

Consultation demandes de congé:

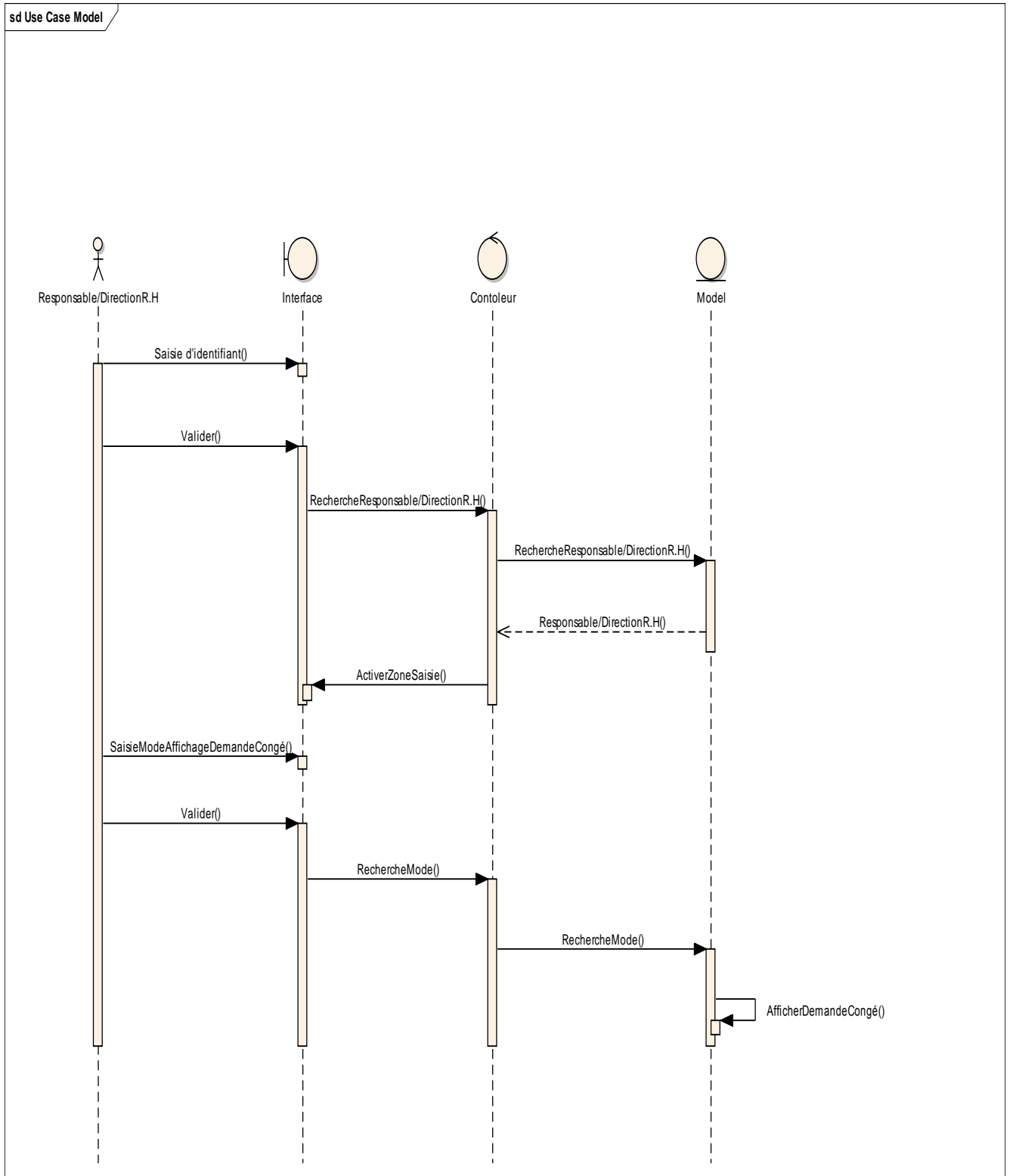


Figure 15: Diagramme de séquence pour de consultation des demandes

☒ Réponse aux demandes :

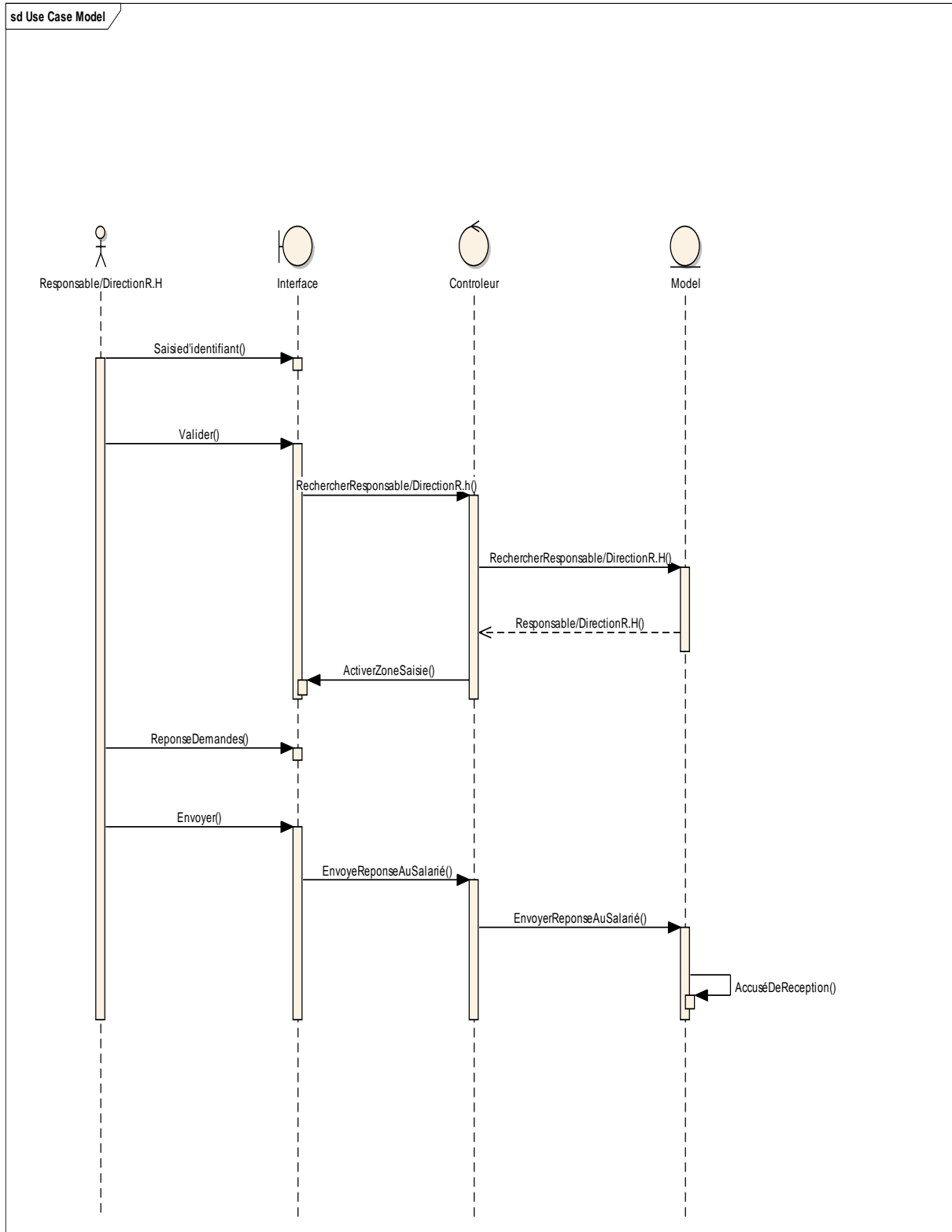


Figure 16: Diagramme de séquence pour l'envoi de réponse de demande

Consultation des employés :

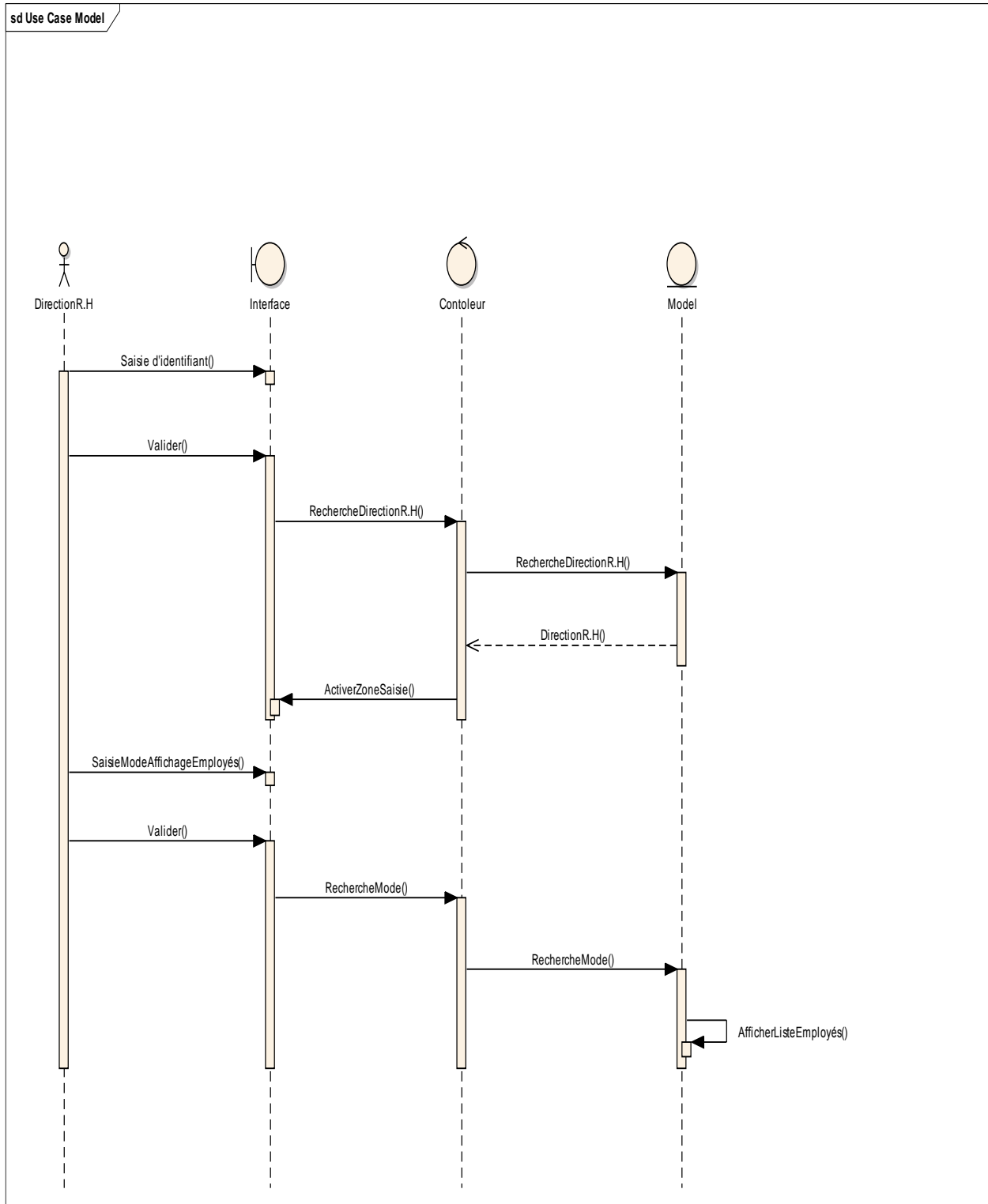
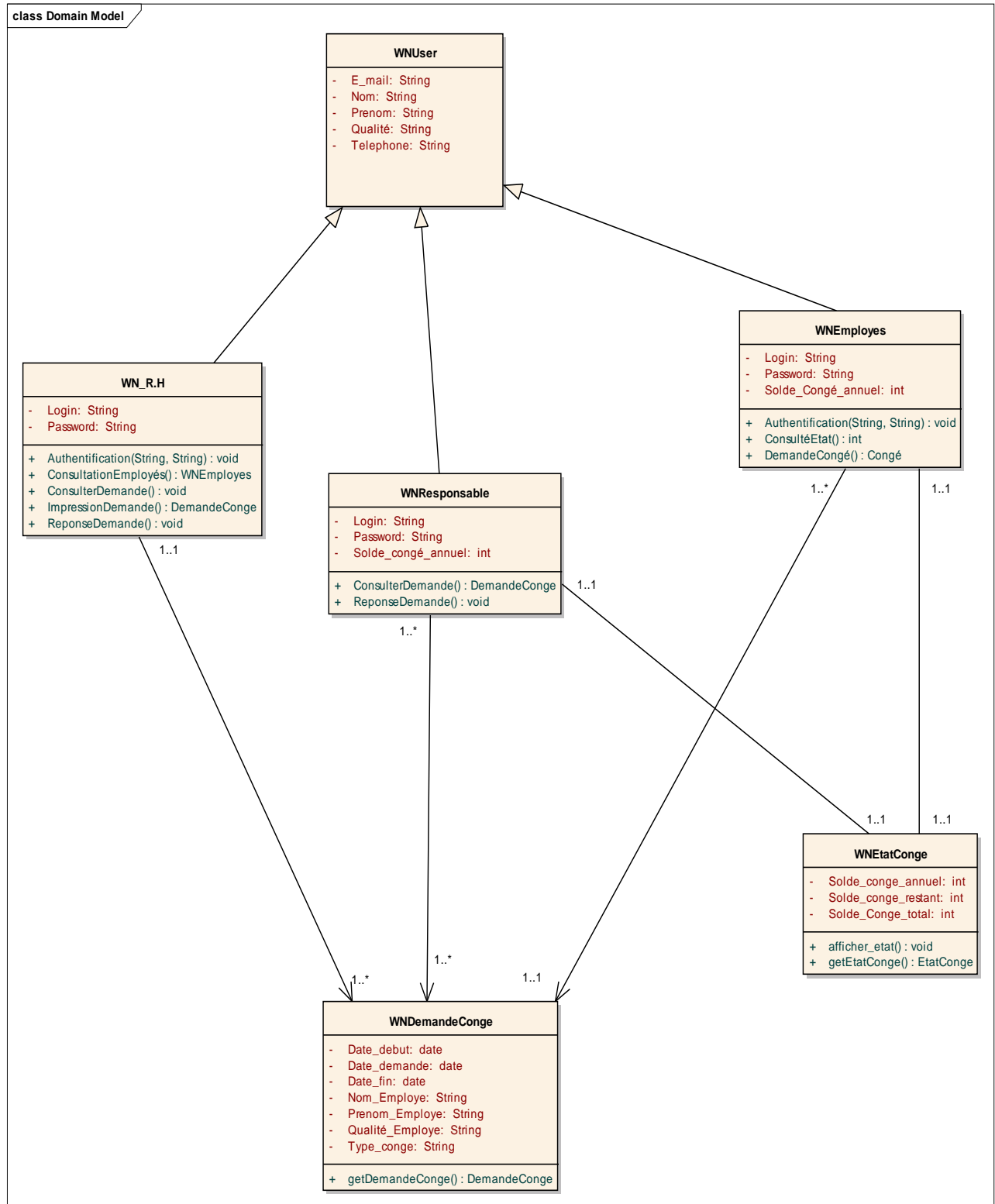


Figure 17: Diagramme de séquence pour l'étape de consultation des employés



VI. Diagramme de classe :



Le diagramme de classe permet de spécifier les différentes classes de mon application, et est indispensable pour la phase du codage.



Chapitre IV :

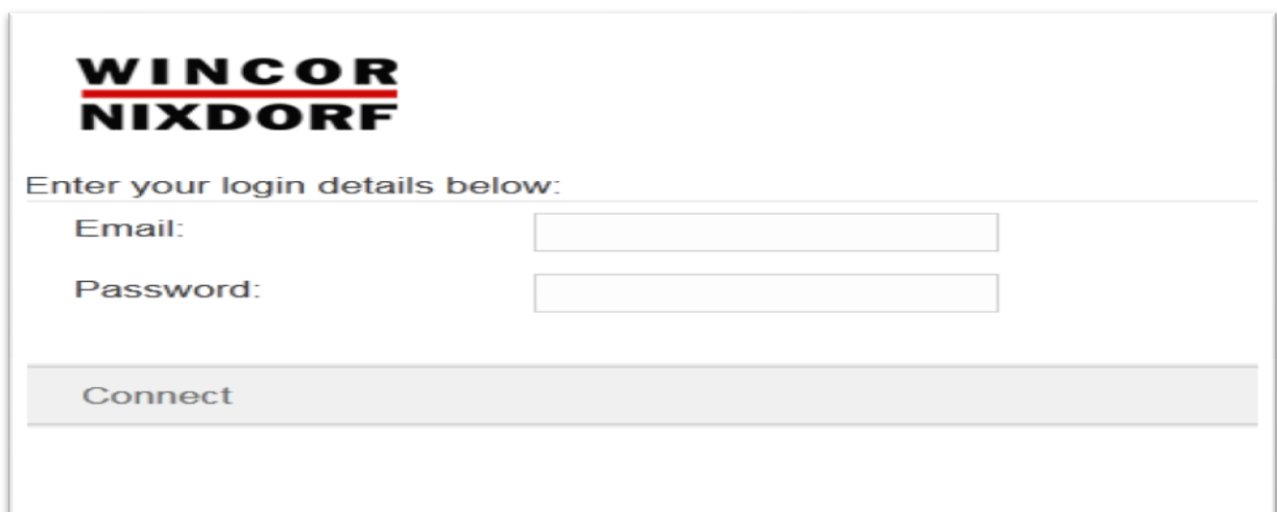
Réalisation

I. Introduction

La phase de réalisation est la plus importante étape dans le cycle de vie de mon application car à la fin de cette phase j'aurais le fruit de toutes les étapes précédentes. Pour réaliser mon application j'ai été face à plusieurs outils de développement dont j'ai choisi «SpringSourcetoolsuite » suite à plusieurs contraintes matérielles et logicielles.

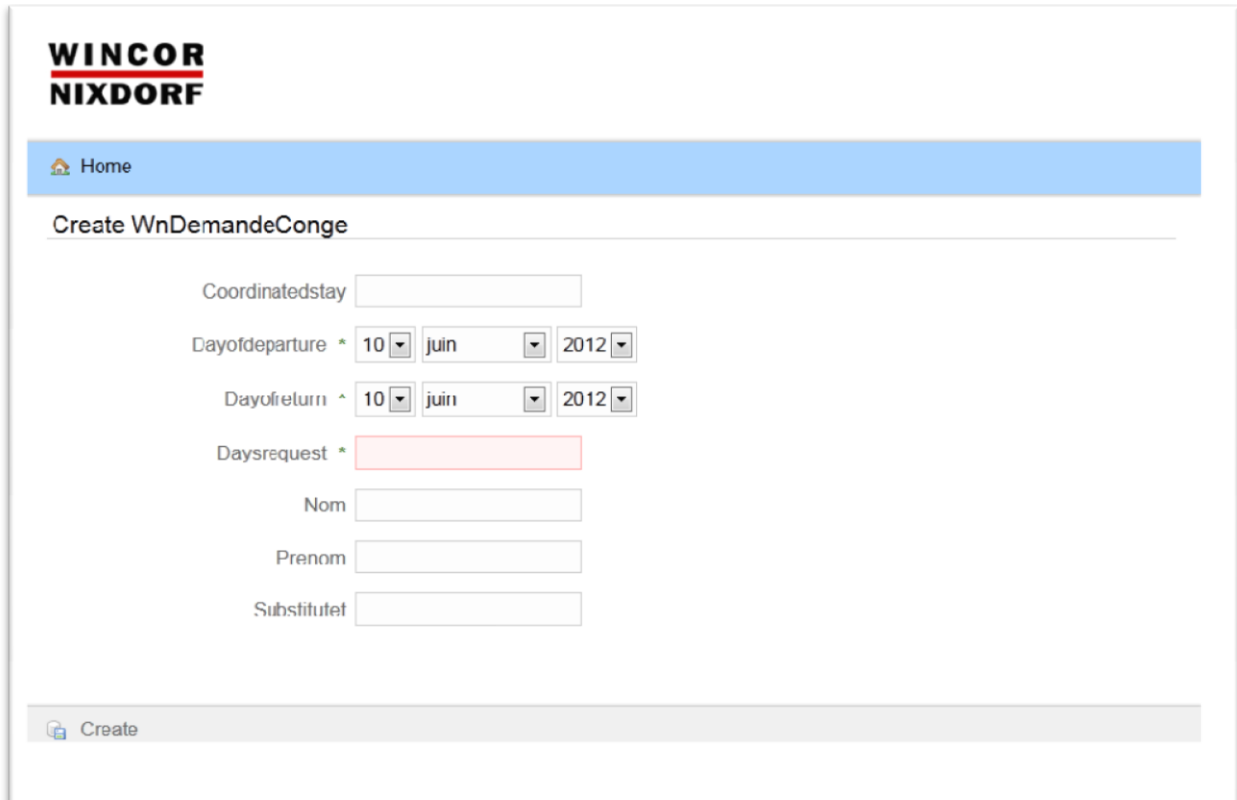
II. Interfaces de l'application :

❖ Authentification :



The screenshot shows the login interface for Wincor Nixdorf. At the top left is the Wincor Nixdorf logo. Below it, the text "Enter your login details below:" is displayed. There are two input fields: "Email:" and "Password:". Below the input fields is a "Connect" button.

❖ Saisie des demandes de congé :



**WINCOR
NIXDORF**

Home

Create WnDemandeConge

Coordinatedstay

Dayofdeparture * 10 juin 2012

Dayofreturn ^ 10 juin 2012

Daysrequest *

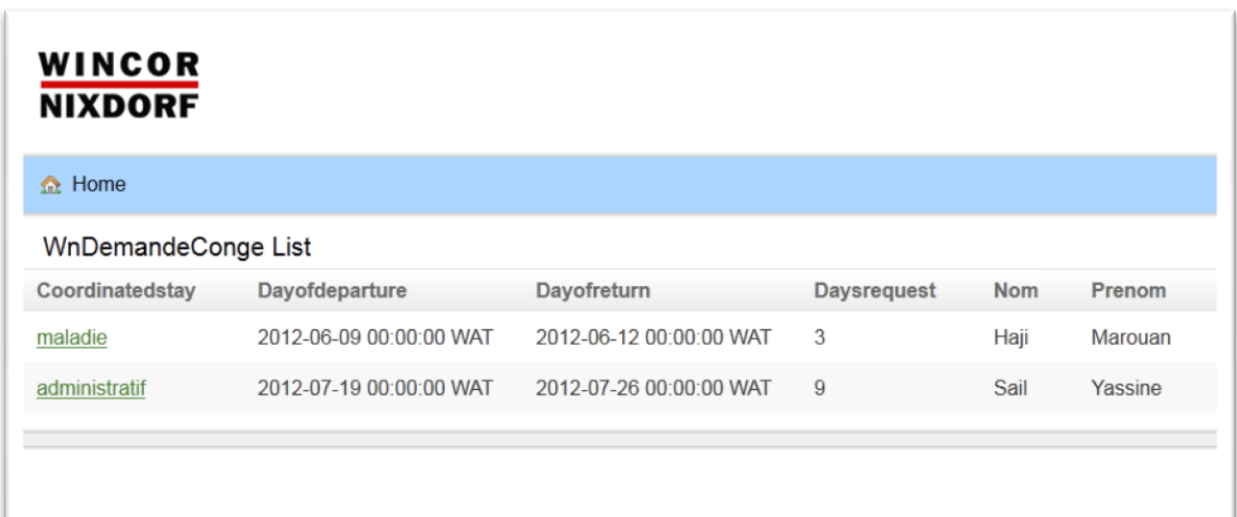
Nom

Prenom

Substitutef

Create

❖ Liste des demandes de congé :



**WINCOR
NIXDORF**

Home

WnDemandeConge List

Coordinatedstay	Dayofdeparture	Dayofreturn	Daysrequest	Nom	Prenom
maladie	2012-06-09 00:00:00 WAT	2012-06-12 00:00:00 WAT	3	Haji	Marouan
administratif	2012-07-19 00:00:00 WAT	2012-07-26 00:00:00 WAT	9	Sail	Yassine



❖ Liste des utilisateurs :

[Home](#)

WnUser List

Email	Login	Nom	Numbersofdaysoff	Password	Prenom
exp@exp.com	log	Saidaoui	24	pass	Khalid
eg@eg.com	login	Haji	24	password	Marouan

❖ L'ajout d'un nouveau utilisateur

WINCOR NIXDORF

[Home](#)

Create WnUser

Email

Login

Nom

Numbersofdaysoff *

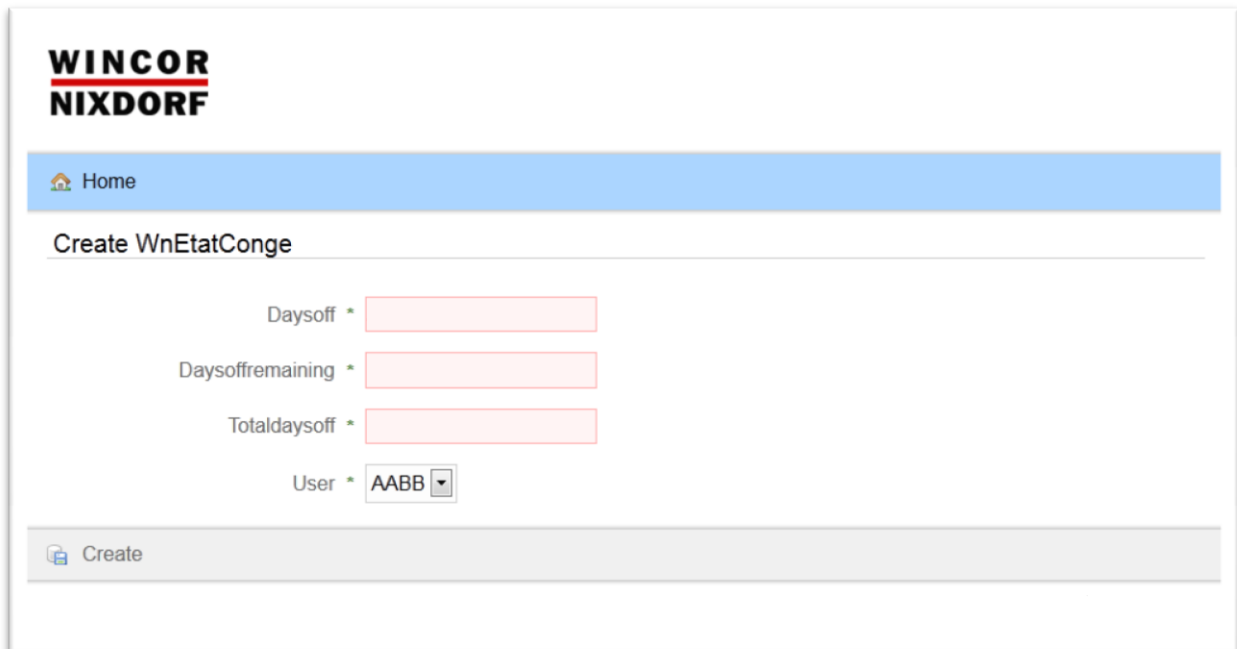
Password

Prenom

Qualite


Tel

❖ Saisie des états de congé



The screenshot shows the 'Create WnEtatConge' form. It features the WINCOR NIXDORF logo at the top left, a 'Home' navigation bar, and the title 'Create WnEtatConge'. Below the title, there are four input fields: 'Daysoff *', 'Daysoffremaining *', 'Totaldaysoff *', and 'User *'. The 'User' field is a dropdown menu currently showing 'AABB'. At the bottom of the form, there is a 'Create' button.

❖ Affichage des états de congé :



The screenshot shows the 'WnEtatConge List' table. It features the WINCOR NIXDORF logo at the top left, a 'Home' navigation bar, and the title 'WnEtatConge List'. Below the title, there is a table with the following data:

Daysoff	Daysoffremaining	Totaldaysoff	User
<u>24</u>	6	30	HajiMarouan

Ces captures d'écran représentent les différentes interfaces de mon application.

Conclusion et perspectives :

La réalisation de ce projet avait comme objectif la simplification et l'amélioration de la gestion et de la saisie des demandes de congés à l'entreprise en question.

J'ai appliqué au maximum possible les règles de bases permettant d'avoir une application performante. J'ai déployé le formalisme UML pour concevoir une grande partie du travail et l'utilisation des technologies J2EE accompagnés du Framework Grails pour mettre en œuvre mon projet.

Ce projet était bénéfique pour moi dans plusieurs sens. Il m'a permis :

- ↳ De perfectionner en améliorant mes connaissances en programmation et en conception.
- ↳ De bien comprendre et mettre en œuvre le déroulement d'un cycle de vie d'un logiciel.
- ↳ De découvrir le monde de l'entreprise (fonctionnement).
- ↳ De concrétiser mes acquis durant mes années d'étude à la faculté des sciences et technique.

Durant toute cette période de stage, j'ai réalisé la grande partie de mon projet qui se focalise dans la réalisation de l'application de gestion des congés, mais suite à une contrainte de temps, je n'ai pas eu l'occasion de compléter le reste des modules de mon application, raison pour laquelle j'ai l'intention de prolonger mon stage au sien de l'entreprise.



Webographie :

☞ Persistance Java 5 par la pratique :

<http://tahe.developpez.com/java/jpa/>

☞ Java EE - Servlet - JSP – JSF :

<http://manu.e3b.org/Java/Tutoriels/J2EE>

http://www.jmdoudoux.fr/accueil_java.htm

☞ Framework grails :

<http://www.grails.org/tutorials>

☞ Installation de SpringSource tool suite :

<http://www.springsource.org/springsource-tool-suite-download>