

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
Ministre de l'enseignement supérieur et de la recherche scientifique
Université Mouloud Mammeri de Tizi-Ouzou
Faculté de Génie Electrique et d'informatique
Département d'informatique



MEMOIRE

DE FIN D'ETUDE

En vue de l'obtention du diplôme de Master 2 en informatique LMD



*Conception et réalisation d'un système de gestion
de flux d'informations*

Cas : « Cevital (agro-industriel) »

Promotrice :

 *M^{me} DJAMAH*

Co-promotrice :

 *Mr TEBANI Hakim*

Réalisé par :

 *M^{lle} HAMZI Ghania*

 *M^{lle} HEDJAR Scoura*

Promotion 2016-2017

Liste des figures

Numéro	Description	Page
01	Organigramme de ce vital agro-industriel	06
02	Organigramme de LALLA KHEDIDJA	09
03	Diagramme de contexte de l'application	15
04	Diagramme de cas d'utilisation globale	18
05	Diagramme de cas d'utilisation «gérer les circuit de validation »	19
06	Diagramme de cas d'utilisation «gérer les employé »	19
07	Diagramme de cas d'utilisation «gérer les parcours »	19
08	Diagramme de cas d'utilisation «Etablir les documents »	20
09	Diagramme de cas d'utilisation « gérer ordre de mission »	21
10	Diagramme de cas d'utilisation « gérer effectuer une demande pour employé »	22
11	Diagramme de cas d'utilisation «valider une demande »	24
12	Diagramme de cas d'utilisation gérer les comptes utilisateurs »	25
13	Diagramme de cas d'utilisation « gérer les circuits de validation »	25
14	Tableau récapitulatif des scénarios	26
15	Diagramme de séquence illustrant le cas d'utilisation « valider une demande de congé de récupération»	30
16	Diagramme de séquence illustrant le cas d'utilisation «effectuer une demande d'absence »	31
17	Diagramme de séquence illustrant le cas d'utilisation «ajouter un compte utilisateur »	32
18	Diagramme d'activité du cas d'utilisation «Identification »	34
19	Diagramme d'activité du cas d'utilisation «Effectuer une demande »	35
20	Diagramme d'activité du cas d'utilisation «Valider une demande »	36
21	Diagramme de classe du cas d'utilisation «Ajouter un employé »	39
22	Diagramme de classe du cas d'utilisation «Modifier compte utilisateur »	40
23	Diagramme de classe du cas d'utilisation «Effectuer une réservation d'hôtel »	41
24	Diagramme de classe du cas d'utilisation «Valider une demande »	41
25	Diagramme de classe du paquetage « ADMIN »	42
26	Diagramme de classe du cas d'utilisation «GRH »	44
27	Diagramme de classe du cas d'utilisation « GMGX »	45
28	Diagramme de classe du cas d'utilisation « USER »	46
29	Diagramme de classe des interfaces de l'application	47
30	Page démarrage NetBeans	68
31	Interface de PHPMyadmin	70
32	Page « Login »	72
33	Page « Modifier pwd »	72
34	Page « GRH »	73
35	Page « GMGX »	73
36	Page « Administrateur »	74
37	Messagerie personnelle	74
38	Page « effectuer une demande »	75

SOMMAIRE

INTRODUCTION GENERALE.....	01
Chapitre 1 : Analyse de l'existant	
Introduction	03
1. Présentation de CEVITAL agro-industriel.....	04
1.1. Historique.....	04
1.2. Présentation de la direction générale de CEVITAL Agro-industriel	04
1.3. L'organigramme de CEVITAL Agro-industriel	06
2. Repérage du domaine d'étude.....	07
2.1 Présentation de LALLA KHEDIDJA	07
2.2 Présentation de l'organisme de LALLA KHEDIDJA	07
2.3 L'organisme de LALLA KHEDIDJA	09
3. Présentation de la technologie <i>WORKFLOW</i>	10
3.1 Définition du <i>WORKFLOW</i>	10
3.2 Utile du Workflow	10
3.3 L'accès à l'information dans un <i>WorkFlow</i>	10
3.4 Types d'applications <i>WorkFlow</i>	11
3.5 Domaines d'application des <i>WorkFlows</i>	11
3.6 Avantages d'une application <i>WorkFlow</i>	12
4. Présentation du langage de conception	13
4.1. Choix de la méthode de conception	13
4.2. Présentation du langage UML	13
4.3. Définition des 9 diagrammes du langage UML	13
4.3.1 Le diagramme de classe	13
4.3.2 Le diagramme de cas d'utilisation	14
4.3.3 Le diagramme de séquence	14
4.3.4 Le diagramme d'objet	14
4.3.5 Le diagramme d'état transition	14
4.3.6 Le diagramme d'activité	14
4.3.7 Le diagramme de collaboration	14
4.3.8 Le diagramme de composants	14
4.3.9 Le diagramme de déploiement	14
4.4. Présentation d'un paquetage (package)	14
5. Répartition du système existant en paquetages	15
6. Découverte des informations	16
7. Modélisation du système	17
7.1. Diagramme de cas d'utilisation globale	17
7.2. Diagrammes de cas d'utilisation détaillés de chaque paquetage.....	19
7.2.1 Paquetage section gestion des ressources humaines (GRH).....	19
7.2.2 Paquetage section gestion des moyens généraux (GMGX).....	21
7.2.3 Paquetage section utilisateur	22
7.2.4 Paquetage section administrateur	25

7.3. Spécification des scénarios et des cas d'utilisation	25
7.3.1 Spécification des scénarios	25
7.3.2 Spécification des cas d'utilisation	27
7.4. Diagramme de séquence des cas d'utilisation	30
7.5. Diagramme d'activité	33
7.5.1. Diagramme d'activité du cas d'utilisation « Identification »	34
7.5.2. Diagramme d'activité du cas d'utilisation « Effectuer une demande »	35
7.5.3. Diagramme d'activité du cas d'utilisation « Valider une demande »	36
Conclusion	37

Chapitre 2 : Conception

Introduction	38
1. Niveau applicatif	39
1.1 Définition du diagramme de classe	39
1.2 Diagramme de classes du cas d'utilisation « Ajouter Employé ».....	39
1.3 Diagramme de classe du cas d'utilisation « Modifier un compte utilisateur ».....	40
1.4 Diagramme de classe du cas d'utilisation « Effectuer une réservation d'hôtel »	41
1.5 Diagramme de classe du cas d'utilisation « Valider une demande »	41
1.6 Diagramme de classe du paquetage « ADMIN »	42
1.7 Diagramme de classe du paquetage « GRH »	43
1.8 Diagramme de classe du paquetage « GMGX ».....	45
1.9 Diagramme de classe du paquetage « USER »	46
1.10Le Diagramme de classe des interfaces de l'application	47
2. Niveau données	48
2.1 Dictionnaire des données.....	48
2.2 Description des classes	54
2.3 Description des associations	59
2.4 Les règles de gestion.....	62
2.5 Le diagramme de classe finale.....	63
2.6 Le modèle logique des données	64
Conclusion	66

Chapitre 3 : Réalisation

Introduction	67
1. Matériel informatique utilisé	68
2. Description de l'environnement de travail	68
2.1 Les outils utilisés	68
2.2 Les langages de programmation	70
3. Accès aux bases de données	71

3.1 Les outils nécessaires pour utiliser JDBC	71
3.2 La connexion à la base de données.....	71
4. Présentation quelques interfaces de l'application	72
5. Présentation quelques tables de l'application	75
Conclusion.....	76
Conclusion générale.....	77

INTRODUCTION GENERALE

L'entreprise du XXI^e siècle est entièrement différente de celle d'il y a quelques années, les changements que le monde a subis notamment dans le domaine de l'informatique ont fait que les fondements de l'entreprise ont évolué, que ce soit dans l'industrie ou les services. L'information n'est plus considérée comme ressource opérationnelle, mais aussi comme une ressource stratégique pour une gestion rigoureuse de l'organisme, son système d'information devient un facteur moteur lui assurant les progrès dans son environnement.

« La force d'une entreprise réside dans son travail collaboratif » c'est la valeur que défend et affiche **CEVITAL Agro-industriel**, qui cherche sans cesse à développer de nouvelles solutions pour son système d'information afin d'assurer un travail collaboratif dans l'ensemble du groupe.

Le travail collaboratif consiste à produire collectivement d'une manière organisée, où chacun participe et surtout se sente responsable. Une entreprise dite collaborative est formée par équilibre très raffiné de l'organisation, des technologies et des ressources humaines. Parallèlement, les entreprises font aujourd'hui largement appel aux technologies de l'information et de la communication (TIC) leur permettant ainsi un meilleur partage de l'information et une meilleure coordination de leurs activités.

Notre projet consiste à montrer comment une de ces technologies, le **WorkFlow**, contribue à la mise en place d'un système de gestion des flux d'information de l'entreprise « **CEVITAL Agro-industriel** » de façon plus rationnelle et plus efficace, cette technologie permet de prendre en charge les différents documents demandés par les salariés de l'entreprise auprès de l'administration du personnel.

Pour mener à bien notre projet, nous avons suivi une démarche de travail qui se base sur l'utilisation du langage de modélisation UML (Unified Modeling Language)

Nous avons défini cette démarche en faisant essentiellement référence à la méthode de Chantal MORLEY, proposée dans son livre « UML pour l'analyse d'un système d'information ».

Cette démarche est résumée comme suit :

Chapitre1 : Analyse de l'existant

Il est constitué des étapes suivantes :

- Présentation de l'organisme d'accueil.
- Repérage du domaine d'étude.
- Présentation de la technologie *WorkFlow*.
- Présentation de la méthode de conception.

- Répartition du système existant en paquetages.
- Découverte des informations.
- Modélisation du système.

Chapitre2 : Conception

Il est constitué de deux niveaux qui sont :

1. **Niveau applicatif** : Le travail de ce niveau, consiste à exprimer l'aspect réseau de l'application, il permet de présenter les relations des pages de l'application avec les éléments d'arrière-plan du système (Base de données).
2. **Niveau données** : Le travail de ce niveau, consiste à décrire la base de données de l'application à développer. Pour ce faire, il faut :
 - Définir un dictionnaire de données.
 - Définir les réglés de gestion du système d'information.
 - Décrire le diagramme de classe des différentes entités du système
 - Décrire le modèle logique des données, en utilisant le modèle relationnel.

Chapitre 3 : Réalisation

Dans ce chapitre nous allons présenter :

- L'environnement technique de développement
- Les tables de la base de données
- Quelques interfaces de l'application

Chapitre 1 :
Analyse de l'existant

Introduction :

Ce chapitre est divisé en deux parties, la première est consacrée à la présentation générale de **CEVITAL Agro-industriel**, à travers son historique, son évolution ainsi que son organisation.

La deuxième partie correspond à la prise de connaissance de notre domaine d'étude. Comme la gestion est la même pour toutes les unités de **CEVITAL Agro-industriel**, on se limite à l'étude du système d'information de l'Unité de traitement d'eau minérale **LALLA KHEDIDJA**, par la présentation de la hiérarchie de ses responsabilités, en utilisant un organigramme.

1. Présentation de CEVITAL agro-industriel [2]:

1.1 Historique :

CEVITAL est un groupe familial créé par M^r Issad Rebrab et fils, c'est la première entreprise algérienne à avoir investi dans des secteurs d'activités diversifiés, pour atteindre sa taille et sa notoriété actuelle l'entreprise a traversé d'importantes étapes historiques qui sont :

- 1971 : Lancement de la construction métallique.
- 1988 : Création de META SIDER (SIDERURGIE)
- 1991 : Reprise des activités I.B.M en Algérie /Création du quotidien Liberté
- 1997 : Création de Hyundai Motorsen Algerie
- 1998 : Création de Cevital SPA industrie agroalimentaires
- 2006 : Création de Numidis et Immobis ; acquisition de COJEK
- 2007 : SAMHA- Production et distribution SAMSUNG /Création MFG (verre plat)
- 2008 : Nolis-Transport maritime/commercialisation du verre plat en Europe/ Création de Numilog
- 2009 : Augmentation de la production de sucre
- 2013 : OXXO (Italie/ ALAS (Italie))
- 2014 : BRANDT (Italie)/AFFERPI(Italie) ex LUCCHINI PIOMBINO

1.2 Présentation de la direction générale de CEVITAL Agro-industriel :

CEVITAL Agro-industriel est leader du secteur agro-alimentaire et aussi le plus grand complexe privé en Algérie, La société **CEVITAL** est portée par 18000 collaborateurs dont 15000 en Algérie. Depuis ses débuts le groupe est installé au sein du port de Bejaia où se trouve la direction générale.

Depuis la dernière restructuration du schéma organisationnel et fonctionnel de **CEVITAL** sa direction générale est organisée comme suit :

- **Direction ressources humaines** : pilotée par le directeur des ressources humaines, cette direction remplit au sein de l'entreprise une triple mission :
 - ✓ Assurer l'interface avec la direction générale.
 - ✓ Jouer un rôle de conseil auprès des cadres des directeurs opérationnels.
 - ✓ Prendre en charge l'impulsion, le pilotage, et le suivi des équipes et projets.
- **Direction système d'information** : cette direction maintient la cohérence de l'infrastructure informatique avec les besoins de l'entreprise, cette tâche recouvre tout autant les préoccupations de continuité de service ainsi que l'alignement des technologies de l'information avec les objectifs stratégiques de l'entreprise.
- **Direction finance et comptabilité** : chargée de l'assistance à la direction générale en matière de finance et comptabilité, par la fourniture de toutes les données nécessaires à la prise de décisions. Comme elle assure l'animation, la coordination, et le contrôle de l'ensemble des activités financières et comptables de l'entreprise.

- **Direction commerciale** : réunit les cadres de direction qui sont chargés de définir la place des canaux de vente et de piloter et manager les équipes afin de développer le chiffre d'affaire de l'entreprise et sa marge.
- **Direction gestion des stocks** : Chargée de l'analyse des situations des stocks, de l'achat des produits chez les fournisseurs ou sous traitements dans un souci d'économie et dans un respect de la qualité. Cette direction intervient sur tout le périmètre lié aux approvisionnements de la société.
- **Direction approvisionnement** : Cette direction est chargée de la planification des besoins, des appels aux marchés, de la gestion des commandes, de la gestion contractuelle selon les lois et règlements en vigueur.
- **Direction exploitation** : Assure la direction des périmètres de production, décline la stratégie de l'entreprise au niveau des opérations de production et de services, comme elle gère aussi la productivité et la rentabilité des structures qui lui sont rattachées.

Les 7 directions présentées sont organisées autour des unités suivantes :

- **Usine raffinerie huile** : Située à Bejaia, Spécialisée dans le raffinage de la matière première de l'huile.
- **Usine conditionnement huile** : Située à Bejaia, Spécialisée dans le conditionnement de l'huile dans des bouteilles.
- **Usine raffinerie sucre** : Située à Bejaia, Spécialisée dans le raffinage de la matière première du sucre.
- **Usine conditionnement sucre** : Située à Bejaia, Spécialisée dans le conditionnement du sucre.
- **Usine margarinerie** : Située à Bejaia, Spécialisée dans la production et le conditionnement de la margarine (Matina, Fleurial, Smen...).
- **Usine céréales** : Située à Bejaia, Spécialisée dans le conditionnement des céréales.
- **Usine cojake** : Située à Elkseur, Spécialisée dans la production du jus Tchina, de la confiture, de la tomate...
- **Usine four à chaux et CO₂** : Située à Constantine, Spécialisée dans la production des chaux et du CO₂
- **Usine lalla khedidja** : Située à Tizi-Ouzou, Spécialisée dans la production d'eau minérale.
- **Usine plasturgie** : Située à Tizi-Ouzou, Spécialisée dans la production de barquettes (Margarine, Smen, ...) et la production préforme de bouteilles d'huiles.

1.3L'organigramme de CEVITAL Agro-industriel [2]:

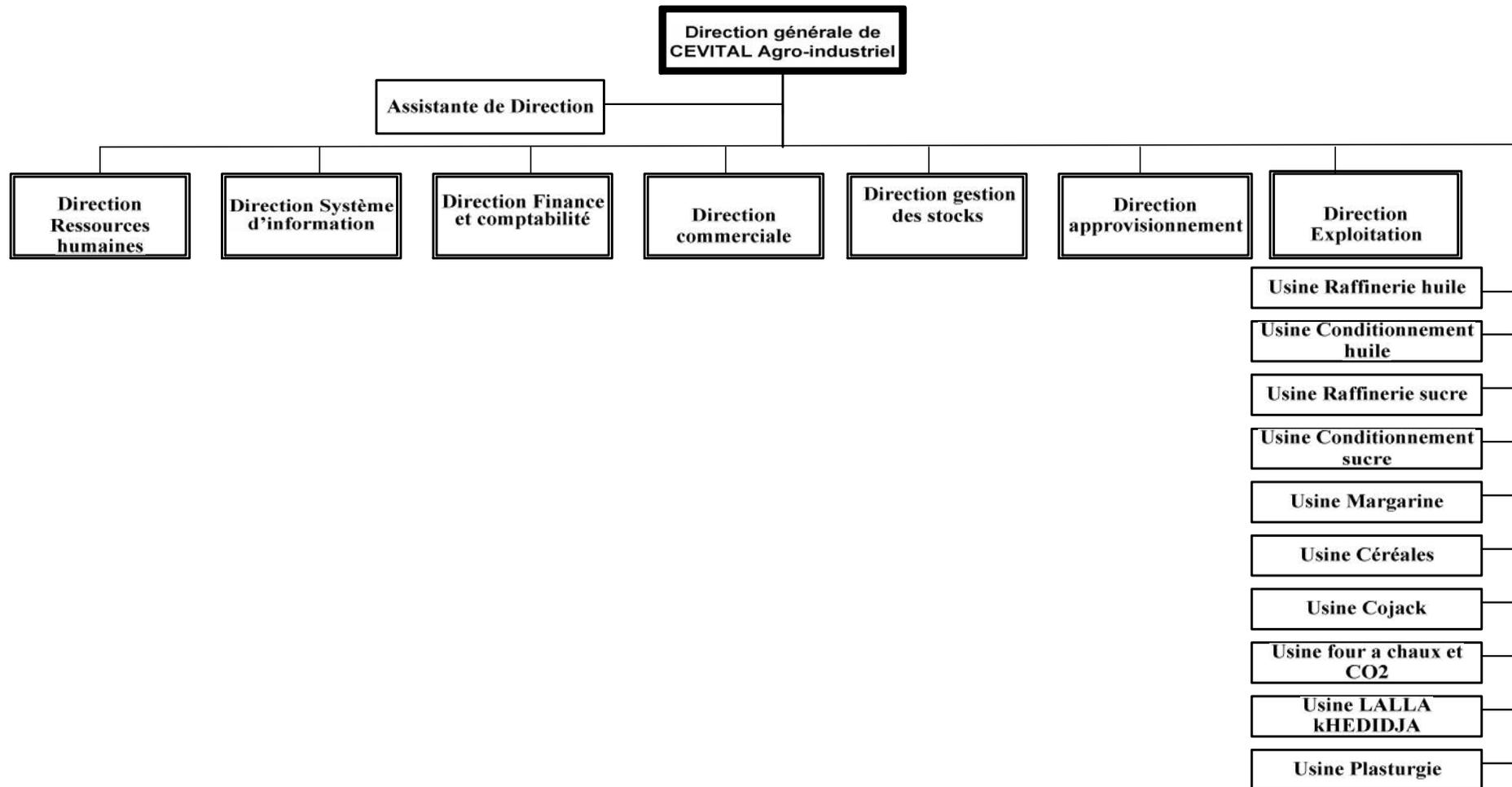


Figure1 : « Organigramme de CEVITAL Agro-industriel ».

2 Repérage du domaine d'étude :

2.1. Présentation de LALLA KHEDIDJA [3]:

L'unité d'eau minérale « **LALLA KHEDIDJA** » du groupe **CEVITAL Agro-industriel** située au pied du mont Djurdjura dans la commune d'Agouni Gueghrane, à environ 35 Kms au sud-ouest du chef-lieu de la wilaya de TIZI OUZOU, puise son eau de la source Thinzet située au flanc du mont Kouriet.

L'eau de « **LALLA KHEDIDJA** » prend son origine au plus haut sommet du Djurdjura, Pour parler de ses caractéristiques, il s'agit d'une eau oligominérale non gazeuse, riche en minéraux essentiels à la vie, réputée pour sa légèreté et sa pureté. C'est une eau de montagne dont le parcours géologique est protégé contre toute pollution. Elle est ainsi directement embouteillée sans subir aucun traitement chimique. En juillet 2007 la célèbre eau minérale prend sa place sur le marché.

En plus de deux lignes de conditionnement destinées pour l'eau minérale, l'unité dispose d'une troisième ligne destinée à la production de diverses boissons non alcoolisées.

2.2 Présentation de l'organigramme de LALLA KHEDIDJA [3]:

Vu que notre projet consiste à informatiser le système d'information du groupe **CEVITAL Agro-industriel** en développant une application *WorkFlow*, et comme on s'est limité à l'étude de l'unité **LALLA KHEDIDJA**, il est inéluctable de présenter d'une manière approfondie l'organisation de sa direction générale.

Cette direction est organisée autour des structures suivantes :

- Chef de département production.
- Chef de département maintenance.
- Chef de département logistique.
- Responsable de l'hygiène, Sécurité et environnement(HSE).
- Responsable de la qualité.
- Responsable de l'administration du personnel.
- Administrateur système d'information.
- Assistante de direction.
- Contrôleurs de gestion.

Ces structures sont organisées comme suit :

- **Chef de département production :**
 - Un méthodiste production.
 - Un chef de quart.
 - Un chef de ligne.
 - Des opérateurs.
 - Des caristes.
 - Des manutentionnaires.
- **Chef de département maintenance :**
 - Méthodiste maintenance.
 - Des maintenanciers.
 - Un agent technique.
- **Chef de département logistique :**
 - Un responsable expédition.
 - Un chargé des moyens généraux.
 - Un gestionnaire des stocks.
 - Un responsable approvisionnement.
- **Responsable hygiène, sécurité et environnement (HSE):**
 - Superviseur hygiène, sécurité et environnement (HSE).
 - Chef de quart HSE.
- **Responsable de qualité :**
 - Contrôleur de qualité.
- **Responsable de l'administration du personnel :**
 - Gestion des ressources humaines.
 - Standardiste.
- **Administrateur système d'information**

2.3 L'Organigramme de LALLA KHEDIDJA [3]:

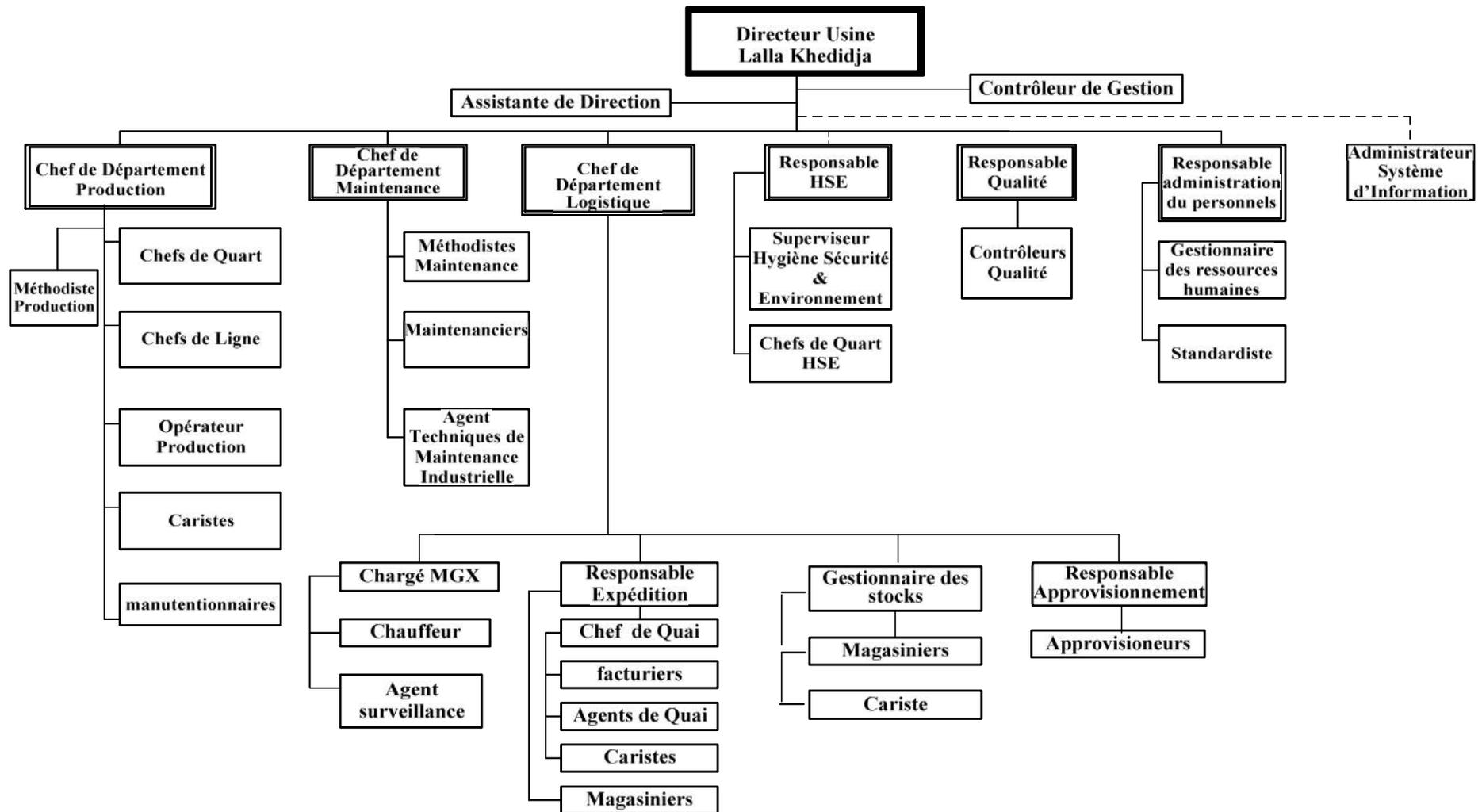


Figure 2 : « Organigramme de LALLA KHEDIDJA ».

3. Présentation de la technologie *WORKFLOW* [4] :

3.1 Définition du *WORKFLOW*:

Un workflow traduit littéralement « flux de travail » est défini par la WFMC (Workflow Management Coalition : association de plus de 300 fabricants, des utilisateurs, des consultants et des chercheurs dans le domaine de la [gestion des flux](#) de [travail](#) .) par : « Automatisation de tout ou une partie d'un processus d'entreprise, au cours duquel l'information circule d'une activité à l'autre, c'est-à-dire d'un participant à l'autre, en fonction d'un ensemble de règles de gestion ».

Le Workflow permet de coordonner et contrôler un processus d'entreprise. Impliquant des tâches humaines et automatisées dans un environnement organisationnel distribué et informatisé.

3.2. Utilité du *WorkFlow* :

A l'heure actuelle, les entreprises entrent dans l'ère de business et pour rester compétitives, elles doivent améliorer constamment leur efficacité opérationnelle.

Par conséquent les technologies de l'information (TIC) et plus particulièrement internet, sont devenus un vecteur de communication.

De plus, harmoniser le travail des hommes dans un système d'information global est devenu une priorité.

Face à ces enjeux, le *WorkFlow* apporte la solution efficace pour :

- Modéliser les procédures de travail
- Contrôler et suivre l'avancement des projets
- Automatiser la circulation des documents
- Impliquer les partenaires dans les procédures
- Mesurer les coûts

3.3. L'accès à l'information dans un *WorkFlow* :

Dans un *WorkFlow* il existe deux façons d'accéder à l'information

➤ **En cherchant l'information dans une file d'attente partagée :**

Il s'agit d'une invocation de type base de données, une file d'attente étant gérée comme une table à laquelle accèdent les différents acteurs d'une procédure. L'avantage de cette méthode d'invocation est la possibilité de faire partager un ensemble de tâches par un groupe d'utilisateurs. En effet, lorsqu'une tâche est affectée à un profil donné, on ne va pas forcément l'affecter à une personne donnée. On mettra l'ensemble des tâches à disposition de toutes les personnes correspondantes au profil, et on aura ainsi une régulation de la charge qui pourra se faire au mieux.

➤ En recevant l'information dans sa boîte de réception personnelle :

Il s'agit d'une invocation de type messagerie, puisque le meilleur moyen d'envoyer une tâche à exécuter à une personne est de la lui notifier à travers sa messagerie, et donc à travers sa boîte de réception. On utilise ici une infrastructure existante, et un mode de communication standard par rapport aux habitudes d'un utilisateur.

Les produits de *WorkFlow* fonctionnant sur ce principe sont donc plus légers, et l'utilisateur final à l'impression de traiter ses tâches tout comme il traite son courrier électronique. Il ne va pas chercher spécifiquement les tâches qu'il a à effectuer, puisque les notifications arrivent directement via sa messagerie sur son poste de travail.

Classiquement la plupart des produits de *WorkFlow* se répartiront ainsi entre les deux catégories. En terme de performances, les produits s'appuyant sur l'infrastructure de messagerie sont beaucoup plus légers, donc plus faciles à mettre en œuvre, et générant en particulier des charges réseau moins importantes, puisqu'ils s'intègrent dans une infrastructure existante. Certains n'utilisent pas du tout de base de données, et se contentent de fournir des pilotes leurs permettant de se connecter aux bases existantes dans l'entreprise.

3.4. Types d'applications *WorkFlow* :

Il existe plusieurs types d'applications *WorkFlow* :

- **Le *WorkFlow* de production** : il s'appelle aussi *WorkFlow* procédural ou *WorkFlow* directif, il correspond à la gestion des processus de base de l'entreprise.
- **Le *WorkFlow* administratif** : il correspond à tout ce qui est routage de formulaires, basé en générale sur une infrastructure de messagerie.
- **Le *WorkFlow* ad-hoc** : Ce types de *WorkFlow* est utilisé pour la gestion des procédures non déterminées ou mouvantes.
- **Le *WorkFlow* coopératif** : il gère des procédures évoluant assez fréquemment et liées à un groupe de travail restreint dans l'entreprise.

3.5. Domaines d'application des *WorkFlows* :

Il y a plusieurs domaines dans l'entreprise qui peuvent être concernés par une application *WorkFlow* dont on peut citer :

- L'organisation
- La collaboration
- La production
- La gestion des ressources humaines
- La gestion des relations client...

Notre application *WorkFlow* est Adaptée de plus en plus aux procédures bien définies à savoir :

- Gérer les demandes de congé (annuel, récupération, sans solde, anticipées).
- Gérer les demandes d'absence autorisée.
- Gérer les demandes de pointage manuel.
- Gérer les demandes de visite médicale.
- Gérer les demandes d'heures supplémentaires.
- Gérer les demandes d'ordre de mission.
- Gérer les demandes d'attestation de travail.
- Gérer les demandes de formation.
- Gérer les demandes d'avance sur salaire.

3.6 Avantages d'une application *WorkFlow* :

- 1. Amélioration de la productivité :** gains de productivité réalisés par une application de *WorkFlow* sont de 20 à 50% sur la part des tâches qu'elle automatise.
- 2. Temps de réponse réduit :** la réduction du temps de réponse entre la prise en compte de l'événement déclencheur d'un cas et son traitement complet est une des caractéristiques essentielles des applications de *WorkFlow*.
- 3. Information claire sur l'état d'avancement :** l'application de *WorkFlow* permet de renseigner avec précision sur l'état d'un cas.
- 4 Sécurité accrue :** avec une application de *WorkFlow*, c'est le système qui affecte les tâches aux participants.
- 5 Maîtrise de la qualité et des coûts :** une application de *WorkFlow* enregistre systématiquement le journal de tous les événements qu'elle contrôle, avec la date et l'heure, la procédure et la tâche concernée et le participant actif. Des outils de traitement de ce journal fournissent des rapports de synthèse sur les coûts et délais de traitement des tâches et des procédures.

4. Présentation du langage de conception :

4.1. Choix de la méthode de conception :

Pour concevoir le système d'information, nous avons opté pour le langage UML pour une multitude de raisons :

- La volonté et la curiosité de découvrir une méthode de conception orientée objet.
- UML est riche, couvre toutes les phases d'un cycle de développement.
- UML est ouvert, il est indépendant du domaine d'application et des langages d'implémentation.
- Sa notation graphique permet d'exprimer visuellement une solution objet.
- L'aspect formel de sa notation limite les ambiguïtés et les incompréhensions.
- Son aspect visuel facilite la comparaison et l'évaluation de solutions.
- Il facilite la compréhension des représentations abstraites complexes.
- Son caractère polyvalent et sa souplesse en font un langage universel.

4.2. Présentation du langage UML [1] :

UML est l'acronyme de « Unified Modeling Language » qui peut se traduire par « langage unifié de modélisation ».

UML est le résultat de la fusion de trois méthodes à savoir : OOD « Object Oriented Design, OMT « Object Modeling Technique » et OOSE « Object Oriented Software Engineering ».

UML n'impose ni façon d'utiliser ces modèles, ni processus de développement. Aussi, il rencontre un intérêt croissant auprès des développeurs, notamment lorsqu'ils disposent d'un outil permettant de générer du code Java, C++ ou Visual Basic à partir de diagrammes UML.

UML est un langage de modélisation Objet, il se base sur une sémantique précise et sur une notation graphique expressive qui s'appuie sur neuf diagrammes non forcément utilisés à la fois, tout dépend de la nature du projet.

Les diagrammes UML permettent de communiquer l'architecture d'un système aux concepteurs, développeurs et à toute personne externe à l'aspect technique d'un système.

4.3. Définition des 9 diagrammes du langage UML [1]:

4.3.1 Le diagramme de classe :

Le diagramme de classe exprime de manière générale la structure statique d'un système, il contient des classes ainsi que leurs associations. L'intérêt majeur de ce diagramme est de représenter les entités du système d'information.

4.3.2 Le diagramme de cas d'utilisation :

Du point de vu utilisateur, le diagramme de cas d'utilisation définit les limites du système et les relations entre le système et l'environnement.

Un cas d'utilisation est une façon d'utiliser le système, l'intérêt majeur de ce diagramme est de représenter les fonctionnalités du système selon les besoins de l'utilisateur.

4.3.3 Le diagramme de séquence :

Le diagramme de séquence montre des interactions entre objet selon un point de vue temporel, le contexte des objets n'est pas représenté de manière explicite. La représentation se concentre sur l'expression des interactions. Généralement ce diagramme définit les objets acteurs et le système.

4.3.4 Le diagramme d'objet :

Le diagramme d'objet permet de mettre en évidence des liens entre les objets, instances de classe reliée par une instance d'association. Le diagramme d'objet utilise les mêmes concepts que le diagramme de classe, il est utilisé pour illustrer des parties complexes d'un diagramme de classe.

4.3.5 Le diagramme d'état transition :

Le diagramme d'état transition a pour objectif de représenter des traitements permettant de gérer le domaine étudié. Ce diagramme met en évidence l'enchaînement des différents états d'une classe.

4.3.6 Le diagramme d'activité :

Le diagramme d'activité permet de représenter la dynamique du système, on peut le considérer comme une variante du diagramme d'état transition. Il est attaché à une classe, un processus, un acteur ou une entité, à un cas d'utilisation ou bien à une opération.

4.3.7 Le diagramme de collaboration :

Le diagramme de collaboration met en place les interactions entre les différents objets du système étudié et les messages qu'ils s'échangent. Dans le cadre de l'analyse, ce diagramme est utilisé pour faire apparaître les échanges d'informations entre le système étudié et les autres systèmes.

4.3.8 Le diagramme de composants :

Le diagramme de composants décrit les éléments physiques et leurs relations dans l'environnement de réalisation, il montre les choix de réalisation.

4.3.9 Le diagramme de déploiement :

Ces diagrammes montrent la disposition des différents matériels, (les nœuds) qui entrent dans la composition d'un système dans la répartition des programmes exécutables sur ces matériels.

4.4. Présentation d'un paquetage (package) :

Le paquetage offre un mécanisme pour la division et l'organisation des modèles, ainsi que le groupement des éléments de modélisation. Il contient, selon le modèle des classes, des objets, des composants, ainsi que les diagrammes associés. Les paquetages sont utilisés pour faciliter la modélisation.

5. Répartition du système existant en paquetages :

Pour simplifier la modélisation tout en gardant la cohérence de l'ensemble du système, nous avons décomposé le système existant en paquetages. Chaque paquetage est structuré de plusieurs cas d'utilisation.

Après la phase repérage du domaine d'étude, les principales sections du système de gestion de l'unité « LALLA KHEDIDJA » ont été identifiées, ainsi que leurs acteurs qui seront les futurs utilisateurs de l'application *WorkFlow* à développer.

Ces sections sont :

Section	Acteur	Mnémonique
Section gestion des ressources humaines	Agent de la section de gestion des ressources humaines	AGRH
Section gestions des moyens généraux	Agent de la section de gestion des moyens généraux	AGMGX
Section utilisateur	L'utilisateur	USER

En plus de ces trois paquetages principaux, il est indiscutable d'ajouter le paquetage Administrateur (acteur Administrateur), son rôle consiste à gérer les comptes utilisateurs, ainsi que les circuits de validation de l'application à développer et l'entretien du système.

Ces différents paquetages sont représentés dans le diagramme de contexte suivant :

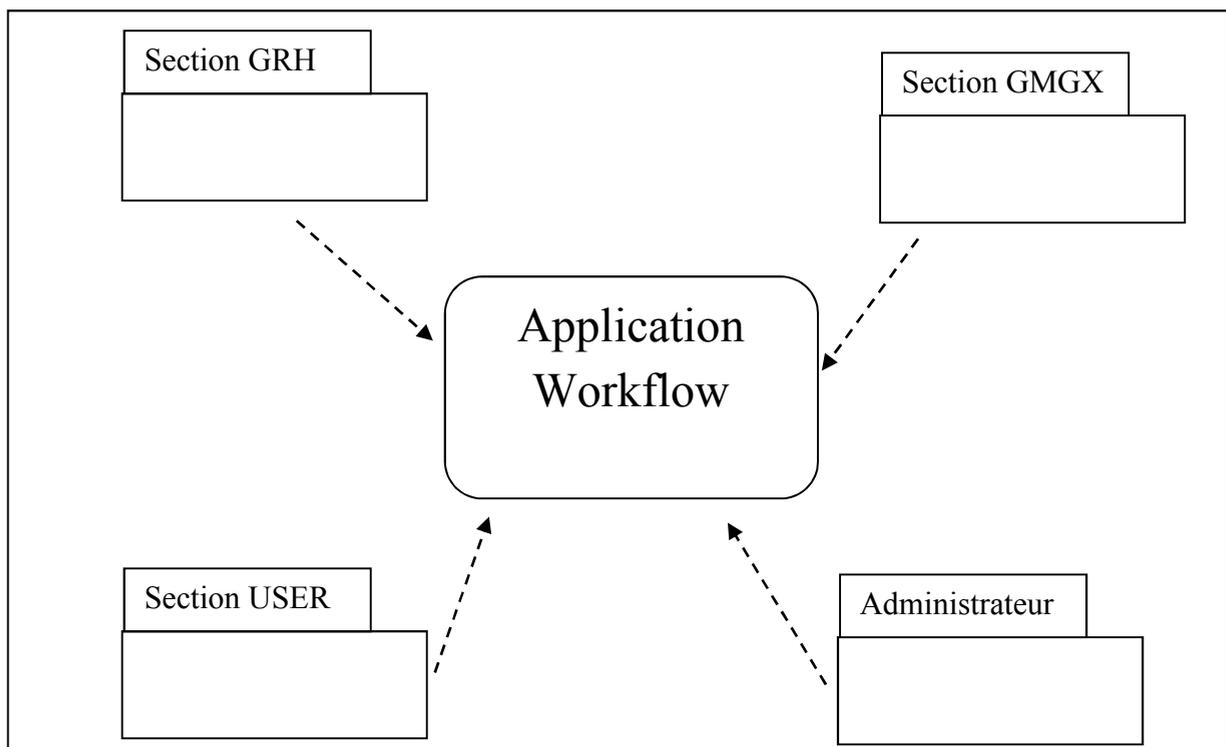


Figure 3 : « Diagramme de contexte de l'application ».

6. Découverte des informations [3]:

Dans cette phase nous avons interviewé de façon plus approfondie les acteurs des différents services concernés par l'étude, à savoir : le chef du département production, le chef du département maintenance, le chef du département logistique, le responsable HSE, le responsable qualité, le responsable administration du personnel et l'administrateur du système d'information.

Ensuite, les différents documents opérationnels demandés par les salariés de l'unité LALLA KHEDIDJA ont été analysés.

Les interviews réalisées et les analyses des différents documents ont permis d'assembler des informations sur les procédures de gestion de l'unité LALLA KHEDIDJA, afin de pouvoir comprendre le système d'information existant afin de récupérer les concepts d'informations qui vont gérer notre application *WorkFlow*, qui sont les suivants :

1. **Demande d'attestation de travail** : Chaque employé a le droit de faire une demande d'attestation de travail une fois par mois.
2. **Demande d'ordre de mission** : Si un employé fait une demande d'ordre de mission, le service de gestion des moyens généraux va prendre en charge les réservations d'hôtel, les réservations de véhicules avec ou sans chauffeur ainsi que les réservations de billets d'avion ou de train.
3. **Demande de congé annuel** : Si un employé fait une demande de congé annuel, le nombre de jours demandés doit être inférieur ou égale au solde des congés annuels. Chaque congé pris sera débité du solde des congés de l'employé et la prise en charge des nouveaux congés se fait à partir du premier juillet de chaque année.
4. **Demande de congé de récupération** : Si un employé fait une demande de congé de récupération, le nombre de jours demandé doit être inférieur ou égale au solde des congés de récupération ainsi que la journée de récupération doit être différente du jeudi et dimanche. Chaque jour de récupération pris sera débité du solde des récupérations.
5. **Demande de congé sans solde** : L'employé a le droit à un congé sans solde à l'acceptation de son responsable et une fois que le congé anticipé est consommé. La durée du congé sans solde est entre 1 et 12 mois.
6. **Demande de congé anticipé** : Si le nombre de jours demandé est inférieur au solde des congés annuel, l'employé a la possibilité de demander un congé anticipé qui sera débité de son solde des congés annuel après sa mise à jour à partir du premier juillet de l'année en cours.
7. **Demande d'absence autorisé** : l'employé a le droit à un congé de 3 jours dans les cas suivants : mariage de l'employé, naissance d'un enfant de l'employé, mariage d'un enfant de l'employé, décès, circoncision d'un enfant de l'employé.
8. **Demande de formation** : l'employé a le droit à une formation à l'acceptation de son responsable.
9. **Demande de pointage manuel** : L'employé a le droit de faire une demande de pointage manuel une fois par mois.

10. **Demande d'avance sur salaire** : l'employé a le droit de demander une avance sur salaire à n'importe quel moment, mais une fois le virement est fait la date du prochain salaire sera mise à jour.
11. **Demande d'effectuer des heures supplémentaires** : Le nombre d'heures supplémentaires demandées par l'employé ne doit pas dépasser 32 heures par mois.
12. **Demande de visite médicale** : L'employé peut effectuer une demande de visite médicale qui sera inscrite dans la liste des patients du médecin chargé des employés de l'unité LALLA KHEDIDJA.

7. Modélisation du système :

L'objectif des phases précédentes y compris cette phase est de comprendre le fonctionnement du système existant, mais l'objectif le plus précis de cette dernière est d'identifier le rôle des différents acteurs du domaine d'étude et la façon dont ils jouent ensemble pour atteindre les finalités de ce domaine.

Pour ce faire nous avons utilisé trois diagrammes UML, à savoir, le diagramme de cas d'utilisation, le diagramme de séquence et le diagramme d'activité.

❖ Diagramme de cas d'utilisation :

Les cas d'utilisation sont une technique de description du système étudié privilégiant le point de vue de l'utilisateur. Un cas d'utilisation est une façon spécifique d'utiliser le système. Il est composé d'un ensemble d'actions déclenchées par un acteur et qui produit un résultat identifiable.

Ce regroupement d'actions ne correspond pas à une structure implémentée ou à implémenter directement du système. Il correspond plutôt à une séquence perçue comme logique par un utilisateur. Un cas d'utilisation représente la relation entre un acteur et une fonctionnalité du système. Dans la modélisation d'un système d'information, on considère que les processus sont des cas d'utilisation.

Les cas d'utilisation sont recensés dans des diagrammes « **les diagrammes de cas d'utilisation** » qui montrent les liens entre les cas d'utilisation et les acteurs.

Chaque cas est ensuite décrit par une définition générique (le cas d'utilisation) et des définitions illustrées (les scénarios).

7.1. Diagramme de cas d'utilisation globale :

La figure 4 montre le diagramme global des cas d'utilisation de l'application *WorkFlow*.

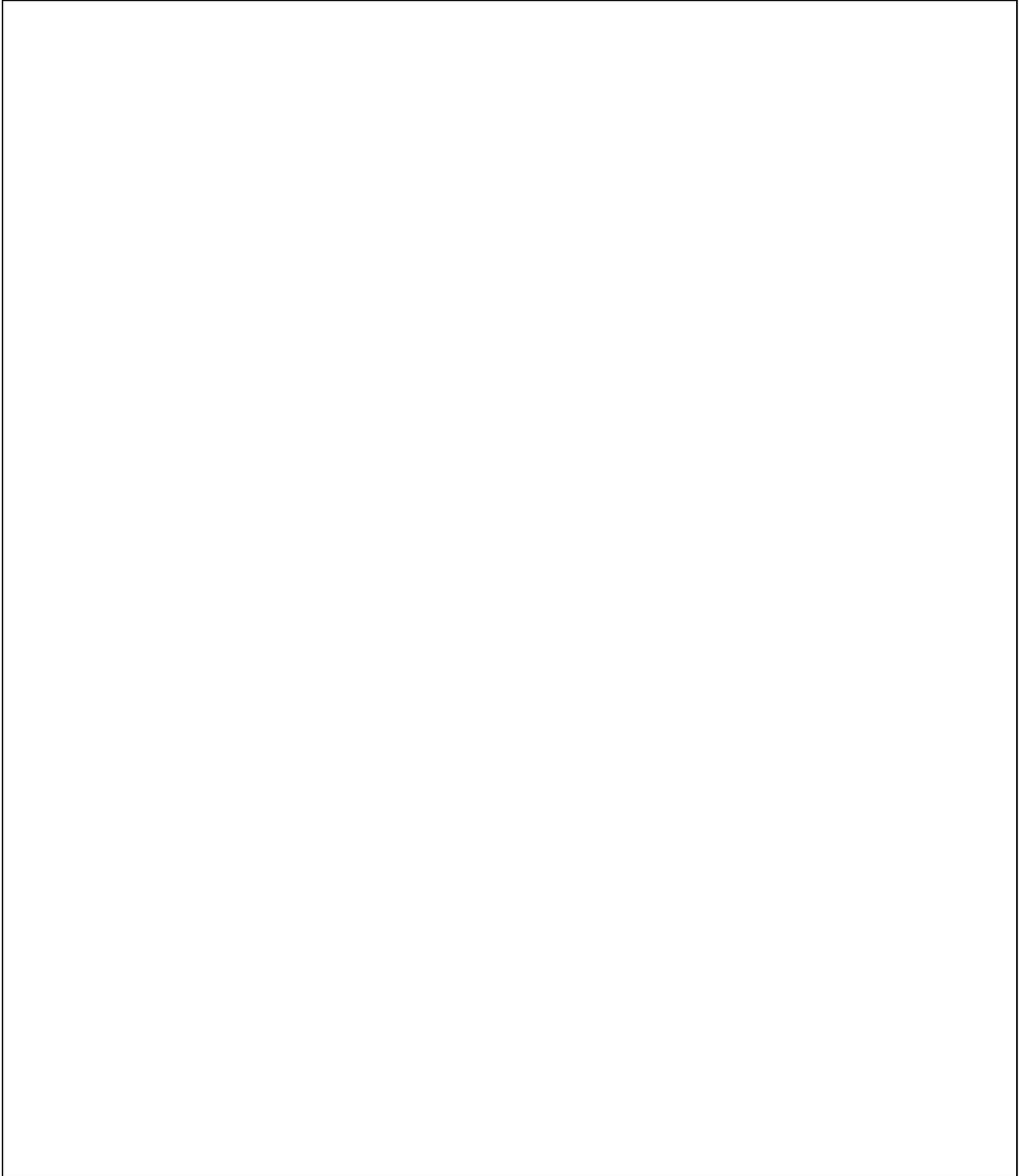


Figure4 : « Diagramme de cas d'utilisation global ».

7.2. Diagrammes de cas d'utilisation détaillés de chaque paquetage :

7.2.1 : Paquetage section gestion des ressources humaines (GRH):

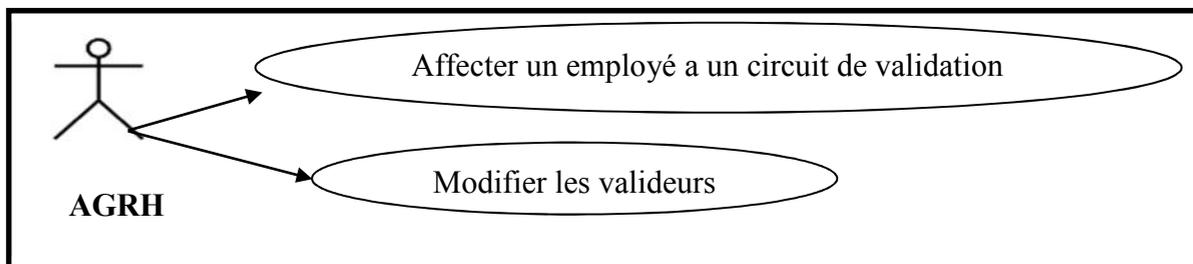


Figure 5 : Diagramme de cas d'utilisation « gérer les circuits de validation »

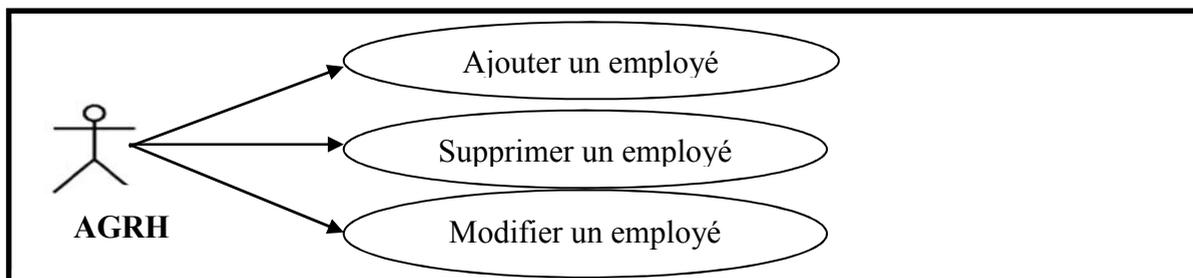


Figure 6 : Diagramme de cas d'utilisation « gérer les employés »

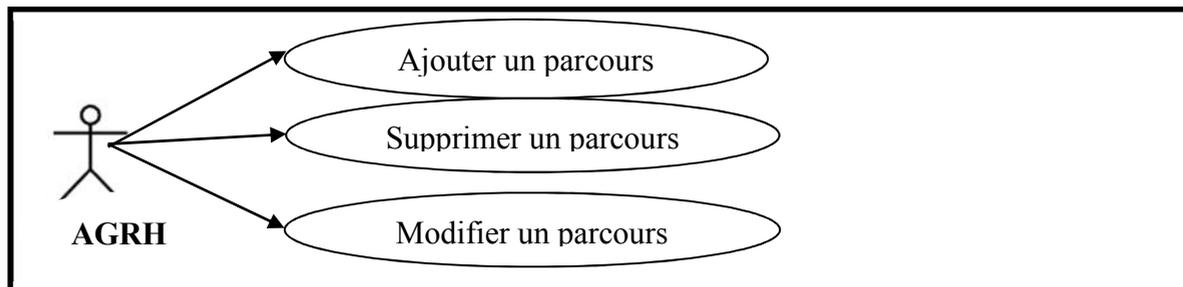


Figure 7 : Diagramme de cas d'utilisation « gérer les parcours »

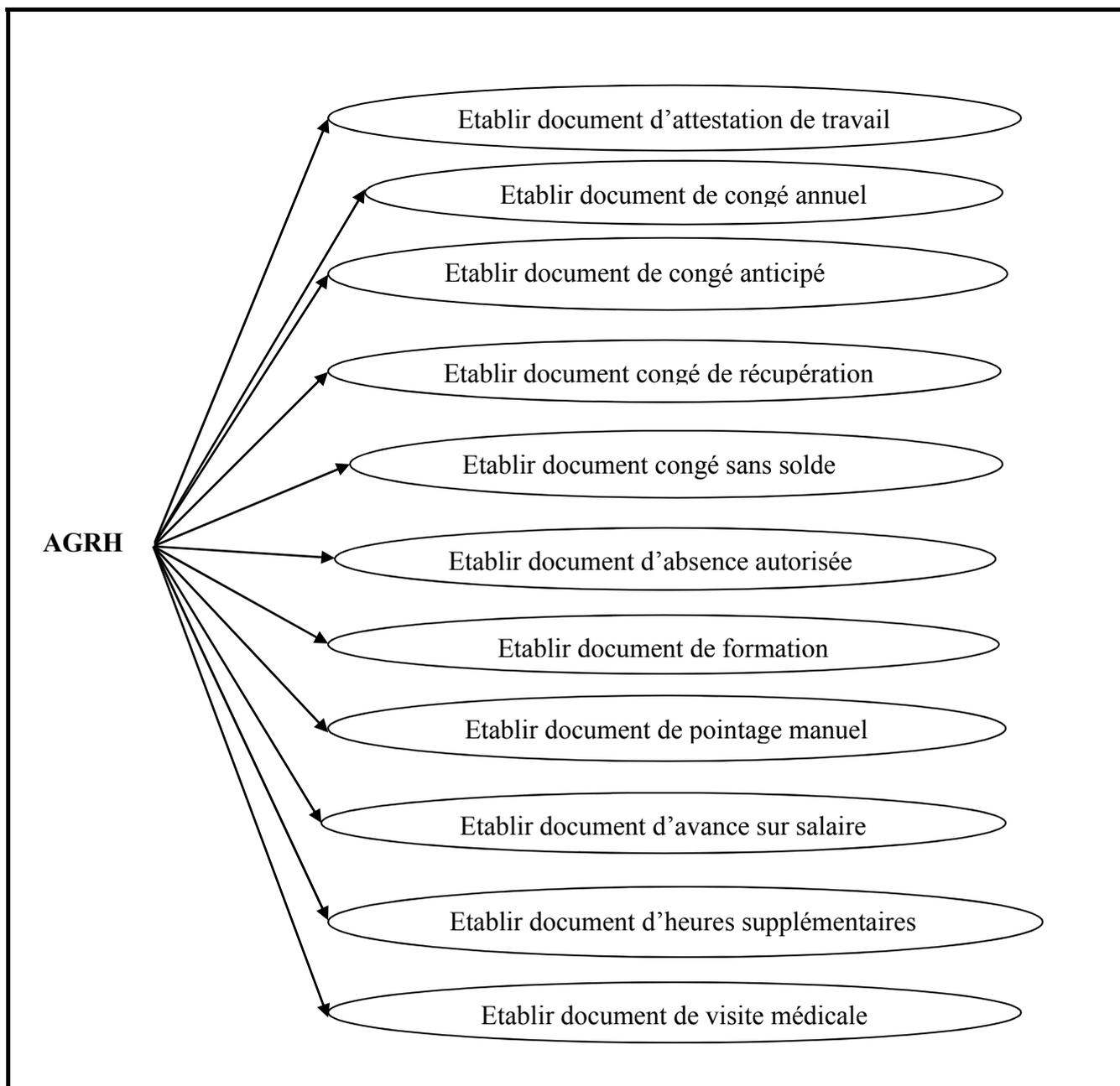


Figure 8 : Diagramme de cas d'utilisation « établir les documents ».

7.2.2. Paquetage section gestion des moyens généraux (GMGX):

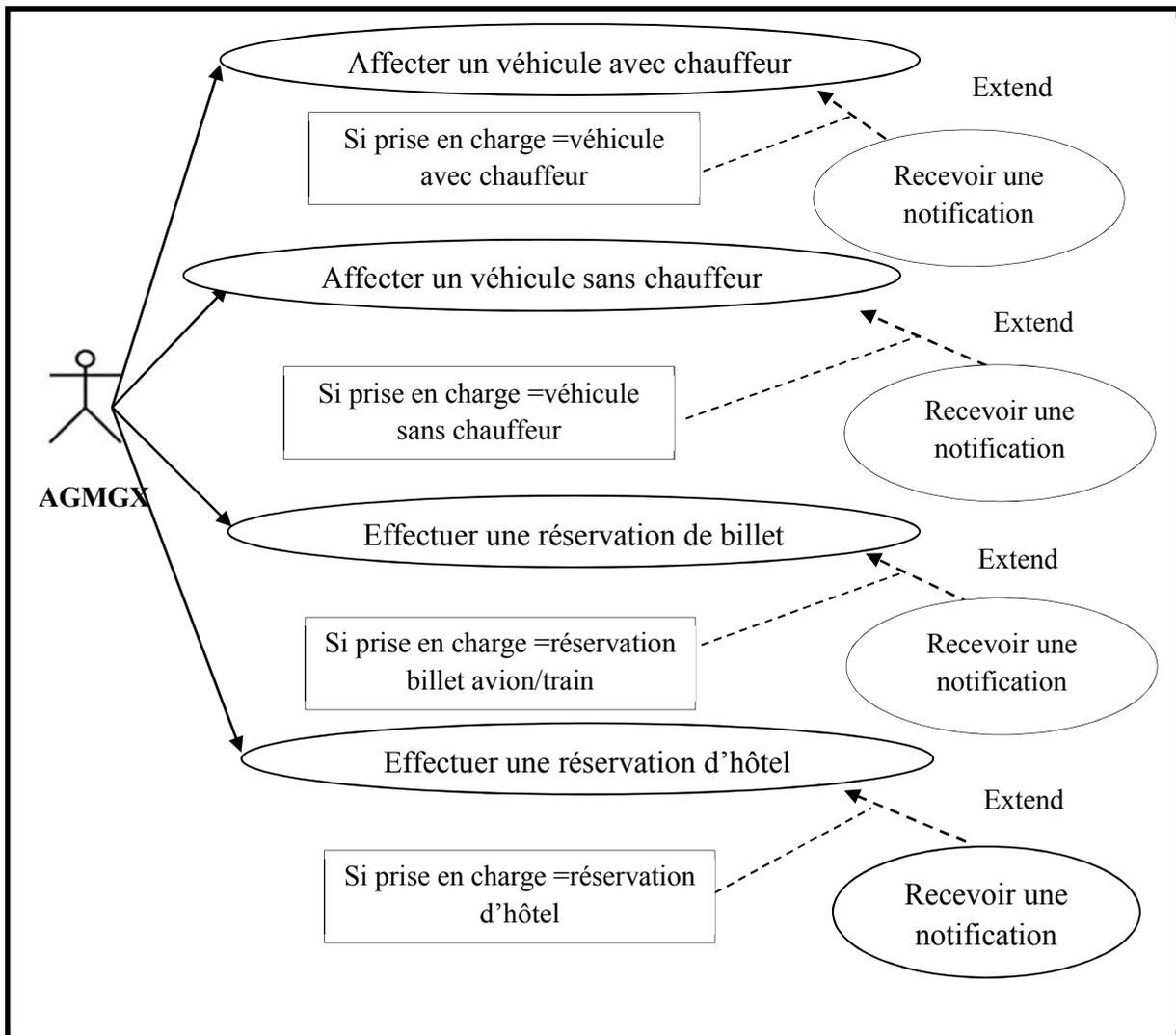


Figure 9 : Diagramme des cas d'utilisation « gestion des moyens de transport / gestion des réservations d'hôtel ».

7.2.3. Paquetage section utilisateur :

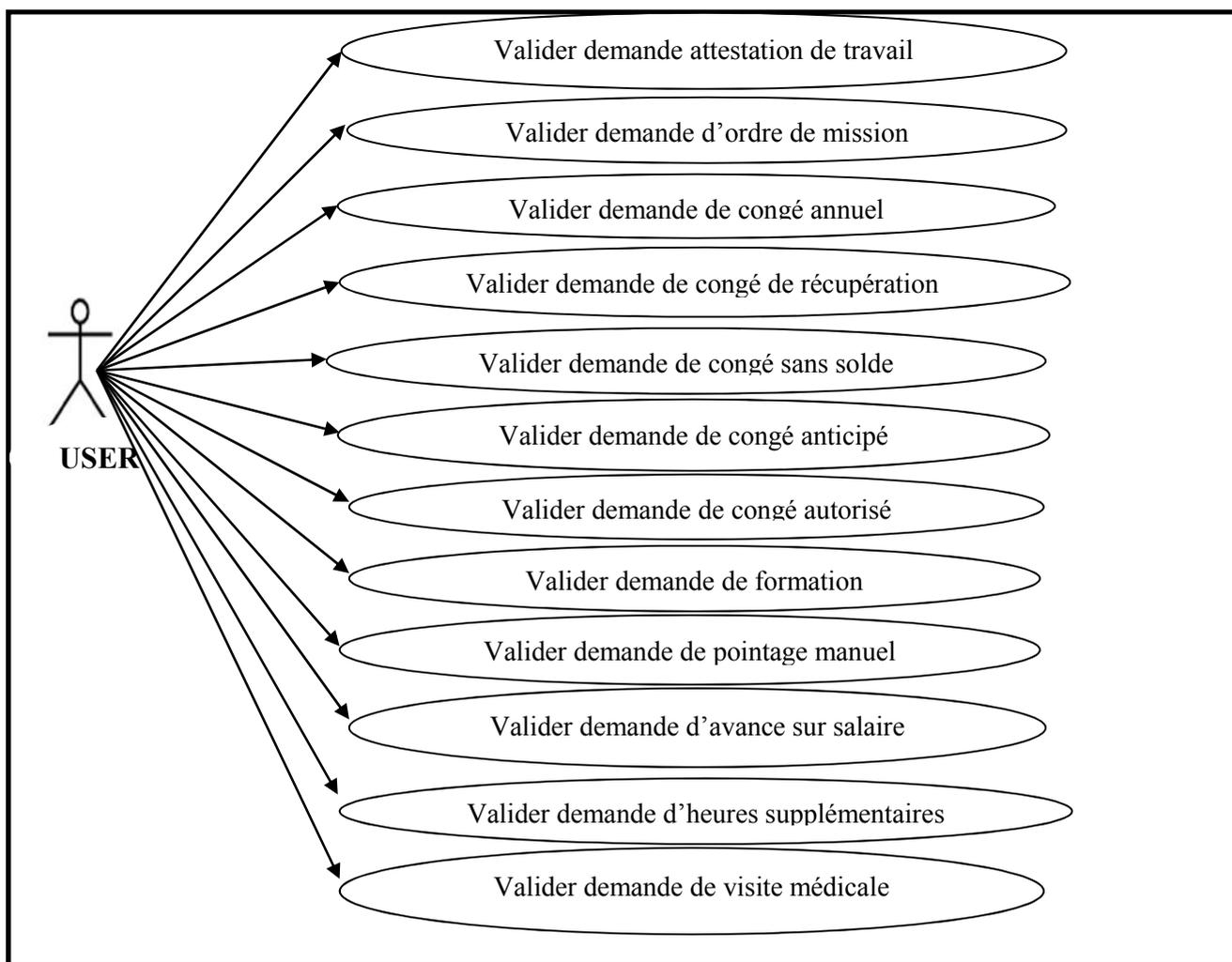
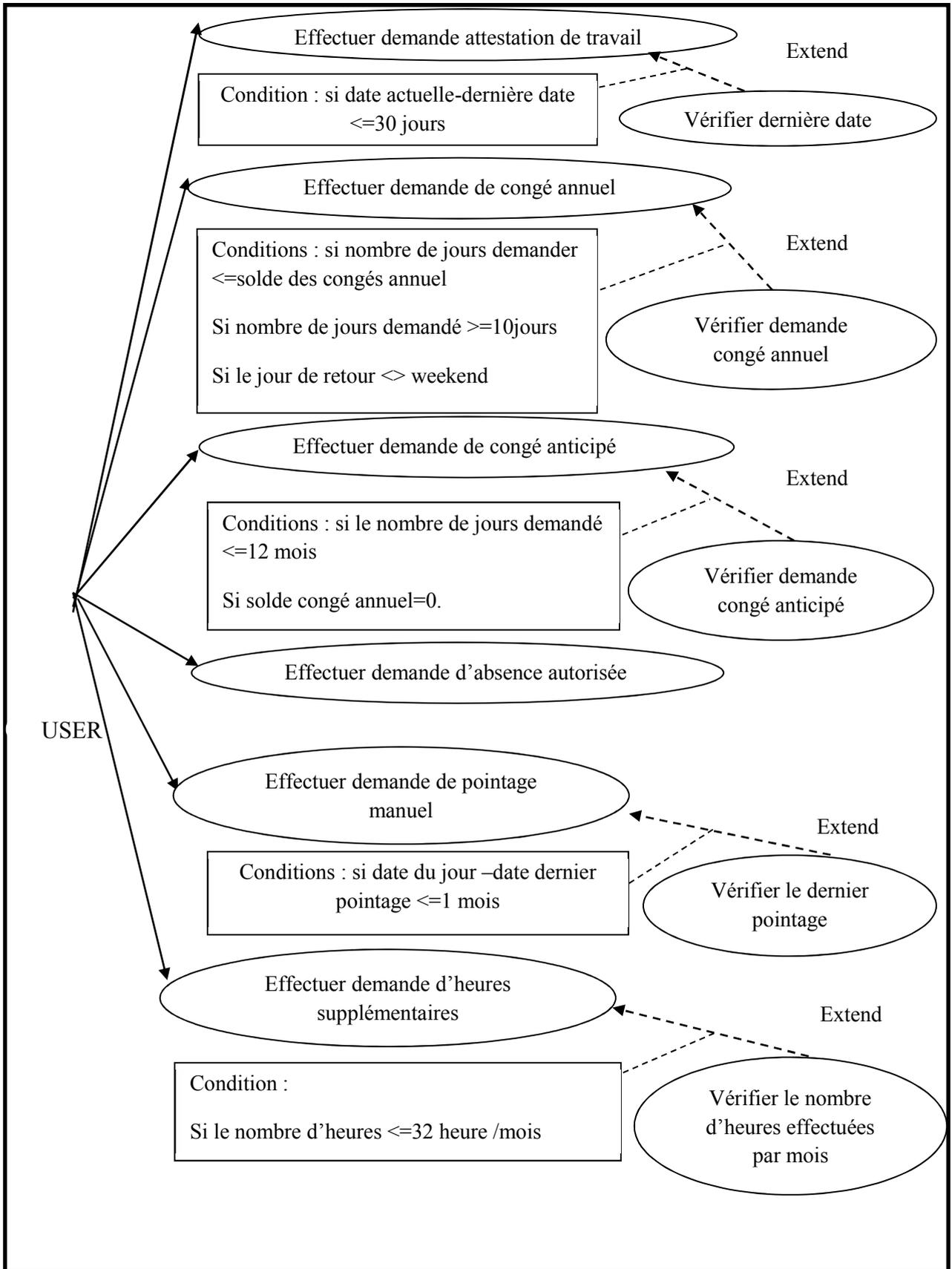


Figure 10 : Diagramme de cas d'utilisation « Valider les demandes ».



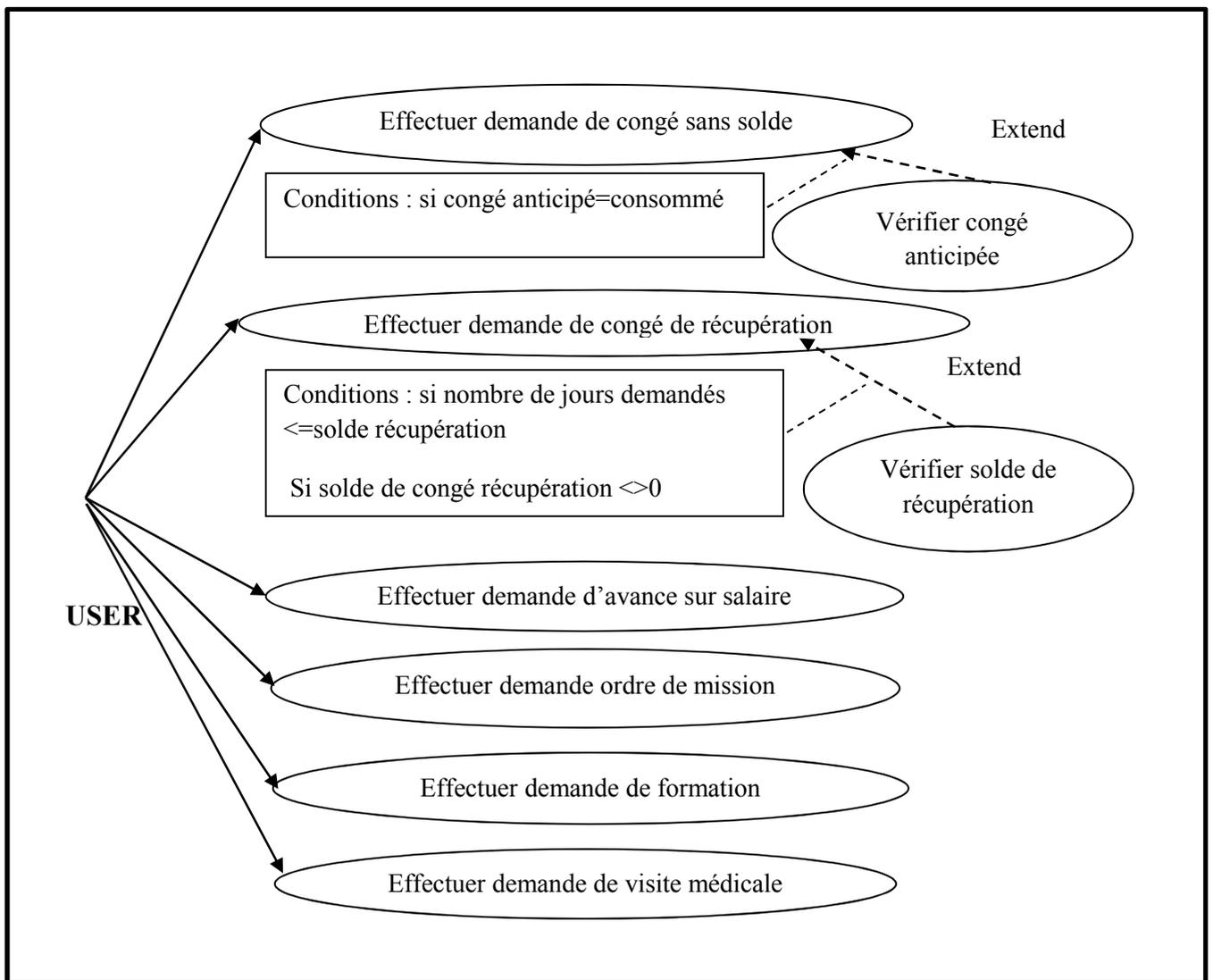


Figure 11 : Diagramme de cas d'utilisation « Effectuer une demande ».

7.2.4 : Paquetage section administrateur :

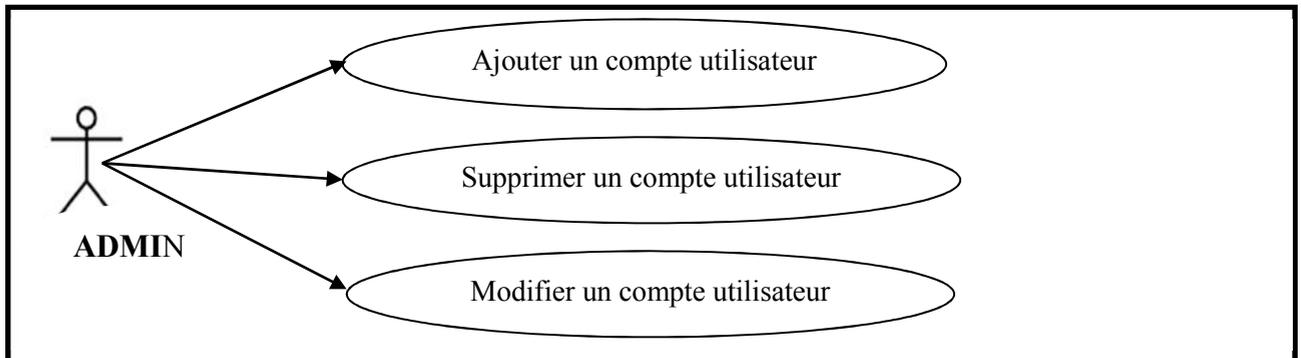


Figure 12 : Diagramme cas d'utilisation « gérer les comptes utilisateurs »

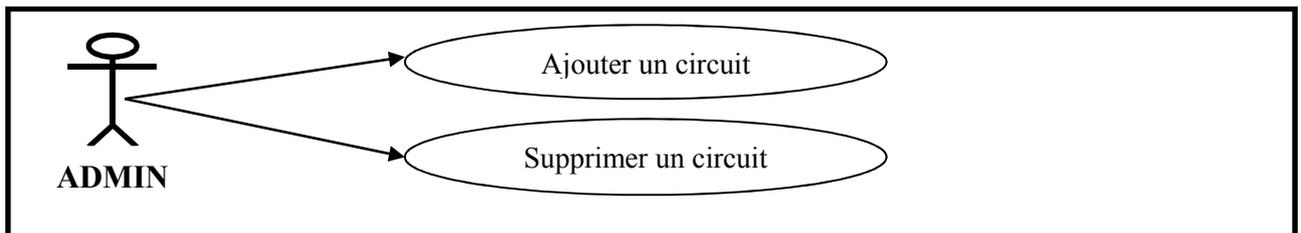


Figure 13 : Diagramme de cas d'utilisation « Gérer les circuits de validation ».

7.3. Spécification des scénarios et des cas d'utilisation :

7.3.1 : Spécification des scénarios :

On appelle scénario une instance de processus, c'est-à-dire chaque résultat possible de l'exécution d'un cas d'utilisation est un scénario.

Un ensemble de scénarios pour un cas d'utilisation identifie tous ce qui peut arriver lorsque ce cas d'utilisation est mis en œuvre, ils facilitent la compréhension des cas et préparent les travaux des recettes à venir.

Remarque : Vu le nombre important de cas d'utilisation ainsi que de scénarios, nous ne décrivons que quelques scénarios.

Acteur	Taches	Scénarios par tache
USER	T1 : Effectuer une demande de congé annuel (personnel/Employé).	S0 : Atteindre l'espace User. S1 : Cliquer sur effectuer une demande. S2 : Cliquer sur demande de congé annuel. S3 : Remplir le formulaire. S4 : Cliquer sur le bouton envoyer.
	T2 : Modifier le mot de passe.	S5 : Atteindre la page Authentification. S6 : Cliquer sur modifier pwd. S7 : Changer le mot de passe.
Admin	T3 : Ajouter un compte utilisateur.	S8 : Atteindre l'espace Admin. S9 : Cliquer sur gestion des comptes utilisateur puis sur ajouter compte S10 : remplir le formulaire. S11 : Cliquer sur le bouton ajouter.
	T4 : Créer un circuit de validation.	S12 : Atteindre l'espace Admin. S13 : Cliquer sur gestion des circuits de validation. S14 : Saisir le nom du circuit. S15 : Choisir les valideurs. S16: Cliquer sur le bouton ajouter.
AGRH	T5 : Ajouter un employé.	S17 : Atteindre l'espace GRH. S18 : Cliquer sur gestion des employés. S19 : Choisir ajouter un employé. S20 : remplir le formulaire. S21 : Cliquer sur ajouter.
	T6 : Ajouter un parcours .	S22 : Atteindre l'espace GRH. S23 : Cliquer sur gestion des parcours puis sur ajouter un parcours. S24 : Saisir le parcours. S25 : Cliquer sur ajouter.
AGMGX	T7 : Affecter un véhicule avec chauffeur.	S26 : Atteindre l'espace GMGX. S27 : Cliquer sur gestion des moyens de transport. S28 : Choisir le véhicule. S29 : Choisir le chauffeur. S30 : Cliquer sur le bouton enregistrer.
	T8 : Effectuer une réservation d'hôtel.	S31 : Atteindre l'espace GMGX. S32 : Cliquer sur gestion des réservations d'hôtel. S33 : Choisir l'hôtel. S34 : Cliquer sur le bouton enregistrer.

Figure 14 : Tableau récapitulatif des scénarios.

7.3.2 : Spécification des cas d'utilisation :

La description des cas d'utilisation est libre. Cependant, cette description prend souvent une forme rédigée qui convient mieux à la communication avec les utilisateurs. Ce qui suit présente les descriptions de quelques cas d'utilisation du système étudié.

❖ Description du cas d'utilisation « **Effectuer une demande de congé annuel (personnel/employé)** ».

Use case : Effectuer une demande de congé annuel.

Scenarios: S0, S1, S2, S3, S4

Acteur : Utilisateur.

Description :

1. L'utilisateur accède à l'espace utilisateur.
2. L'utilisateur clique sur effectuer une demande.
3. Le système lui affiche les différentes demandes disponibles.
4. L'utilisateur clique sur Demande de congé annuel.
5. Le système lui renvoie le formulaire de la demande de congé annuel.
6. L'utilisateur remplit les champs d'information et clique sur le bouton envoyer.
7. Le système envoie le formulaire au valideur numéro 1.
8. Le système envoie une notification à la messagerie personnelle du valideur numéro 1 pour lui indiquer qu'une demande de congé annuel est en attente de validation.

❖ Description du cas d'utilisation « **Modifier mot de passe** ».

Use case : Modifier mot de passe.

Scenarios : S5, S6, S7.

Acteur : Utilisateur.

Description :

1. L'utilisateur accède à la page d'authentification.
2. L'utilisateur clique sur modifier mot de passe.
3. Le système lui renvoie un formulaire contenant les champs utilisateur : pwd actuel, nouveau pwd
4. L'utilisateur remplit le formulaire et clique sur le bouton modifier.
5. Le système lui affiche un message de confirmation si le pwd a été changé sinon, il affiche un message d'erreur et renvoie le même formulaire.

❖ Description du cas d'utilisation « ajouter un compte utilisateur ».

Use case : Ajouter un compte utilisateur.

Scenarios: S8, S9, S10, S11.

Acteur : Administrateur.

Description :

1. L'administrateur atteint l'espace Admin.
2. L'administrateur clique sur gestion des comptes utilisateur puis sur nouveau compte.
3. Le système lui renvoie un formulaire avec les champs utilisateur.
4. L'administrateur remplit le formulaire et clique sur le bouton ajouter.
5. Le système affiche une page de confirmation si le compte est bien créé sinon un message d'erreur et envoie une notification à la messagerie personnelle de l'employé concerné pour lui transmettre son nom d'utilisateur et son mot de passe.

❖ Description du cas d'utilisation « créer un circuit de validation »

Use case : Créer un circuit de validation.

Scenarios: S12, S13, S14, S15, S16.

Acteur : Administrateur.

Description :

1. L'administrateur atteint l'espace Admin.
2. L'administrateur clique sur gestion des circuits de validation.
3. Le système renvoie un formulaire contenant les champs concernant le circuit de validation.
4. L'administrateur remplit le formulaire et clique sur ajouter.
5. Le système affiche une page de confirmation si le circuit est bien créé sinon un message d'erreur.

❖ Description du cas d'utilisation « Ajouter un employé »

Use case : Ajouter un employé.

Scenarios: S17, S18, S19, S20, S21.

Acteur : Agent de gestion des ressources humaines(AGRH).

Description :

1. L'AGRH atteint l'espace GRH.
2. L'AGRH clique sur gérer les employés.
3. L'AGRH clique sur ajouter employé.
4. Le système renvoie un formulaire avec les champs concernant l'employé.
5. L'AGRH remplit le formulaire puis clique sur le bouton ajouter.
6. Le système affiche un message de confirmation si l'employé est bien ajouté sinon un message d'erreur.

❖ Description du cas d'utilisation « Ajouter un parcours »

Use case : Ajouter un parcours.

Scenarios: S22, S23, S24, S25.

Acteur : Agent de gestion des ressources humaines(AGRH).

Description :

1. L'AGRH atteint l'espace GRH.
2. L'AGRH clique sur gestion des parcours.
3. Le système renvoie un formulaire.
4. L'AGRH remplit le formulaire et saisit le parcours (ville de départ, ville d'arrivée) et clique sur ajouter.
5. Le système affiche un message de confirmation si le parcours est bien ajouté sinon un message d'erreur.

❖ Description du cas d'utilisation « Effectuer une réservation d'hôtel »

Use case : Effectuer une réservation d'hôtel.

Scenarios: S26, S27, S28, S29, S30.

Acteur : Agent de gestion des moyens généraux (AGMGX).

Description :

1. L'AGMGX atteint l'espace GMGX.
2. L'AGMGX clique sur gestion des réservations d'hôtel.
3. Le système renvoie un formulaire contenant les champs de l'employé et la liste des hôtels disponibles.
4. L'AGMGX remplit le formulaire et choisit un hôtel puis clique sur OK.
5. Le système envoie une notification à la messagerie personnelle de l'employé pour lui confirmer la réservation.
6. L'AGMGX envoie une notification au GRH pour lui transmettre l'hôtel réservé .

❖ Description du cas d'utilisation « Affecter un véhicule avec chauffeur »

Use case : Affecter un véhicule avec chauffeur.

Scenarios: S31, S32, S33, S34.

Acteur : Agent de gestion des moyens généraux (AGMGX).

Description :

1. L'AGMGX atteint l'espace GMGX.
2. L'AGMGX clique sur gestion des moyens de transport puis sur affecter véhicule avec chauffeur.
3. Le système lui renvoie un formulaire contenant les champs de l'employé et la liste des véhicules et chauffeurs disponibles.
4. L'AGMGX remplit le formulaire et choisit un véhicule et un chauffeur et clique sur enregistrer.
5. Le système envoie une notification à la messagerie personnelle de l'employé pour lui confirmer la réservation.
6. L'AGMGX envoie une notification au GRH pour confirmer l'affectation du véhicule.

7.4. Diagramme de séquence des cas d'utilisation :

Le diagramme de séquence représente des échanges de messages entre objets. Il permet de mieux visualiser la séquence des messages par une lecture de bas en haut.

Son utilisation dans l'analyse nous permet de représenter un processus de façon simplifié, en se centrant sur le *Workflow* et les échanges entre acteurs ou avec le système d'information, voire le système informatique.

On pourra donc l'utiliser pour représenter un processus existant, sans entrer dans le détail des activités.

Dans ce qui suit, nous allons décrire quelques cas d'utilisation, en utilisant le diagramme de séquence.

❖ **Diagramme de séquence : « Valider/Refuser une demande de congé de récupération »**

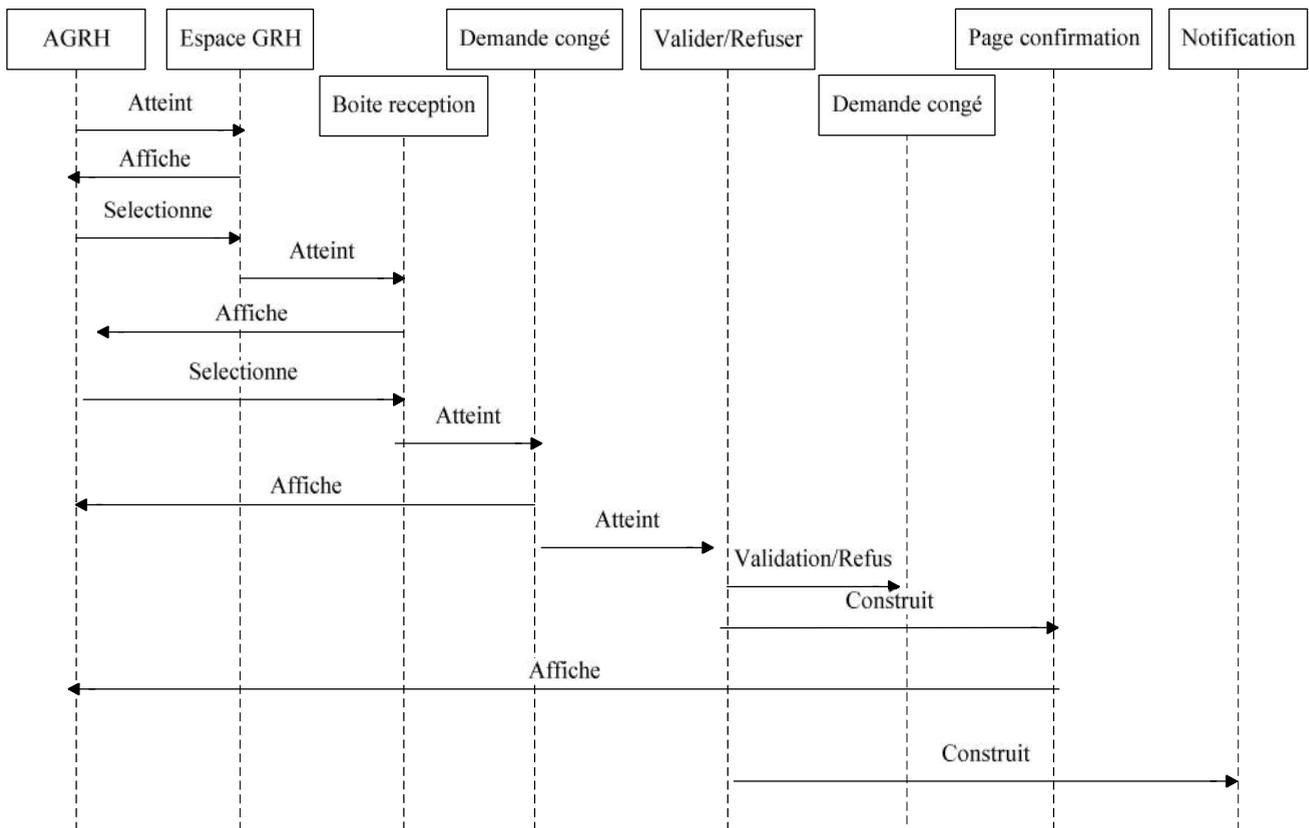


Figure 15 : Diagramme de séquence illustrant le cas d'utilisation « Valider /Refuser une demande de congé de récupération par AGRH ».

Description :

- ✓ L'AGRH atteint son espace.
- ✓ L'AGRH atteint sa boîte de réception.
- ✓ L'AGRH ouvre la demande de congé de récupération non validée et clique sur le bouton valider ou refuser.
- ✓ Si l'AGRH clique sur valider/refuser le système affiche une page de confirmation et envoie une notification à la messagerie personnelle de l'employé concerné.

❖ **Diagramme de séquence : « Effectuer une demande d'absence »**

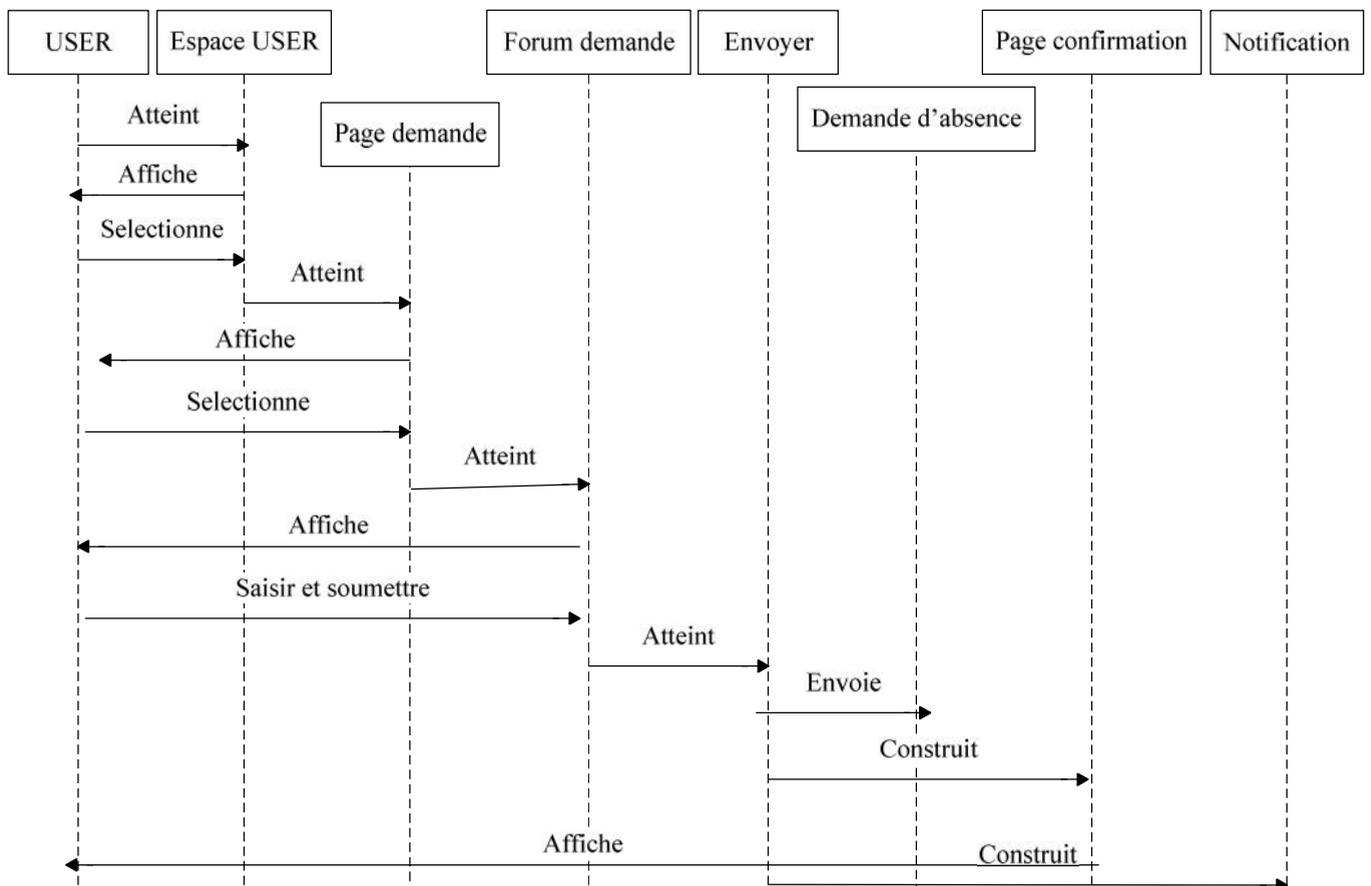


Figure 16 : Diagramme de séquence illustrant le cas d'utilisation « Effectuer une demande d'absence »

Description :

1. L'utilisateur atteint son espace.
2. L'utilisateur atteint le page des demandes.
3. L'utilisateur ouvre le formulaire de la demande d'absence.
4. L'utilisateur saisit et soumet les informations puis clique sur envoyer.
5. Le système envoie la demande à l'autorisateur numero1 de l'employé concerné.

6. Le système affiche une page de confirmation et envoie une notification à l'autorisateur numero1 de l'employé concerné.

7.

❖ **Diagramme de séquence : « Ajouter un compte utilisateur ».**

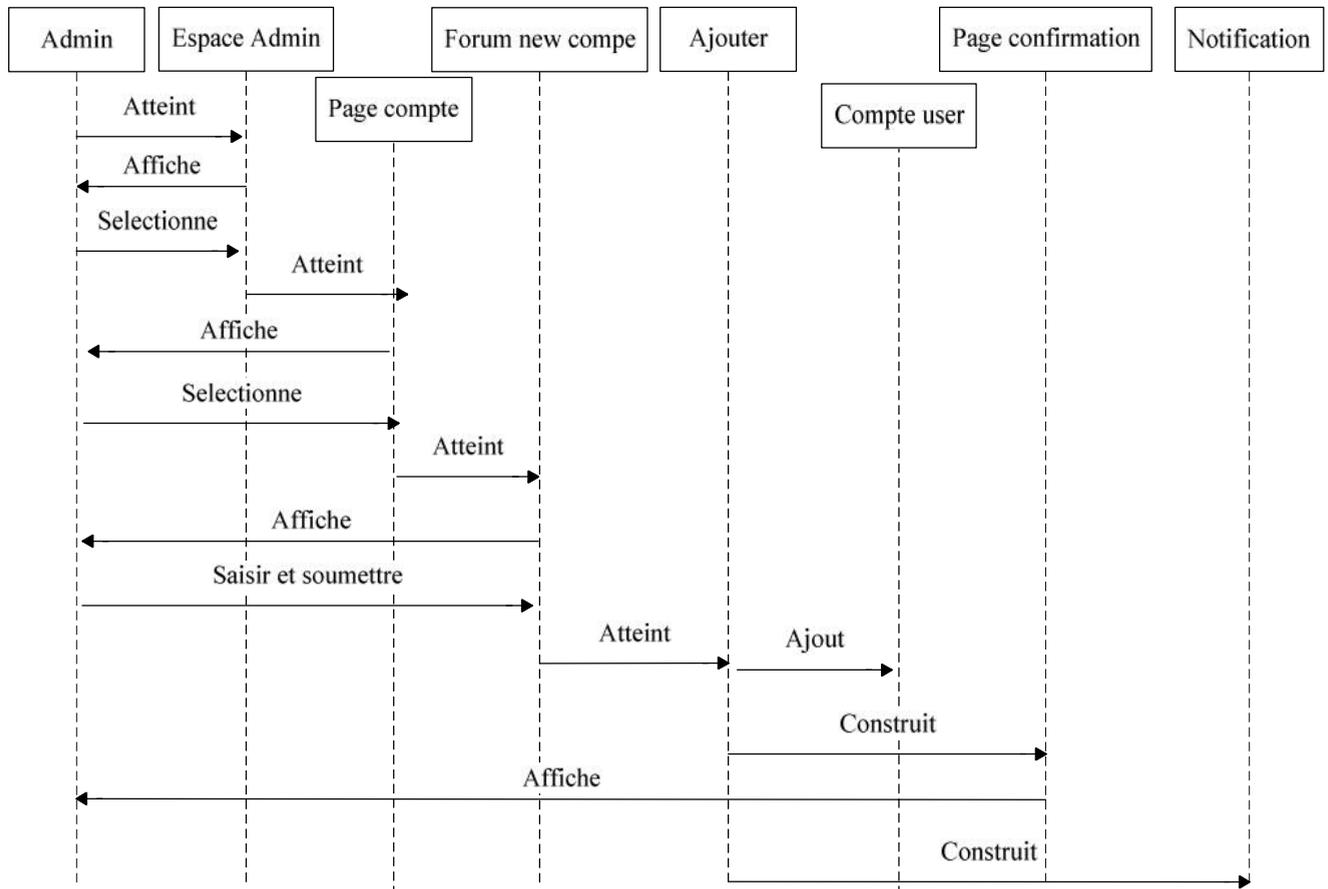


Figure 17 : Diagramme de séquence illustrant le cas d'utilisation « Ajouter un compte utilisateur ».

Description :

1. L'Administrateur atteint son espace.
2. L'Administrateur sélectionne la page gestion des comptes utilisateur.
3. L'Administrateur ouvre le formulaire pour ajouter un nouveau compte utilisateur.
4. L'Administrateur saisie et soumet les informations puis clique sur ajouter.
5. Le système affiche une page de confirmation.
6. Le système envoie une notification à la messagerie personnelle de l'employé concerné pour lui transmettre son nom d'utilisateur et son mot de passe.

7.5. Diagramme d'activités:

Le diagramme d'activité fait partie des diagrammes UML utilisés pour la modélisation des aspects dynamiques des systèmes d'information. Il est attaché à une classe, processus, acteur ou entité a un cas d'utilisation ou bien à une opération, il peut concerner plusieurs classes.

L'apport de l'utilisation de diagramme d'activités dans l'étape d'analyse est essentiel pour modéliser les processus. Il permet de représenter aussi bien les traitements à effectuer que les acteurs impliqués, ainsi que les utilisateurs des informations.

Pour modéliser l'aspect dynamique du système existant, nous allons présenter quelques diagrammes d'activités.

7.5.1. Diagramme d'activité du cas d'utilisation «Authetification » :

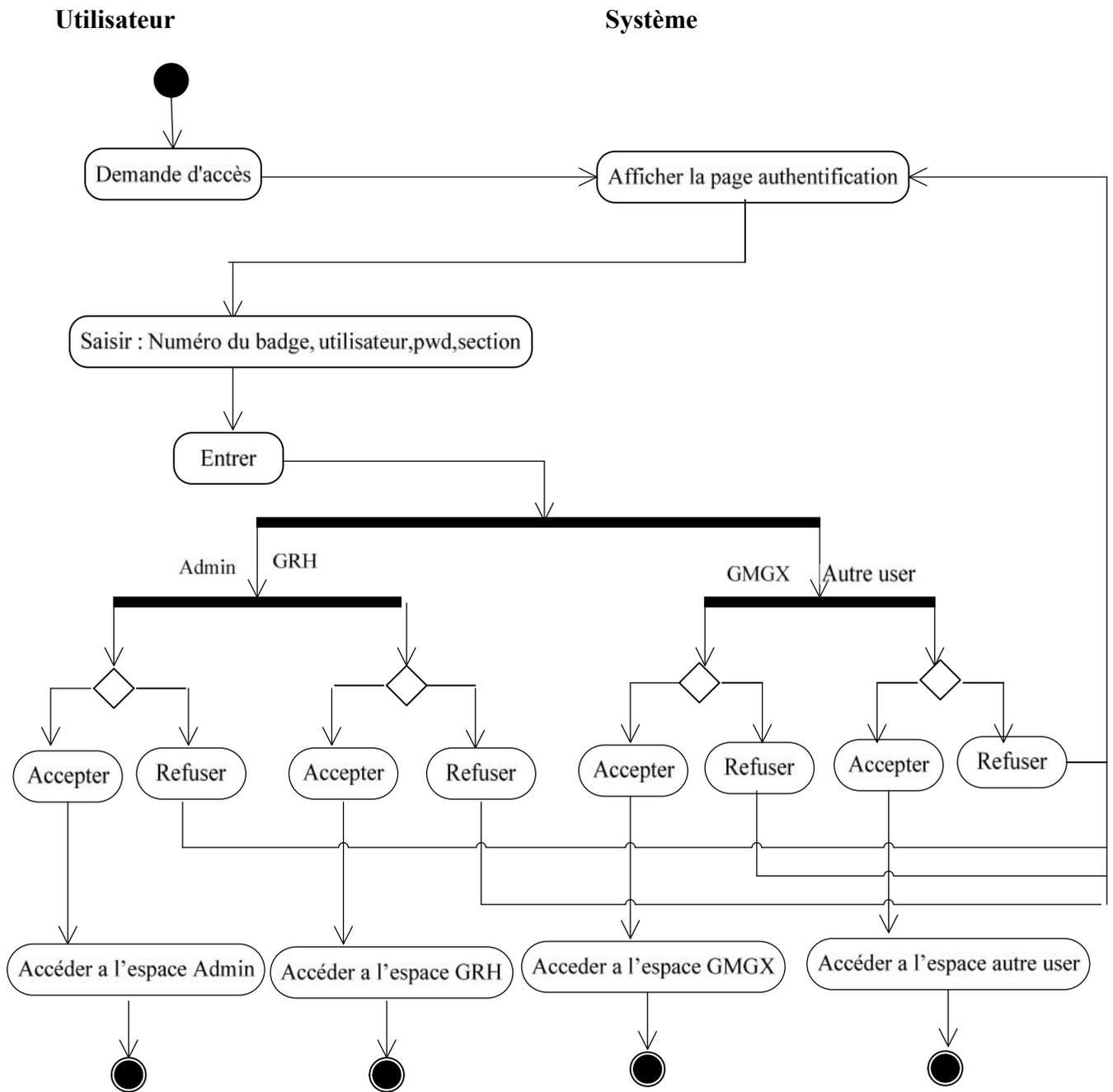


Figure 18: Diagramme d'activité du cas d'utilisation « Identification ».

7.5.2. Diagramme d'activité du cas d'utilisation « Effectuer une demande » :

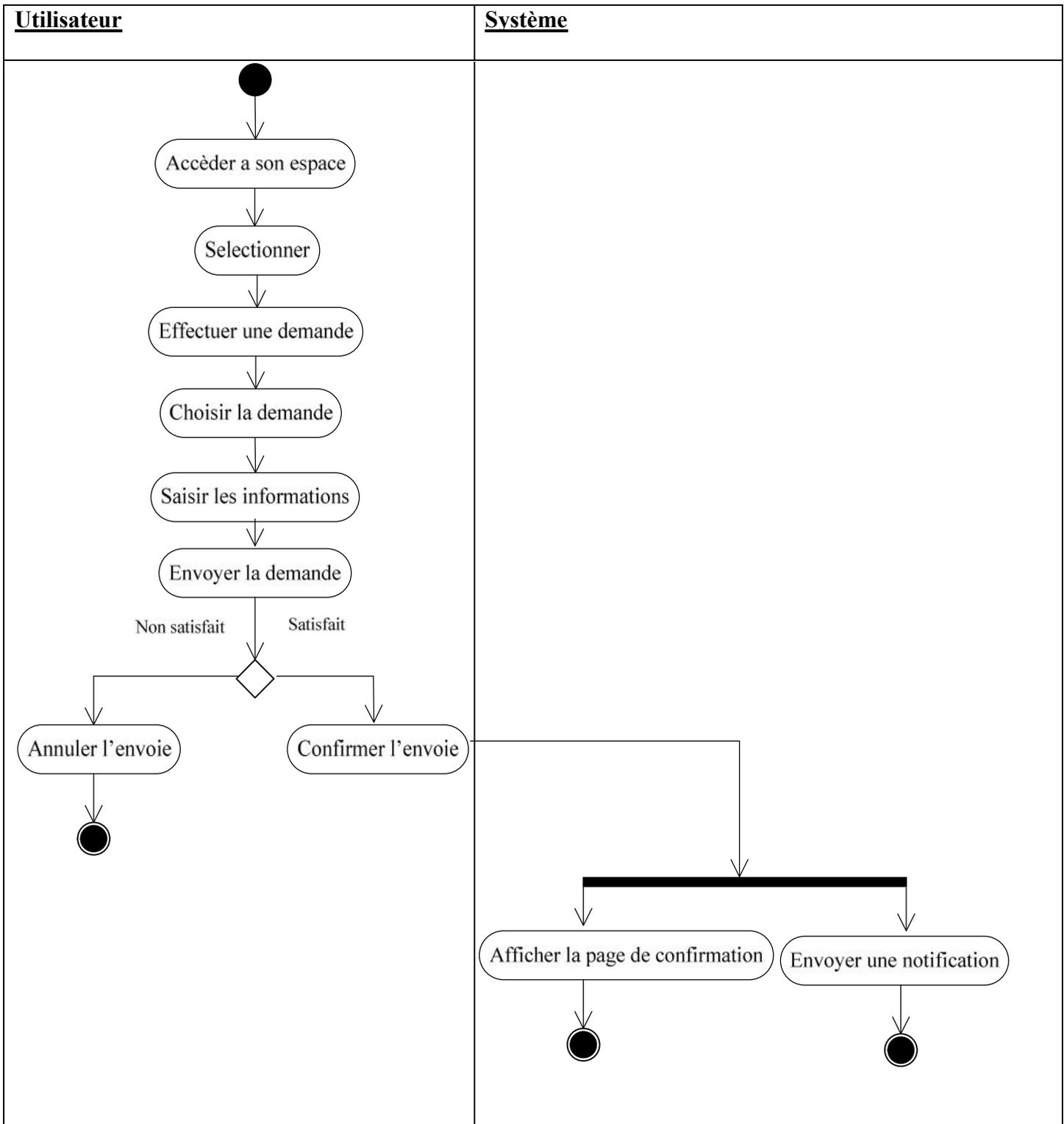


Figure 19 : Diagramme d'activité du cas d'utilisation « Effectuer une demande ».

7.5.3. Diagramme d'activité du cas d'utilisation « Valider une demande » :

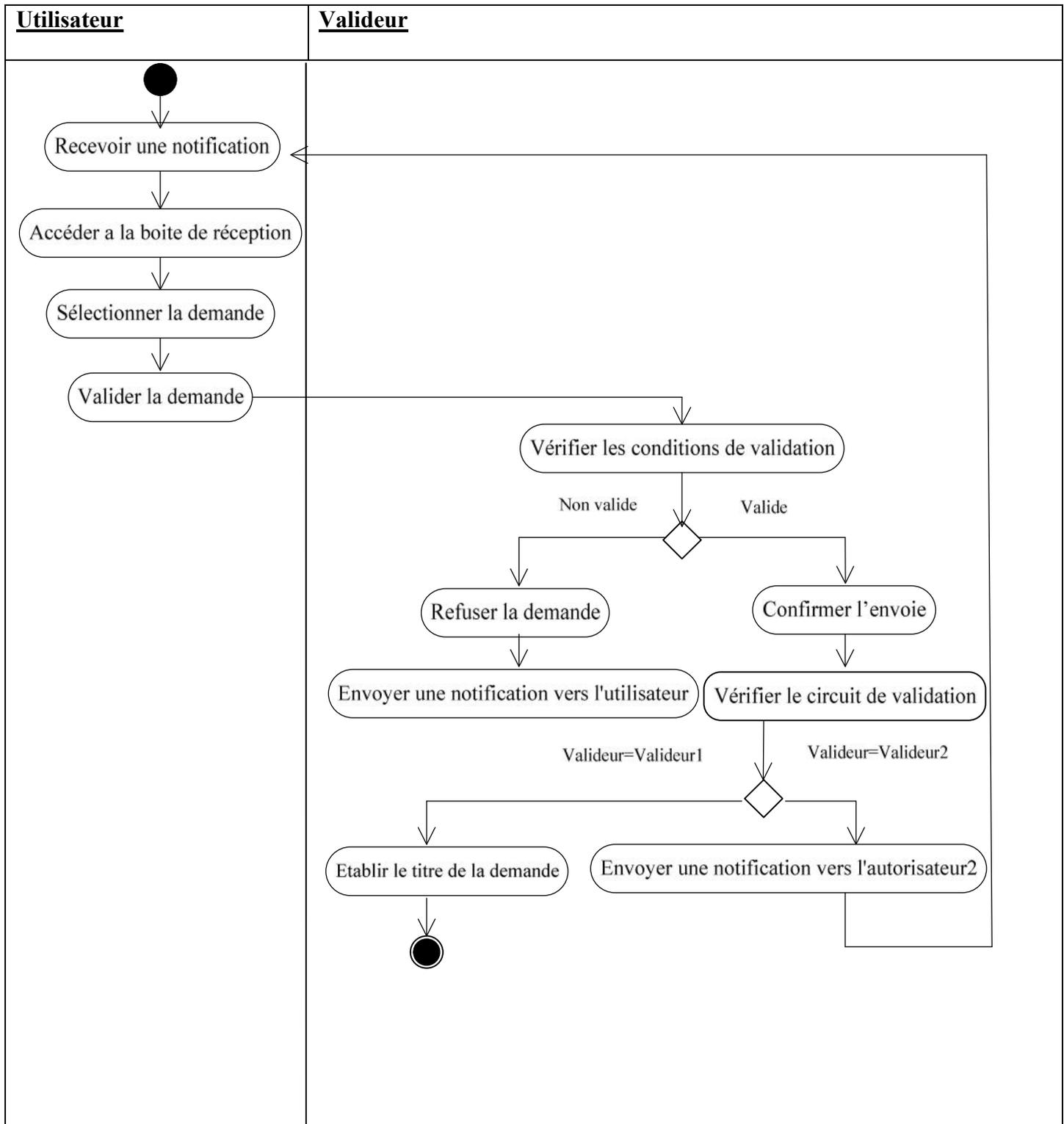


Figure 20: Diagramme d'activité du cas d'utilisation « Valider une demande ».

Conclusion :

La partie analyse nous a permis de comprendre l'organisation actuelle du domaine étudié, ainsi que le fonctionnement de ces différents services.

Dans cette partie nous avons pu identifier :

- L'organisation du domaine : ses différents services ainsi que ses acteurs (responsables, missions, tâches)
- Les traitements effectués dans le domaine et ses procédures de travail.
- Ce que devra faire le futur système (les différentes activités à informatiser).

Dans la partie qui suit nous allons présenter l'architecture de la future application et la description de ses composants.

Chapitre 2 :
Conception

Introduction :

L'objectif de cette partie est de concevoir le système informatique pouvant satisfaire les besoins et les exigences formulées à l'issue de l'étape d'analyse.

Au cours de cette étape, nous allons décrire deux niveaux :

1. Niveau applicatif :

Le travail de ce niveau, consiste à décrire le diagramme de classe des différents cas d'utilisation, ainsi que le diagramme de classe des quatre paquetages définis dans la partie d'analyse, et permet la description des diagrammes de classe des interfaces de l'application Workflow à développer. Ce type de diagramme visualise l'arborescence d'une application.

2. Niveau données :

Le travail de ce niveau, consiste à décrire la base de données de l'application Workflow à développer. Pour ce faire, il faut :

- Définir un dictionnaire de données.
- Définir les règles de gestion du système d'information.
- Décrire le diagramme de classe des différentes entités du système.
- Décrire le modèle logique des données, en utilisant le modèle relationnel.

La description de ces deux niveaux, nous facilite énormément l'implémentation de la base de données, ainsi que la programmation des différentes fonctions de l'application Workflow à développer.

1.3 Diagramme de classe du cas d'utilisation « Modifier un compte utilisateur ».

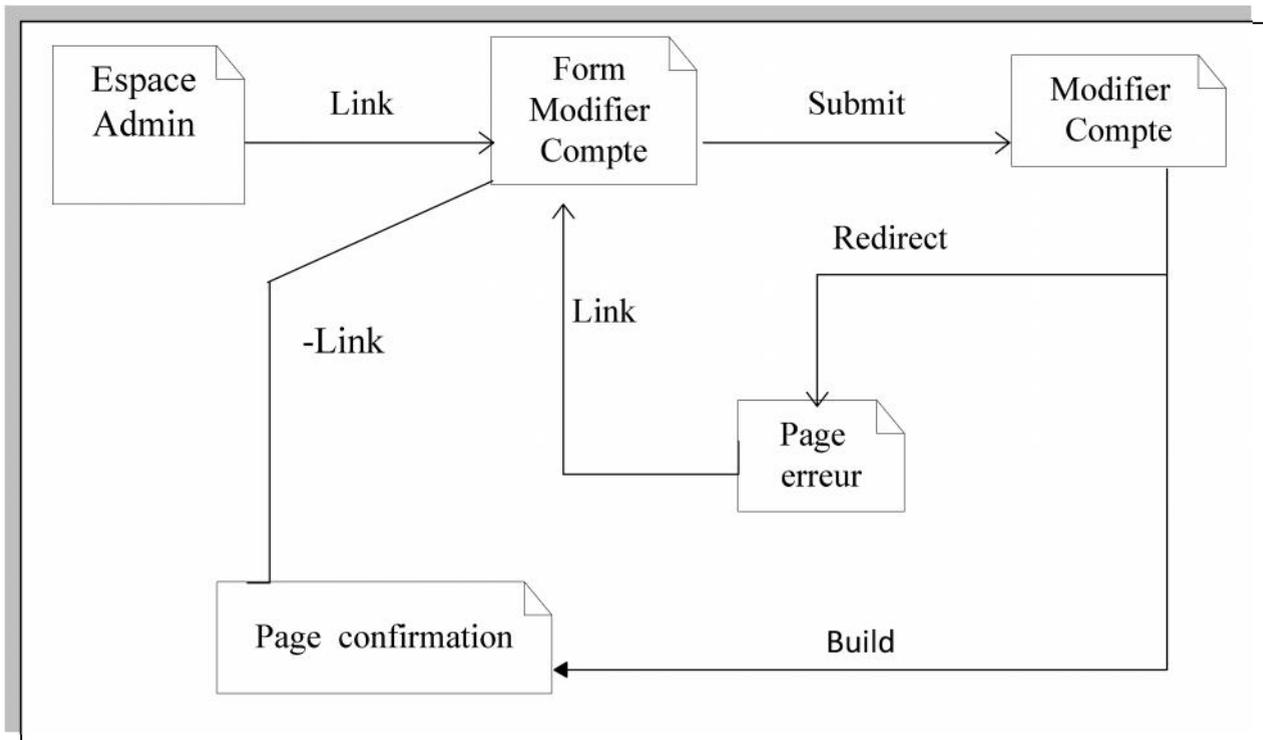


Figure 22: Diagramme de classe du cas d'utilisation « Modifier compte utilisateur ».

1.4 Diagramme de classe du cas d'utilisation « Effectuer une réservation d'hôtel ».

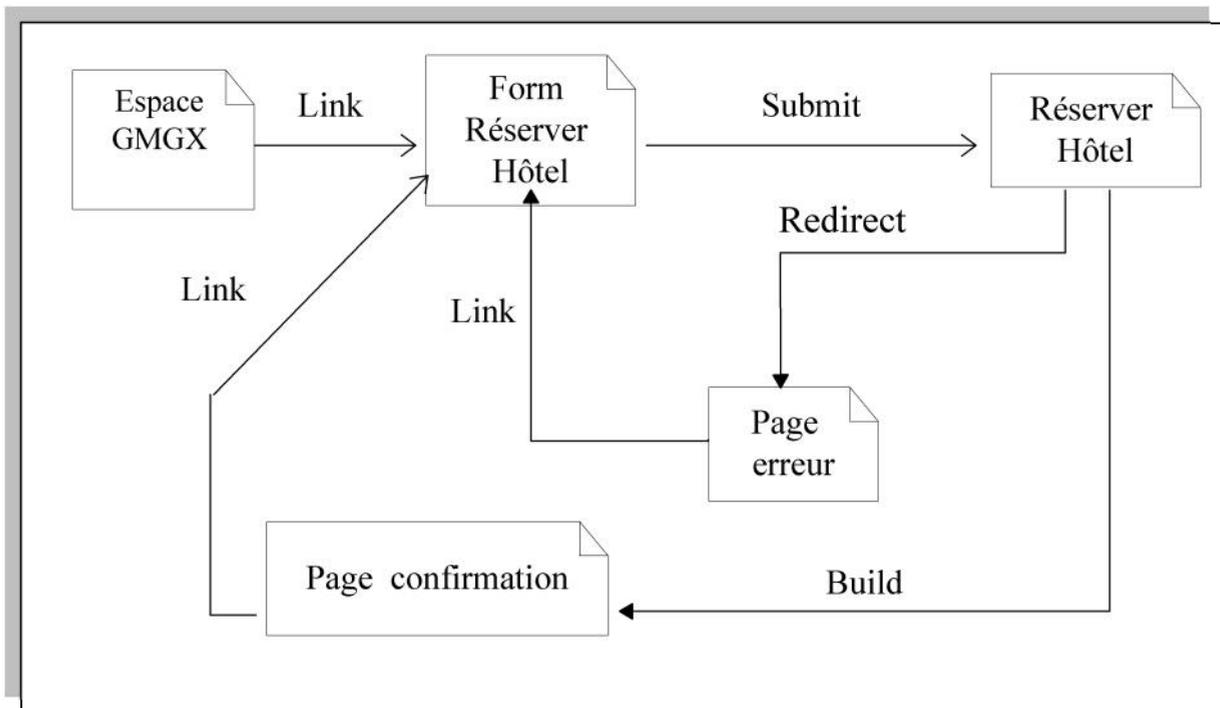


Figure 23: Diagramme de classe du cas d'utilisation « Effectuer une réservation d'hôtel ».

1.5 Diagramme de classe du cas d'utilisation « Valider une demande »

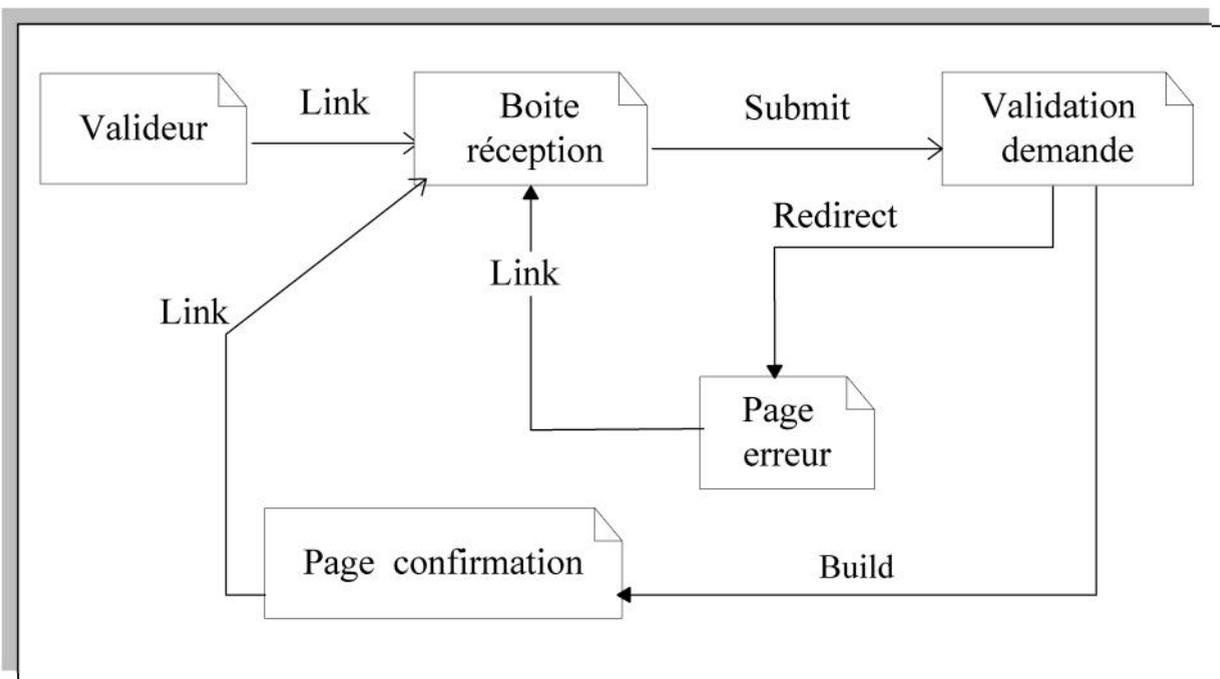


Figure 24: Diagramme de classe du cas d'utilisation « Valider une demande »

1.6 Diagramme de classe du paquetage « ADMIN » :

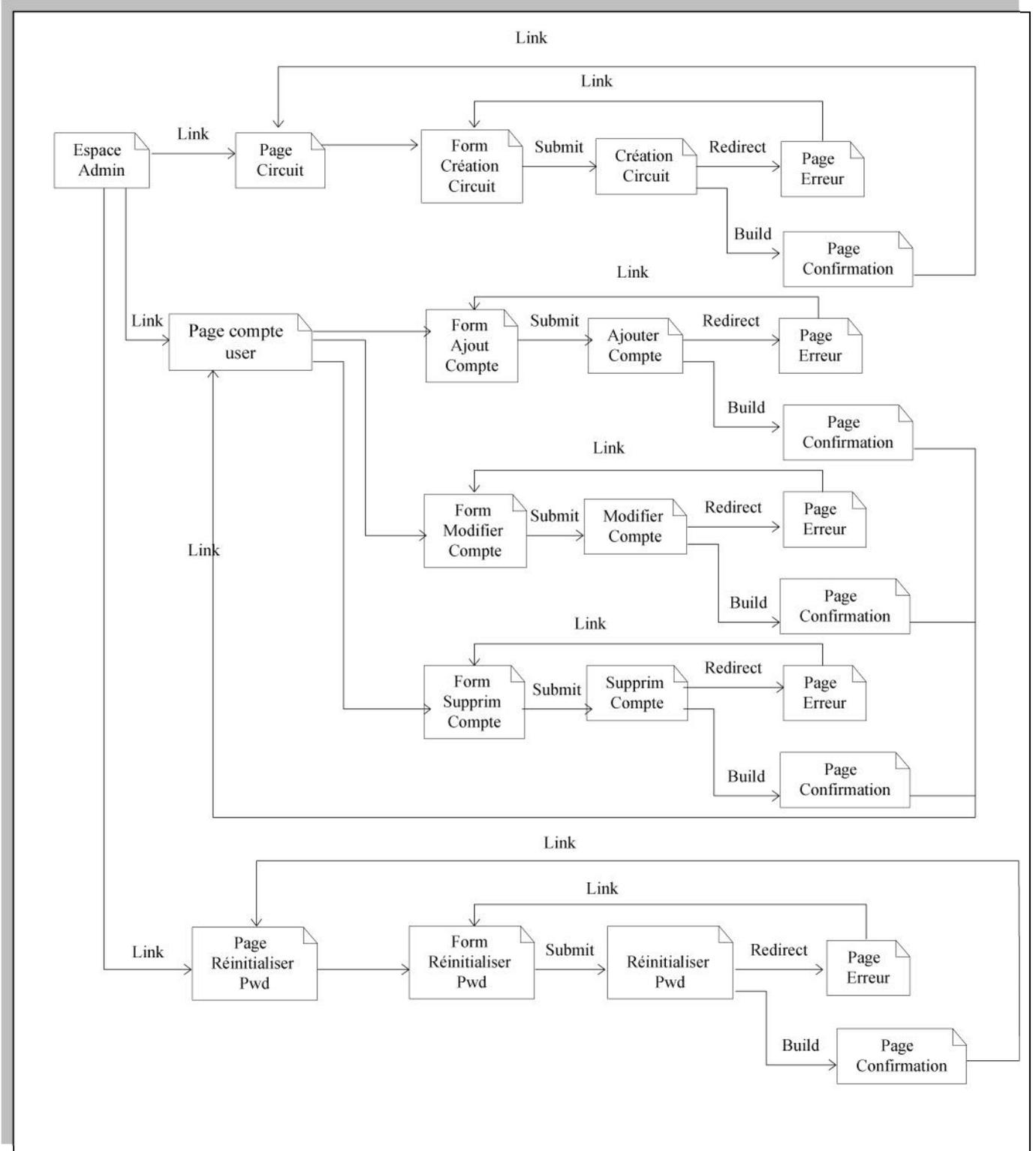
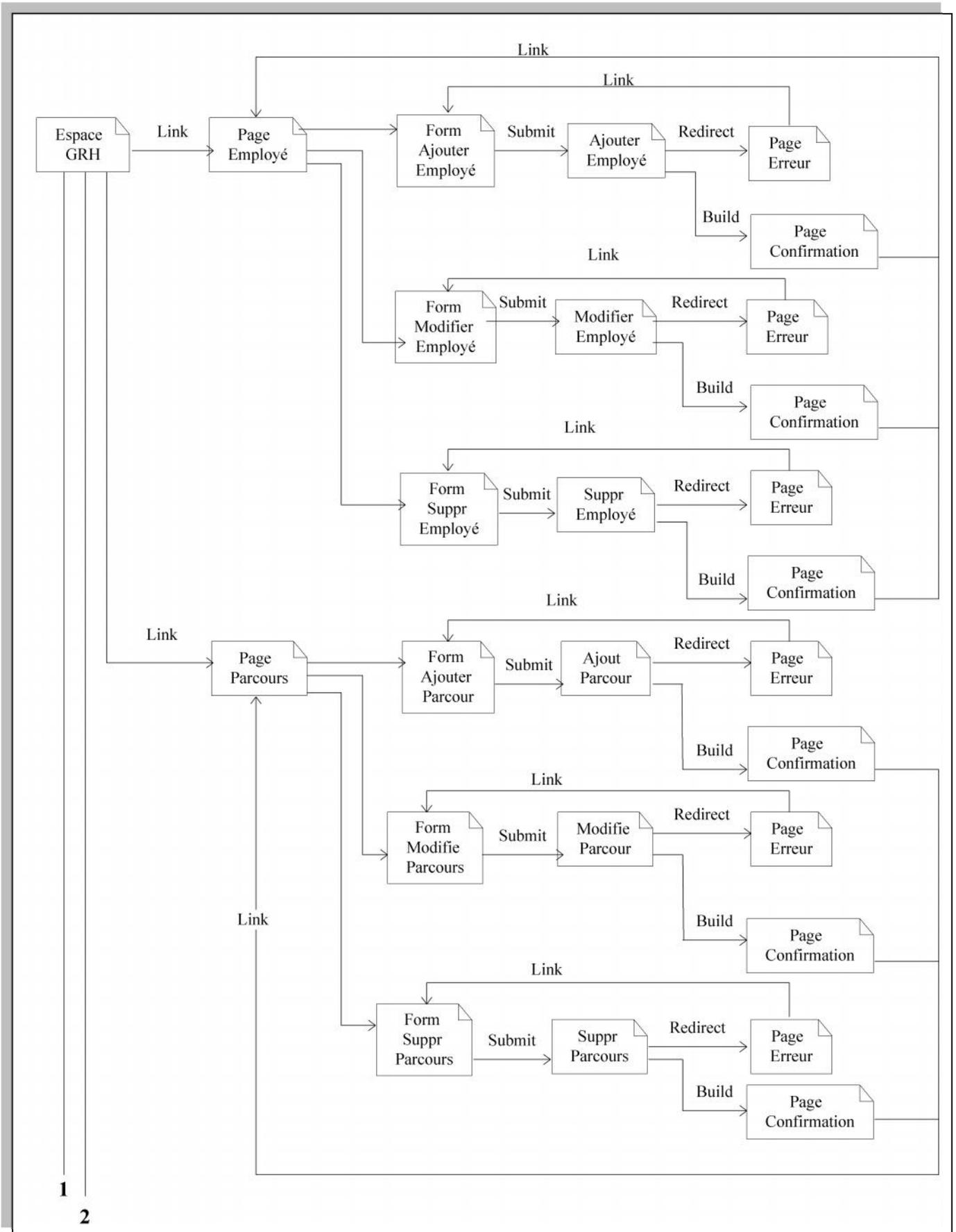


Figure 25: Diagramme de classe du paquetage « ADMIN ».

1.7 Diagramme de classe du paquetage « GRH » :



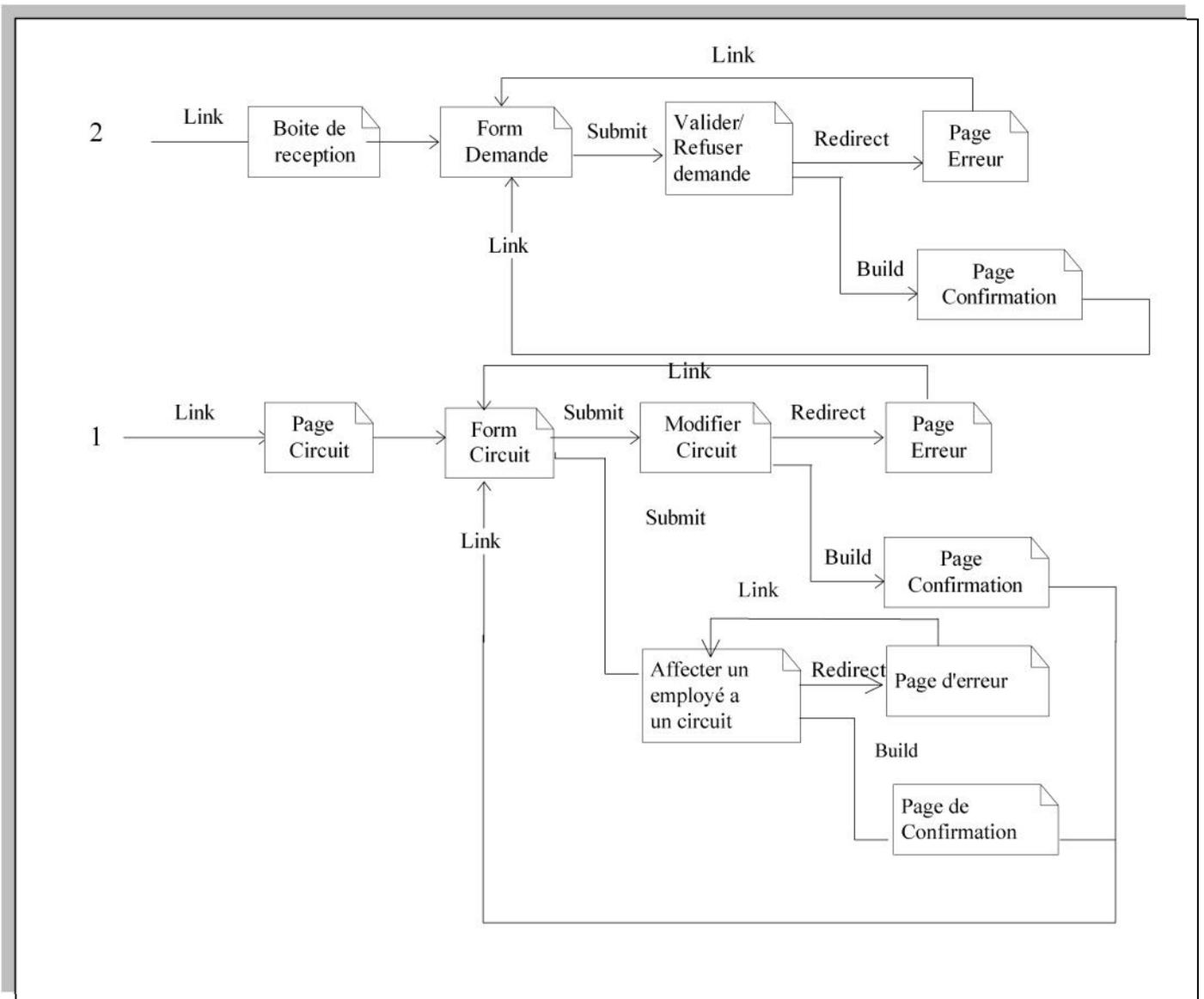


Figure 26 : Diagramme de classe du paquetage « GRH »

1.8 Diagramme de classe du paquetage « GMGX » :

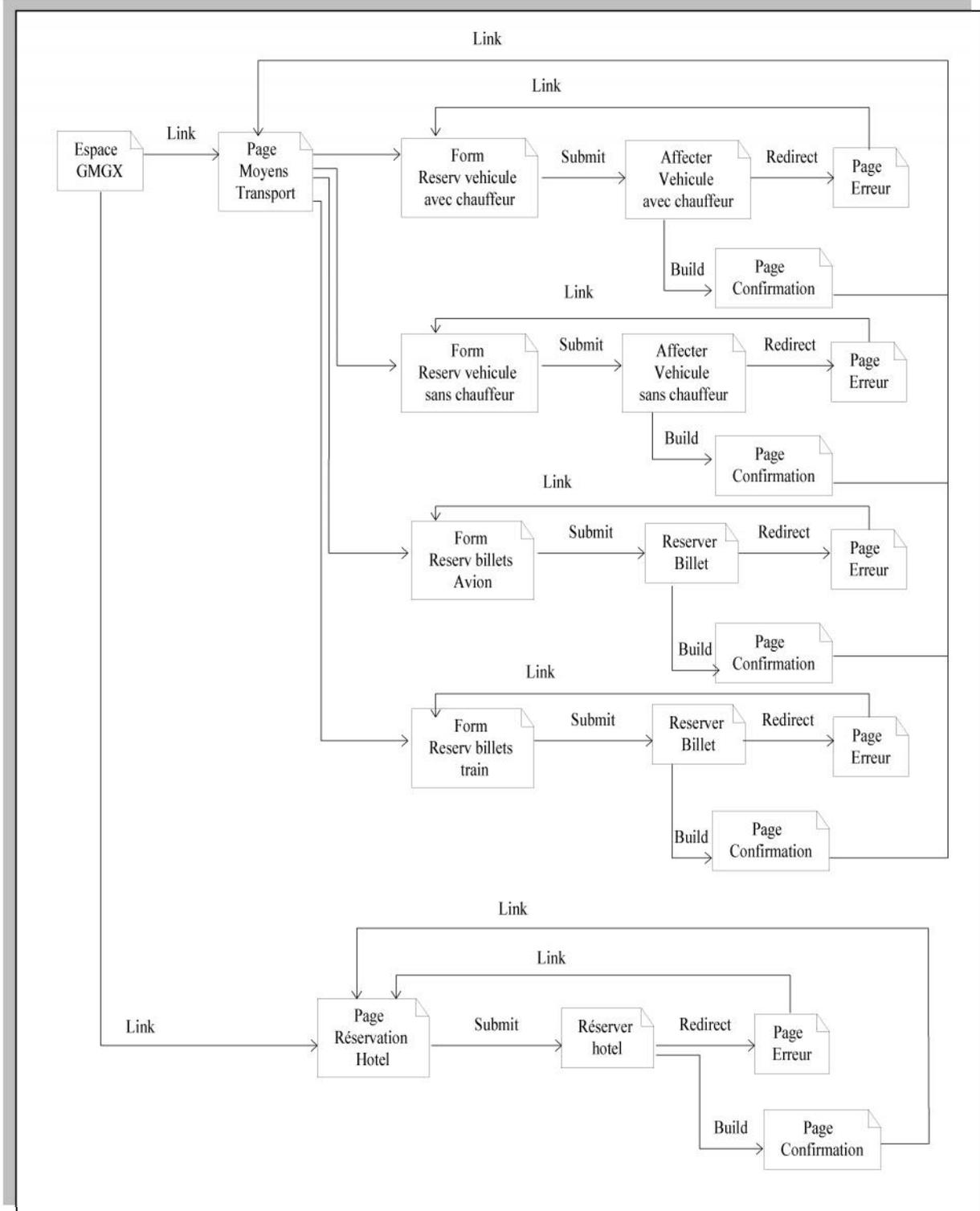


Figure 27: Diagramme de classe du paquetage « GMGX ».

1.9 Diagramme de classe du paquetage « USER » :

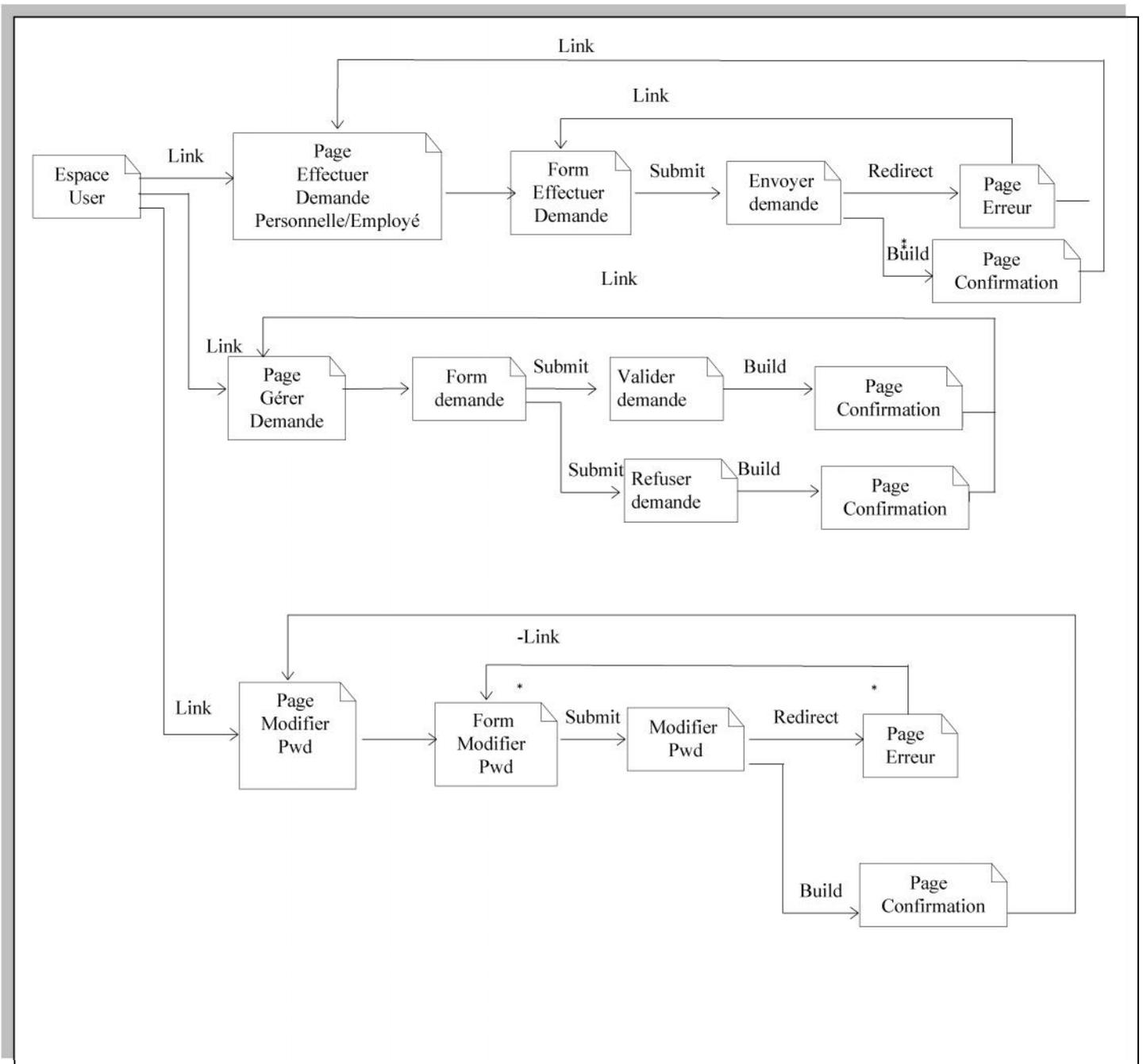


Figure 28: Diagramme de classe du paquetage « USER».

1.10 Le Diagramme de classe des interfaces de l'application :

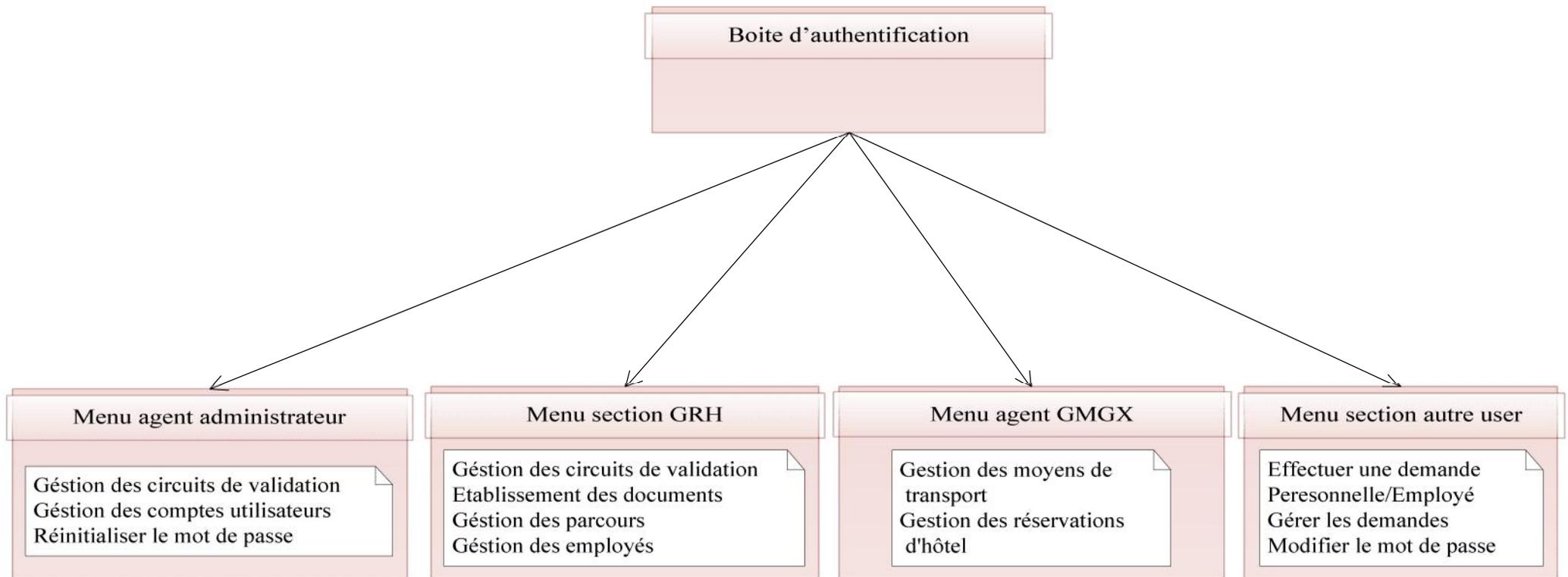


Figure29 : « Diagramme de classe des interfaces de l'application ».

2. Niveau données :

L'objectif de ce niveau, est donc de décrire la base de données de l'application. Pour se faire, il faut d'abord décrire le diagramme de classe final (modèle conceptuel de données), qui sera ensuite traduit en modèle logique de données en utilisant le modèle relationnel.

2.1 Dictionnaire des données :

Durant le stage que nous avons effectué au sein de l'entreprise, nous avons fait l'inventaire de la totalité des propriétés manipulées dans le système d'information (SI) suite à l'étude des différents documents et fichiers qui sont en usage dans les différents postes de l'unité **LALLA KHADIDJA**.

Cet ensemble de propriétés constitue un dictionnaire des données brut, après l'application des règles d'épuration : élimination des redondances, synonymes, valeurs calculées, nous avons obtenus un dictionnaire des données purifié que nous présentons ci-après :

Code	Désignation	Type	Taille	Observation
Adr_CA	Adresse pendant le congé annuel	AN	40	
Adr_emp	Adresse de l'employé	AN	40	
Adr_hot	Adresse de l'hôtel	AN	40	
Billet	Désignation du billet	AN	20	
Cod_User	Code de l'utilisateur	AN	4	
Cod_veh	Code du véhicule	N	10	
Cod_hot	Code de l'hôtel	N	10	
Cod_reserv_bil	Code demande de congé annuel	N	10	
Cod_A_T	Code de l'attestation de travail	N	10	
Cod_D_O_M	Code demande d'ordre de mission	N	10	
Cod_D_C_S_S	Code demande congé sans solde	N	10	
Cod_D_C_R	Code demande congé de récupération	N	10	
Cod_D_A_A	Code demande absence autorisée	N	4	
Cod_D_CA	Code demande congé annuel	N	10	
Cod_D_F	Code demande de formation	N	10	
Cod_D_P_M	Code demande de pointage manuel	N	10	
Cod_D_A_S	Code demande d'avance sur salaire	N	10	
Cod_D_C_A_N	Code demande de congé anticipé	N	10	

Cod_D_H_S	Code demande d'heures supplémentaires	N	10	
Cod_D_V_M	Code demande de visite médicale	N	10	
Cod_T_O_M	Code titre d'ordre de mission	N	10	
Cod_T_A_A	Code titre d'absence autorisée	N	10	
Cod_T_A_S	Code titre d'avance sur salaire	N	10	
Cod_T_A_T	Code titre d'attestation de travail	N	10	
Cod_T_CA	Code titre de congé annuel	N	10	
Cod_T_C_AN	Code titre de congé anticipé	N	10	
Cod_T_C_R	Code titre de congé de récupération	N	10	
Cod_T_C_S	Code titre de congé sans solde	N	10	
Cod_T_F	Code titre de formation	N	10	
Cod_T_H_S	Code titre d'heures supplémentaires	N	10	
Cod_T_O_M	Code titre d'ordre de mission	N	10	
Cod_T_P_M	Code titre de pointage manuel	N	10	
Cod_T_V_M	Code titre de visite médicale	N	10	
Cod_parc	Code du parcours			
Dat_effect_D_A_S	Date effectuer demande d'avance sur salaire	Date		JJ/MM/AA
Dat_nais_emp	Date de naissance de l'employé	Date		//
Dat_recrut	Date de recrutement	Date		//
Desig_circuit	Désignation du circuit	AN	20	
Desig_sect	Désignation de la section	AN	20	
Desig_veh	Désignation du véhicule	AN	20	
Desig_hot	Désignation de l'hôtel	AN	20	
Designation D_C_R	Désignation de la demande de congé de récupération	AN	20	
Designation D_C_AN	Désignation de la demande de congé anticipé	A	20	
Dat_etab_T_C_S	Date établissement du titre de congé sans solde	Date		JJ/MM/AA
Dat_depart_C_S_S	Date départ congé sans solde	Date		//
Dat_recup_souh	Date de récupération	Date		//

	souhaite			
Dat_retour_C_S_S	Date retour congé sans solde	Date		//
Dat_effect_D_A_T	Date effectuer demande d'attestation de travail	Date		//
Designation_A_T	Désignation de l'attestation de travail	A	20	
Dat_effect_D_O_M	Date effectuer demande d'ordre de mission	Date		//
Designation_O_M	Désignation de l'ordre de mission	A	20	
Designation_C_S_S	Désignation du congé sans solde	A	20	
Dat_effect_D_C_S_S	Date effectuer demande de congé sans solde	Date		//
Dat_sortie_C_S_S	Date sortie congé sans solde	Date		//
Dat_reprise_C_S_S	Date reprise congé sans solde	Date		//
Dat_effect_H_S	Date effectuer des heures supplémentaires	Date		//
Dat_effect_D_C_R	Date effectuer demande de congé de récupération	Date		//
Dat_sortie_C_R	Date sortie congé de récupération	Date		//
Dat_reprise_C_R	Date reprise congé de récupération	Date		//
Dat_sortie_A_A	Date sortie absence autorisée	Date		//
Dat_repr_A_A	Date reprise absence autorisée	Date		//
Dat_effect_D_A_A	Date effectuer demande d'absence autorisée	Date		//
Designation_D_A_A	Désignation de l'absence autorisée	A	20	
Dat_sort_CA	Date sortie congé annuel	Date		//
Dat_repr_CA	Date reprise congé annuel	Date		//
Dat_effect_D_CA	Date effectuer demande de congé annuel	Date		//
Designation_CA	Désignation de congé annuel	A	20	
Dat_sort_F	Date sortie pour formation	Date		//
Dat_effect_D_F	Date effectuer demande de formation	Date		//
Dat_repr_F	Date reprise de la formation	Date		//
Destin	Destination	AN	20	
Desingation_D_F	Désignation de la formation	A	20	

Dat_effect_D_P_M	Date effectuer demande de pointage manuel	Date		//
Designation_D_P_M	Désignation de la demande de pointage manuel	A	20	
Designation_D_A_S	Désignation de la demande d'avance sur salaire	A	20	
Dat_effect_D_A_S	Date effectuer demande avance sur salaire	Date		//
Dat_sort_C_AN	Date de sortie du congé anticipé	Date		//
Dat_repr_C_AN	Date de reprise congé anuel	Date		//
Dat_effect_D_C_AN	Date effectuer demande de congé annuel	Date		//
Dat_effect_D_H_S	Date effectuer demande d'heures supplémentaires	Date		//
Designation_D_H_S	Désignation de la demande d'heures supplémentaires	Date		//
Dat_effect_D_V_M	Date effectuer demande de visite médicale	Date		//
Designation D__V_M	Désignation demande visite médicale	A		20
Dat_etab_T_O_M	Date établissement titre d'ordre de mission	Date		//
Dat_depart_O_M	Date départ ordre de mission	Date		//
Dat_retour_O_M	Date retour ordre de mission	Date		//
Dat_etab_T_A_T	Date établissement titre d'attestation de travail	Date		//
Dat_etab_T_CA	Date établissement titre de congé annuel	Date		//
Dat_depart_CA	Date départ congé annuel	Date		//
Dat_retour_CA	Date retour congé annuel	A	30	
Dat_etab_T_C_R	Date établissement de titre de congé de récupération	A	20	
Dat_recup	Date de récupération	A	15	
Direc_rattach	Direction de rattachement de l'employé	A	30	
Dat_etab_T_C_AN	Date d'établissement du titre de congé annuel	Date		//
Dat_depart_C_AN	Date départ congé annuel	Date		//
Dat_retour_C AN	Date retour congé annule	Date		//
Dat_etab_T_A_A	Date établissement titre d'absence autorisée	Date		//
Dat_depart_A_A	Date départ absence	Date		//

	autorisée			
Dat_retour_A_A	Date retour absence autorisée	Date		//
Dat_etab_T_F	Date établissement titre de formation	Date		//
Dat_F	Date de formation	Date		//
Dat_etab_T_P_M	Date établissement titre de pointage manuel	Date		//
Dat_etab_T_A_S	Date établissement titre d'avance sur salaire	Date		//
Dat_etab_T_H_S	Date établissement titre d'heures supplémentaires	Date		//
Dat_effect_H_S	Date effectuer des heures supplémentaires	Date		//
Dat_etab_T_V_M	Date établissement titre de visite médicale	Date		//
Dat_arrive_hot	Date d'arriver à l'hôtel	Date		//
Dat_sortie_hot	Date sortie de l'hôtel	Date		//
Dat_V_M	Date de la visite médicale	Date		//
Dat_depart_O_M	Date départ d'ordre de mission	Date		//
Dat_retour_O_M	Date retour d'ordre de mission	Date		//
Dat_H_S	Date d'effectuer des heures supplémentaires	Date		//
E_mail	email	AN	20	
H_entree	Heure d'entrée	N	5	
H_sortie	Heure de sortie	N	5	
H_depart_O_M	Heure départ ordre de mission	N	6	
H_retour_O_M	Heure retour ordre de mission	N	6	
Horaires_S	Horaires supplémentaires	N	6	
Id_emp	Identifiant de l'employé	N	10	
Id_circuit	Identifiant du circuit	N	10	
Id_val	Identifiant du véhicule	N	3	
Immat_veh	Immatriculation du véhicule	N	10	
Mot_pass	Mot de passe	AN	15	
Motif_A_A	Motif de l'absence autorisée	A	50	
Motif_P_M	Motif du pointage manuel	A	50	
Nom_emp	Nom de l'employé	A	20	
Nom_chauff	Nom du chauffeur	A	20	
Num_tel_emp	Numéro de téléphone de l'employé	N	10	
Num_ss	Numéro de sécurité	N	13	

	sociale			
Nom_User	Nom de l'utilisateur	A	20	
Nom_val1	Nom du premier valideur1	A	20	
Nom_val2	Nom du premier valideur2	A	20	
Nom_vals1	Nom du valideur secondaire1	A	20	
Nom_vals2	Nom du valideur secondaire2	N	20	
Nbr_jou_effect	Nombre de jours effectué	N	2	
Nbr_jou_recup	Nombre de jours récupérés	N	2	
Nbr_jou_CA	Nombre de jours de congé annuel			
Nbr_jou_C_AN	Nombre de jours de congé anticipé	N	2	
Nbr_jou_C_R	Nombre de jours de congé de récupération			
Nbr_jou_A_A	Nombre de jour d'absence autorisée	N	2	
Nbr_jou_C_S	Nombre de jours congé sans solde	N	2	
Nbr_jou_F	Nombre de jours de formation	N	2	
Nbr_H_S	Nombre de jours d'heures supplémentaires	N	2	
Prise_en_char	Prise en charge	A	3	OUI/NON
Pren_emp	Prénom de l'employé	A	20	
Post_occup	Poste occupé	A	20	
parcour	parcours	A	50	
Sexe_emp	Sexe de l'employé	A	1	F : Féminin M : Masculin
Sit_fam_emp	Situation familiale de l'employé	A	1	M : Marié D : Célibataire V : Veuve/Veuf

2.2 Description des classes :

Classes	Identifiant	Attributs
Employé	Id_emp	Nom_emp Pren_emp Sexe_emp Dat_nais_emp Sit_fam_emp Direc_rattach Post_occup Adr_emp Dat_recrut Num_ss Num_tel_emp Desig_circuit E_mail
Compte user	Cod_user	Nom_user Mot_pass Désig_sect Id_emp
Circuit	Id_circuit	Desig_circuit Nom_val1 Nom_val2 Nom_vals1 Nom_vals2
Parcours	Cod_parc	parcour
Véhicule	cod_veh	Desig_veh Immat_veh
Hôtel	Cod_hôt	Desig_hot
Réservation Billet	Cod_reserv_bil	Billet Destin

Demande attestation de travail	Cod_A_T	Dat_recup_souh Dat_effect_D_A_T Designation_D_A_T Id_emp Id_val
Demande d'ordre de mission	Cod_D_O_M	Parcour Dat_effect_D_O_M Prise_en_char Designation_D_O_M Id_val Id_emp
Demande de congé sans solde	Cod_C_S_S	Designation__D_C_S_S Dat_effect_D_C_S_S Dat_sort_C_S_S Dat_repr_C_S_S Id_val Id_emp
Demande congé de récupération	Cod_D_C_R	Dat_effect_H_S Nbr_jou_effect Dat_effect_D_C_R Dat_sort_C_R Dat_repr-C-R Designation_D_C_R Id_val Id_emp
Demande d'absence autorisée	Cod_D_A_A	Dat_sot A_A Dat_reprA_A Dat_effect_D_A_A Designation_D_A_A Id_val Id_emp
Demande congé annuel	Cod_D_CA	Dat_sort_CA Dat_repr_CA

		Adr_CA Dat_effect_D_CA Designation_CA Id_val Id_emp
Demande de formation	Cod_D_F	Dat_sort_F Dat_effect_D_F Dat_repr_F Destin_F Designation_F Id_val Id_emp
Demande de pointage manuel	Cod_D_P_M	Dat_effect_D_P_M H_entree H_sortie Designation_D_P_M Id_val Id_emp
Demande d'avance sur salaire	Cod_D_A_S	Designation_D_A_S Dat_effect_D_A_S Id_val Id_emp
Demande de congé anticipé	Cod_D_C_AN	Dat_sort_C_AN Dat_repr_C_AN Dat_effect_D_C_AN Designation_D_C_AN Id_val Id_emp
Demande heure supplémentaire	Cod_D_H_S	Dat_effect_D_H_S Designation_H_S Id_val Id_emp

Demande de visite médicale	Cod_D_V_	Dat_effect_D_V_M Designation D_V_M
Titre d'ordre de mission	Cod_T_O_M	Dat_etab_T_O_M Dat_depart_T_O_M Dat_retour_T_O_M Id_emp
Titre attestation de travail	Cod_T_A_T	Dat_etab_T_A_T Id_emp
Titre congé annuel	Cod_T_C_A	Dat_etab_T_CA Nbr_jou_CA Dat_depart_CA Dat_retour_CA Id_emp
Titre congé de récupération	Cod_T_C_R	Dat_etab_T_C_R Dat_recup Nbr_jou_recup Id_emp
Titre de congé sans solde	Cod_T_C_S	Dat_etab_T_C_S Dat_depart_C_S Dat_retour_C_S Nbr_jou_C_S Id_emp
Titre de congé anticipé	Cod_T_C_AN	Dat_etab_T_C_AN Dat_depart_C_AN Dat_retour_C_AN Nbr_jou_C_AN Id_emp
Titre d'absence autorisée	Cod_T_A_A	Dat_etab_T_A_A Nbr_jou_A_A Motif_A_A Dat_depart_A_A Dat_retour_A_A Id_emp

Titre de formation	Cod_T_F	Dat_etab_T_F Dat_F Id_emp
Titre de pointage manuel	Cod_T_P_M	Dat_etab_T_P_M Id_emp
Titre d'avance sur salaire	Cod_T_A_S	Dat_etab_T_A_S Id_emp
Titre heures supplémentaires	Cod_T_H_S	Dat_etab_T_H_S Nbr_H_S Dat_effect_H_S Id_emp
Titre visite médicale	Cod_T_V_M	Dat_etab_T_V_M Id_emp

2.3 Description des associations :

Association	Classes	Multiplicités	Propriétés
Appartenir2	Employé	1	
	Circuit	1.*	
Posséder	Compte user	1	
	Employé	0.1	
Créer1	Compte user	1.*	
	Parcours	1	
Créer2	Compte user	1.*	
	Circuit	1	
Réserver1	Titre d'ordre de mission	1	Dat_entree_hot
	Hôtel	1.*	Dat_sortie_hot
Réserver2	Titre d'ordre de mission	1	
	Affectation Véhicule	1.*	Nom_chauff
Contenir	Titre d'ordre de mission	1	
	Réservation billet	1	
Effectuer1	Compte user	1.*	Motif_A_A
	Demande d'absence autorisée	1.*	Nbr_jou_A_A

Effectuer2	Compte user	1.*	Nbr_jou_CA
	Demande de congé annuel	1.*	
Effectuer3	Compte user	1.*	Nbr_jou_C_R
	Demande de congé de récupération	1.*	
Effectuer4	Compte user	1.*	Nbr_jou_C_S_S
	Demande congé sans solde	1.*	Motif_C_S_S
Effectuer5	Compte user	1.*	Nbr_jou_C_AN
	Demande congé anticipé	1.*	
Effectuer6	Compte user	1.*	
	Demande attestation de travail	1.*	
Effectuer7	Compte user	1.*	Nbr_jou_F
	Demande de formation	1.*	
Effectuer8	Compte user	1.*	Motif_P_M
	Demande de pointage manuel	1.*	
Effectuer9	Compte user	1.*	Dat_A_S
	Demande d'avance sur salaire	1.*	
Effectuer10	Compte user	1.*	Dat_V_M
	Demande de visite médicale	1.*	
Effectuer11	Compte user	1.*	Dat_depart_O_M
	Demande d'ordre de mission	1.*	Dat_retour_O_M
			H_depar_O_M
			H_retour_O_M
Effectuer12	Compte user	1.*	Dat_H_S
	Demande heures supplémentaires	1.*	Horair_S

Etablir1	Compte user	1.*	
	Titre d'absence autorisée	1	
Etablir2	Compte user	1.*	
	Titre congé annuel	1	
Etablir3	Compte user	1.*	
	Titre congé de récupération	1	
Etablir4	Compte user	1.*	
	Titre congé sans solde	1	
Etablir5	Compte user	1.*	
	Titre congé anticipé	1	
Etablir6	Compte user	1.*	
	Titre attestation de travail	1	
Etablir7	Compte user	1.*	
	Titre de formation	1	
Etablir8	Compte user	1.*	
	Titre de pontage manuel	1	
Etablir9	Compte user	1.*	
	Titre d'avance sur salaire	1	
Etablir10	Compte user	1.*	
	Titre de visite médicale	1	
Etablir11	Compte user	1.*	
	Titre d'ordre de mission	1	
Etablir12	Compte user	1.*	
	Titre heures supplémentaires	1	

2.4 Les règles de gestion :

R1 : Un employé possède zéro ou un seul compte user, un compte user est possédé par un et un seul employé.

R2 : un circuit est créé par un et un seul compte user, un compte user crée un ou plusieurs circuits.

R3 : Un compte user crée un ou plusieurs parcours, un parcours est créé par un et un seul compte user.

R4 : Un compte user effectue une ou plusieurs demandes, une demande est effectuée par un ou plusieurs comptes user.

R5 : Un titre est établi par un et un seul compte user, un compte user établit un ou plusieurs titres

R6: pour un titre d'ordre de mission on réserve un et un seul hôtel, un hôtel est réservé pour un ou plusieurs titres d'ordre de mission.

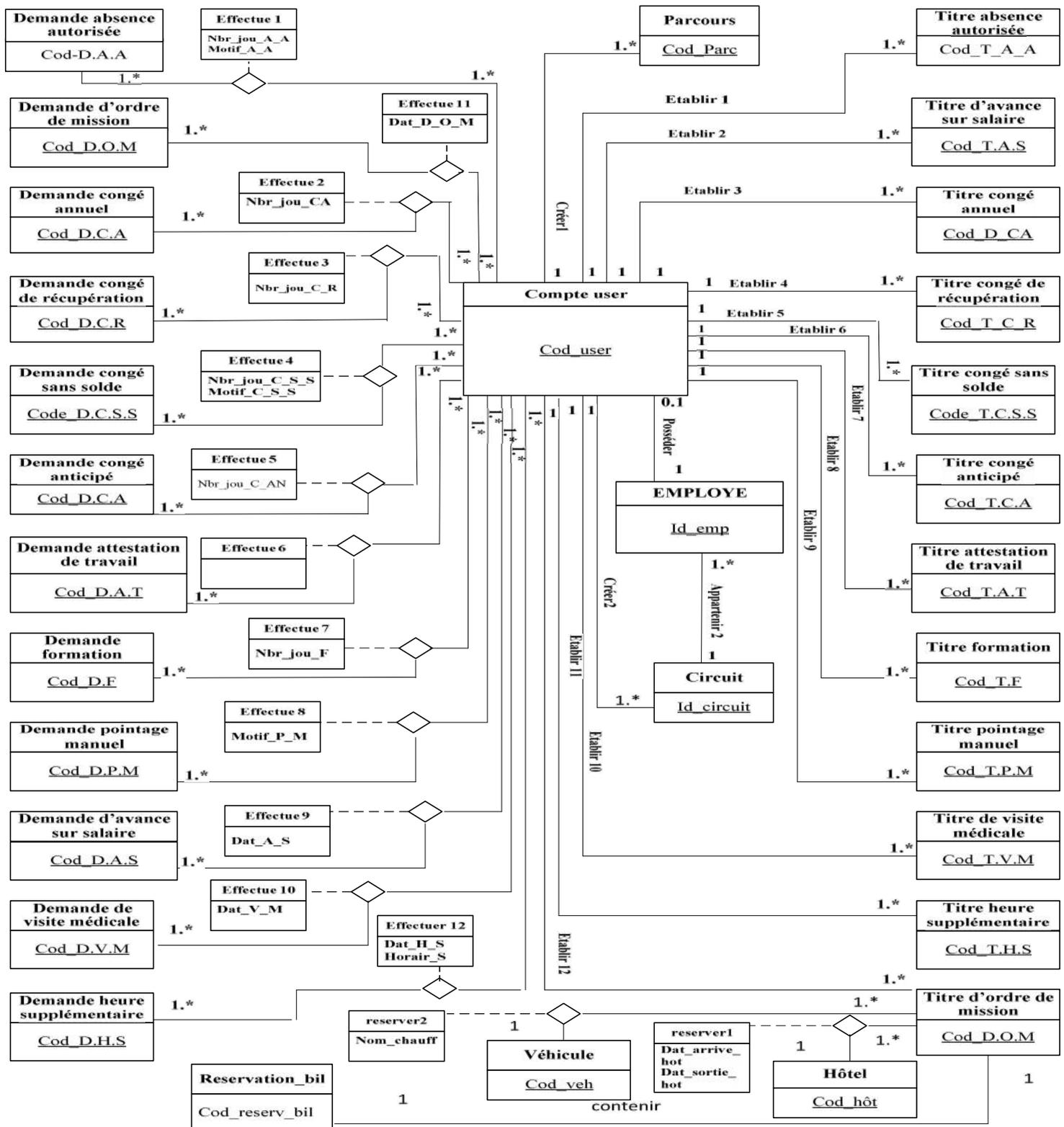
R7 : Un titre d'ordre de mission contient une et une seule réservation de billet, une réservation de billet est contenue dans un et un seul titre d'ordre de mission.

R8: pour un titre d'ordre de mission on réserve un et un seul véhicule, un véhicule est réservé pour un ou plusieurs titres d'ordre de mission.

Remarque :

Vu le nombre important de titres et de demandes, nous avons cité une seule règle pour les demandes et une seule règle pour les titres.

2.5 Le diagramme de classes final :



2.6. Le modèle logique des données :

Une fois le diagramme de classes final établi (modèle conceptuel de données), il est nécessaire de faire le passage au modèle logique des données, tout en respectant les règles de passage.

Pour représenter le niveau logique des données, nous avons opté pour le modèle relationnel, qui offre une modélisation rigoureuse et simple. Aussi, il a permis aux SGBD relationnels de s'imposer dans les projets de développement des systèmes d'information.

➤ Le schéma relationnel correspondant au diagramme de classe final :

Employé (id_emp, Nom_emp, Pren_emp, Sex_emp, Dat_nais_emp, Sit_fam_emp, Direc_rattach, Post_occup, Adr_emp, Dat_recrut, Num_ss, Num_tel_emp, e_mail)

Compte user (Cod_user, Nom_user, Mot_pass, desig_sect, id_emp*)

Circuit (Id_circuit, Desig_circuit, Nom_val1, Nom_val2, Nom_vals1, Nom_vals2, id_emp*)

Parcours (Cod_parc, parcour)

Demande d'absence autorisée (Cod-D-A-A, Dat_sort_A_A, Dat_repr_A_A, Dat_effect_A_A, Designation_A_A, id_val, id_emp*)

Demande congé annuel (Cod_D_CA, , Dat_sort_CA, Dat_repr_CA, Adr_CA, Dat_effect_D_CA, Designation_CA ,id_val , id_emp*)

Demande de congé de récupération (Cod_D_C_R,, Dat_effect_H_S, Nbr_jou_effect, Dat_effect_D_C_R, Dat_repr_C_R, Dat_srtie_C_R ,designation_C_R ,id_emp*,id_val)

Demande congé sans solde (Cod_C_S_S, Dat_sortie_C_S_S, Dat_repr_C_S_S, dat_effect_C_S_S, designation_C_S_S ,id_emp*,id_val)

Demande congé anticipé (Cod_D_C_AN, Dat_sort_C_AN, Dat_repr_C_AN, dat_effect_D_C_AN, designation_C_AN ,id_emp*,id_val)

Demande attestation de travail (Cod_A_T, Dat_recup_souh, dat_effect_A_T, designation_A_T ,id_emp*,id_val)

Demande formation (Cod_D_F, Dat_sort_F, Dat_repr_F, Destin_F, dat_effect_D_F, designation_F ,id_emp*,id_val)

Demande pointage manuel (Cod_D_P_M, H_entree, H_sortie, dat_effect_D_P_M, designation_P_M ,id_emp*,id_val)

Demande d'avance sur salaire (Cod_D_A_S, Dat_A_S ,dat_effect_D_A_S, designation_A_S ,id_emp*,id_val)

Demande de visite médicale (Cod_D_V_M, effect_D_V_M, designation_V_M ,id_emp*,id_val)

Demande d'ordre de mission (Cod_O_M, Parcour, Prise_en_char, dat_effect_O_M, designation_O_M ,id_emp*,id_val)

Demande heure supplémentaire (Cod_D_H_S, dat_effect_D_H_S, designation_H_S ,id_emp*,id_val)

Titre d'absence autorisée (Cod_T_A_A , Dat_etab_T_A_A, Nbr_jou_A_A, Motif_A_A, Dat_depart_T_A_A, Dat_retour_T_A_A, id_emp*)

Titre congé annuel (Cod_T_C_A, Dat_etab_T_C_A, Nbr_jou_C_A, Dat_depart_C_A, Dat_retour_C_A, id_emp*)

Titre de congé de récupération (Cod_T_C_R, Dat_etab_T_C_R, Dat_recup, Nbr_jou_recup,id_emp*)

Titre congé sans solde (Cod_T_C_S, Dat_etab_T_C_S, Dat_depart_C_S, Dat_retour_C_S, Nbr_jou_C_S, id_emp*)

Titre congé anticipé (Cod_T_C_AN, Dat_etab_T_C_AN, Dat_depart_C_AN, Dat_retour_C_AN, Nbr_jou_C_AN, id_emp*)

Titre attestation de travail (Cod_T_A_T, Dat_etab_T_A_T, id_emp*)

Titre formation (Cod_T_F, Dat_etab_T_F, Dat_F, id_emp*)

Titre pointage manuel (Cod_T_P_M, Dat_etab_T_P_M, id_emp*)

Titre d'avance sur salaire (Cod_T_A_S, Dat_etab_T_A_S, id_emp*)

Titre de visite médicale (Cod_T_V_M, Dat_etab_T_V_M, id_emp*)

Titre d'ordre de mission (Cod_T_O_M, Dat_etab_T_O_M, Dat_depart_T_O_M, Dat_retour_T_O_M, id_emp*)

Titre d'heures supplémentaires (Cod_T_H_S, Dat_etab_T_H_S, Nbr_H_S, Dat_effect_HV,id_emp*)

Hotel (cod_hot,designation_hot,cod_T_O_M*)

Vehicule(cod_veh,design_veh,immat_veh,cod_T_O_M*)

Reservation billet (cod_reserv_bil,billet,destin)

Reserver2 (cod_veh,cod_T_O_M,nom_chauf)

Réserver1 (cod_hot,cod_T_O_M,dat_sortie_hot,dat_retour_hot)

Effectuer 1 (Cod_user, Cod_D_A_A, Motif_A_A, Nbr_jou_A_A)

Effectuer 2 (Cod_user, Cod_D_C_A, Nbr_jou_C_A)

Effectuer 3 (Cod_user, Cod_D_C_R, Nbr_jou_C_R)

Effectuer 4 (Cod_user, Cod_C_S_S, Motif_C_S_S, Nbr_jou_C_S_S)

Effectuer 5 (Cod_user, Cod_D_C_AN, Nbr_jou_C_AN)

Effectuer 6 (Cod_user, Cod_A_T)

Effectuer 7 (Cod_user, Cod_D_F, Nbr_jou_F)

Effectuer 8 (Cod_user, Cod_D_P_M, Motif_P_M)

Effectuer 9 (Cod_user, Cod_D_A_S, Dat_A_S_S)

Effectuer 10 (Cod_user, Cod_D_V_M, Dat_V_M)

Effectuer 11 (Cod_user, Cod_O_M, Dat_depart_O_M, retour_O_M, H_depart_O_M, H_retour_O_M)

Effectuer 12 (Cod_user, Cod_D_H_S, dat_H_S, Horair_S)

Conclusion :

Dans les parties analyse et conception, nos efforts étaient accrus, nous avons situé et déterminé l'ensemble des objectifs que doit atteindre l'application à développer.

Ce qui nous ramène à l'étape final (partie implémentation), où nous présenterons le niveau physique des données (tables), l'architecture de développement et les technologies de programmation utilisées.

Chapitre 3 :

Réalisation

Introduction

L'objectif de cette étape est de transformer les éléments décrits lors de la conception en éléments langage cible sur un matériel informatique. Pour ce faire, nous avons utilisé le serveur wamp qui essentiellement regroupe : le serveur web apache, le système de gestion de base de données relationnelle MySQL, le langage de programmation JAVA et l'environnement de développement intégré Netbeans.

Nous avons commencé par la présentation de l'environnement technique développement, ensuite nous avons présenté les tables de base de données ainsi quelques interfaces de notre application.

1. Matériel informatique utilisé :

Pour le développement de notre application, nous avons utilisé un micro-ordinateur avec système d'exploitation : Windows10 Edition intégrale, ayant les caractéristiques suivantes :

- Processeur : Intel®Core™ i5.
- CPU : 16.0Go (15,9 utilisable).
- Type du système : système d'exploitation 64 bits.

2. Description de l'environnement de travail :

2.1 Les outils utilisés :

- **IDE(NetBeans)**

C'est un environnement de développement intégré (IDE) pour java, placé en open source par Sun en juin 2000 sous licence CDDL (Common Développement and Distribution License).

En plus de java, NetBeans permet également de supporter différents autres langages, comme Python, C, C++, XML et HTML. Il comprend toutes les caractéristiques d'un IDE moderne (éditeur en couleur, projet multi-langage, refactoring, éditeur graphique d'interfaces et des pages web) NetBeans est disponible sous Windows, Linux, Solaris (sur x86 et SPARC), MacOS et Open VMS.



Figure 30 : « page démarrage NetBeans ».

- **Wamp serveur [5]:**

WampServeur2 (anciennement WAMP5) est une plate forme de développement Web de type Wamp, permettant de faire fonctionner localement (sans se connecter à un serveur externe) des scripts PHP.

WampServeur n'est pas en soi un logiciel, mais un environnement comprenant deux serveurs (Apache et MySQL), un interpréteur de script(PHP), ainsi qu'une administration pour les deux bases SQL et PhpMyAdmin et SQLiteManager.

Il dispose d'une interface d'administration permettant de gérer et d'administrer ses serveurs au travers d'un try icône (icône près de l'horloge de Windows).

La grande nouveauté deWampServeur2 réside dans la possibilité d'y installer et d'utiliser n'importe quelle version de PHP, Apache et MySQL en un clic. Ainsi chaque développeur peut reproduire fidèlement son serveur de production sur sa machine locale.

Le 21 novembre 2007 sort la version 2.0.Cette version intégré Apache 2.2.6, MySQL 5.0.45 et PHP 5.2.5.

- **PHPMYAdmin [5]:**

PHPMYAdmin est une interface conviviale gratuite réalisée en langage PHP pour le SGBD MySQL afin de faciliter la gestion des bases de données MySQL sur un serveur, et est distribué licence GNU GPL.

Il s'agit de l'une des plus célèbres interfaces pour gérer une base de données MySQL sur un serveur.

PHP. De nombreux hébergeurs, qu'ils soient gratuit ou payant le propose c'est qu'il permet à l'utilisateur de ne pas avoir à installer une base de données. Ce système est très pratique pour sauvegarder une base de données sous forme de fichier.sql et ainsi transférer facilement ses données. De plus celui-ci accepte la formulation de requêtes SQL directement en langage SQL, cela permet de tester ses requêtes par exemple lors de la création d'un site et ainsi de gagner un temps précieux.

PHPMYAdmin peut gérer un serveur MySQL (cela nécessite un compte super-utilisateur) aussi bien qu'une seule base de données. Pour pouvoir administrer une seule base de données concernée. Il vous appartient de consulter la section appropriée dans le manuel de SQL.

Cette interface pratique permet d'exécuter, très facilement et sans grandes connaissances dans le domaine des bases de données, de nombreuses requêtes comme les créations de tables de données, les insertions, les mises à jours, les suppressions, les modifications de structure de tables.

Table	Action	Rows	Type	Collation	Size	Overhead
<input type="checkbox"/> circuit	Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	16 KiB	
<input type="checkbox"/> compte	Browse Structure Search Insert Empty Drop	3	InnoDB	latin1_swedish_ci	16 KiB	
<input type="checkbox"/> demande_c_an	Browse Structure Search Insert Empty Drop	3	InnoDB	latin1_swedish_ci	16 KiB	
<input type="checkbox"/> effectuer2	Browse Structure Search Insert Empty Drop	3	InnoDB	latin1_swedish_ci	16 KiB	
<input type="checkbox"/> employe	Browse Structure Search Insert Empty Drop	4	InnoDB	latin1_swedish_ci	16 KiB	
<input type="checkbox"/> hotel	Browse Structure Search Insert Empty Drop	3	InnoDB	latin1_swedish_ci	16 KiB	
<input type="checkbox"/> parcours	Browse Structure Search Insert Empty Drop	1	InnoDB	latin1_swedish_ci	16 KiB	
<input type="checkbox"/> reservation_bil	Browse Structure Search Insert Empty Drop	0	InnoDB	latin1_swedish_ci	16 KiB	
<input type="checkbox"/> reservation_hotel	Browse Structure Search Insert Empty Drop	0	InnoDB	latin1_swedish_ci	16 KiB	
<input type="checkbox"/> reservation_veh	Browse Structure Search Insert Empty Drop	0	InnoDB	latin1_swedish_ci	16 KiB	
<input type="checkbox"/> vehicule	Browse Structure Search Insert Empty Drop	0	InnoDB	latin1_swedish_ci	16 KiB	
11 tables	Sum	19	InnoDB	latin1_swedish_ci	176 KiB	0

Figure 31: « Interface de PHPMyadmin ».

2.2 . Les langages de programmation :

Afin de réaliser cette application nous avons utilisé les langages suivants :

- **SQL :**

SQL (sigle de Structured Query Language) est un langage informatique normalisé qui sert à demander des opérations sur des bases de données. La partie langage de manipulation de données de SQL permet de rechercher, d'ajouter, de modifier ou de supprimer des données dans des bases de données.

En plus du langage de manipulation de données, la partie langage de définition de donnée permet de créer, et de modifier l'organisation des données dans la base de données, la partie langage de contrôle de transaction permet de commencer et de terminer des transactions, et la partie de contrôle des données permet d'autoriser ou d'interdire l'accès à certaines données à certaines personnes.

Créé en 1974, normalisé depuis 1986, le langage est reconnu par la grande majorité des systèmes de gestion de base de données (SGBD) du marché.

- **java :**

Java est un langage de programmation informatique orienté objet créé par James Gosling et Patrick Naughton de Sun Microsystems. Mais c'est également un environnement d'exécution.

Java peut être séparée en deux parties. D'une part, le programme écrit en langage java et d'autre part, une machine virtuelle(JVM) qui va se charger de l'exécution du programme java. C'est cette plateforme qui garantit la portabilité de java. Il suffit qu'un système ait une machine virtuelle Java pour que tout programme écrit en ce langage puisse fonctionner.

3. Accès aux bases de données :

La technologie JDBC (Java Data Base Connectivity) est une API fournit avec java (depuis sa version 1.1) permettant de se connecter aux bases de données java.

L'API JDBC a été développée de telle façon à permettre à un programme de se connecter à n'importe quelle base de données utilisant la même syntaxe, c'est-à dire que l'API JDBC est indépendante du SGBD.

De plus JDBC bénéficie des avantages de java, dont la portabilité du code, ce qui lui vaut en plus d'être indépendant de la base de données d'être indépendant de la plate forme sur laquelle il s'exécute

3.1 Les outils nécessaires pour utiliser JDBC :

Pour pouvoir utiliser JDBC, il faut un pilote qui est spécifique à la base au quelle on veut accéder. Avec le JDK, Sun fournit un pilote qui permet l'accès à la base de données via ODBC (Open Data Base Connectivity).

Les classes de JDBC sont regroupées dans le package java.sql et sont incluses dans JDK à partir de sa version 1.1.La version 2.0 de cette API est incluse dans la version 1.2 du JDK.

Il y a 4 classes importantes, chacune correspond à une étape de l'accès aux données :

- **Driver Manager** : Charge et configure le driver de la base de données.
- **Connexion** : Réalise la connexion et l'authentification à la base de données.
- **Statement (et Prepared Statement)** : Contient la requête SQL et la transmet à la base de données.
- **ResultSet** : Permet de parcourir les informations retournées par la base de données dans le cas d'une sélection de données.

3.2 La connexion à la base de données :

- **Le chargement du pilote** : Pour se connecter à une base de données via ODBC, il faut tout d'abord charger le pilote JDBC-ODBC qui fait le lien entre les deux.
- **L'établissement de la connexion** : pour se connecter à une base de données il faut instancier un objet de la classe connexion en lui précisant sous forme d'URL la base à laquelle on accède.

4. Présentation de quelques interfaces et fonctionnalité de l'application :

➤ Page « Login »

C'est la première page qui apparaît lors du lancement de l'application, elle demande à l'utilisateur de saisir son badge, son nom d'utilisateur, son mot de passe et sa section pour pouvoir accéder à son espace.



Figure 32 : « Page login ».

➤ Page « modifier pwd »

Cette page permet à un utilisateur de changer son mot de passe

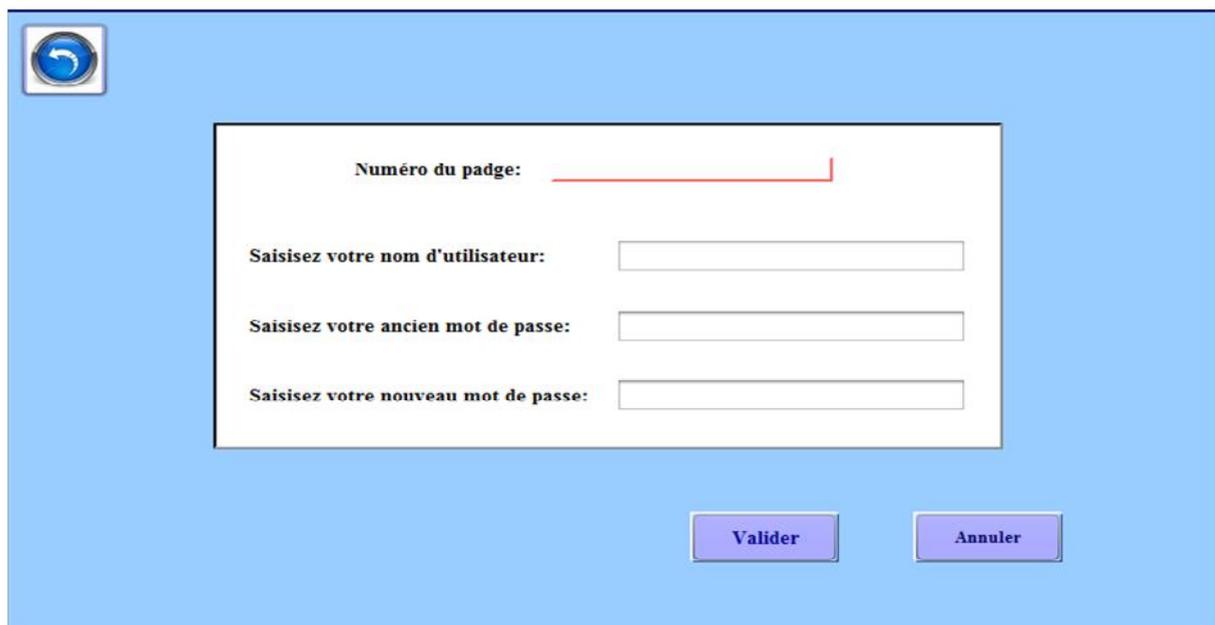


Figure 33 : « Page modifier pwd ».

➤ Page « GRH »

C'est l'espace de la section « gestion des ressources humaines »

The screenshot shows a web application interface for Human Resources Management (GRH). The main heading is "ESPACE GESTION DES RESSOURCES HUMAINES". Below this, there are tabs for "Gestion des employés", "Gestion des circuits de validation", and "Gestion des parcours". The left sidebar contains a "Retour" button and two main buttons: "Boite de réception" and "Effectuer une demande". The central area has a sub-heading "BIENVENUE CHEZ CEVITAL" and a form for adding an employee. The form includes the following fields:

- Numéro du badge: 124
- Date de recrutement: 2015/09/06
- Nom: nom2
- Prénom: prenom2
- Sexe: Féminin
- Date de naissance: 1987/09/09
- Situation familiale: Mariée
- Poste: chef de quart
- Direction: Lalla Achechefs
- Adresse de l'employé: adresse
- Numéro de téléphone: 0549785020
- Numéro de sécurité sociale: 53442442
- Email: emza.fekouss@gmail.com

Buttons for "Ajouter" and "Annuler" are located at the bottom of the form.

Figure 34 : « Page GRH ».

➤ Page « GMGX »

C'est l'espace de la section « gestion des moyens généraux ».

The screenshot shows a web application interface for General Means Management (GMGX). The main heading is "ESPACE GESTION DES MOYENS GENERAUX". Below this, there are tabs for "Gestion des moyens de transport" and "Gestion des réservations d'hôtel". The left sidebar contains a "Retour" button and two main buttons: "Boite de réception" and "Effectuer une demande". The central area has a sub-heading "ESPACE GESTION DES MOYENS GENERAUX" and a form for registering a vehicle. The form includes the following fields:

- Numéro du badge: [input field]
- Nom: [input field]
- Prénom: [input field]
- Destination: [input field]
- Véhicule: [dropdown menu]
- Chauffeur: [dropdown menu]

Buttons for "Enregistrer" and "Annuler" are located to the right of the form.

Figure 35: « Page GMGX ».

➤ Page « Administrateur »

C'est l'espace de la section « Administrateur ».

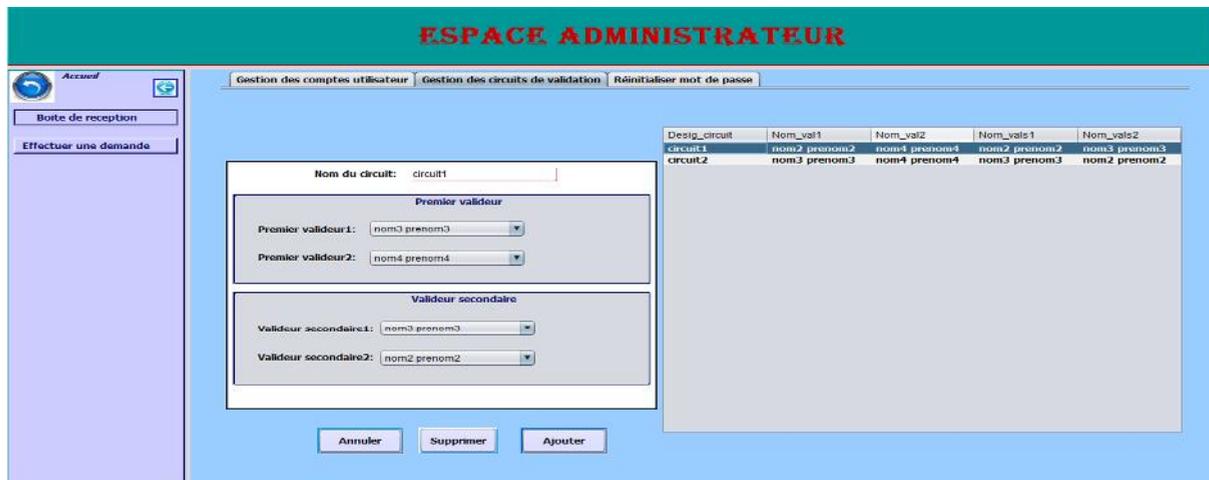


Figure 36 : « Page Administrateur ».

➤ « Messagerie personnelle »

Réception de notifications dans la messagerie personnelle de l'employé



Figure 37: « Messagerie personnelle »

➤ Page « effectuer une demande »

Cette page permet à un utilisateur de choisir une demande et de l'envoyer a son valideur.

Figure 38 : « Page effectuer une demande ».

5. Présentation de quelques tables de la base de données :

➤ Table Compte :

	Cod_User	Nom_User	Mot_pass	Desig_sect	Id_emp
<input type="checkbox"/> Edit Copy Delete	1	employe2	employe2	AGRH	124
<input type="checkbox"/> Edit Copy Delete	2	employe3	employe3	AGRH	125
<input type="checkbox"/> Edit Copy Delete	3	employe4	employe4	AGMGX	126

➤ Table Circuit :

	Id_circuit	Desig_circuit	Nom_val1	Nom_val2	Nom_vals1	Nom_vals2
<input type="checkbox"/> Edit Copy Delete	1	circuit1	nom2 prenom2	nom4 prenom4	nom2 prenom2	nom3 prenom3
<input type="checkbox"/> Edit Copy Delete	2	circuit2	nom3 prenom3	nom4 prenom4	nom3 prenom3	nom2 prenom2

➤ Table Demande congé annuel :

	code_D_CA	Designation	dat_sort	dat_repr	adr_CA	Id_emp	Id_val
<input type="checkbox"/> Edit Copy Delete	1	Demande de congé annuel	2016-05-01	2016-05-16	adresse	123	124
<input type="checkbox"/> Edit Copy Delete	2	Demande de congé annuel	2017-10-01	2017-10-16	GGGGGGG	125	123
<input type="checkbox"/> Edit Copy Delete	3	Demande de congé annuel	2017-10-01	2017-11-01	adresse	123	124

➤ **Table effectuer 2 :**

			code_eff	nbr_jour	date_demd	ld_emp	
<input type="checkbox"/>	 Edit	 Copy	 Delete	1	30	2016-09-23	123
<input type="checkbox"/>	 Edit	 Copy	 Delete	2	15	2017-09-23	124
<input type="checkbox"/>	 Edit	 Copy	 Delete	3	30	2017-09-23	123

➤ **Table employé :**

Id_emp	Nom_emp	Pren_emp	Sexe_emp	Dat_nais_emp	Sit_fam	Direc_rattach	Post_occup	Adr_emp	Dat_recrut	Num_ss	Num_tel_emp	Desig_circuit	e_mail
123	nom1	prenom1	Féminin	1980-04-22	Marié(e)	Lalla khedidja	Maintenancier	adresse	2010-09-01	53454345	549785020	circuit1	lynda.hedjar@gmail
124	nom2	prenom2	Féminin	1987-09-09	Marié(e)	Lalla khedidja	chef de quart	adresse	2015-09-06	53442442	549785020	null	hemza.fekouss@gr
125	nom3	prenom3	Féminin	1992-09-07	Célibataire	Lalla khedidja	cariste	adresse	2016-08-09	4765664	549785020	null	hedjar.lynda@gmail
126	nom4	prenom4	Féminin	1992-09-07	Célibataire	Lalla khedidja	cariste	adresse	2016-08-09	4765664	549785020	null	hedjar.lynda@gmail

Conclusion :

Nous avons abordé dans ce chapitre les différents outils de développement et les langages de programmation utilisés pour le développement de notre application Workflow et nous avons aussi présenté les tables de la base de données et quelques interfaces de l'application développée.

CONCLUSION GENERALE

Dans ce projet, nous avons réalisé une application Workflow pour le système d'information de l'Unité de traitement d'eau minérale « LALLA KHEDIDJA » permettant d'informatiser les différents flux d'information d'une façon plus relationnelle et efficace.

En outre, ce travail nous a permis, d'une part d'avoir un aperçu sur le milieu de travail et d'une autre part de prendre connaissances de la gestion organisationnelle et informationnelle d'une entreprise.

Aussi, nous avons pu mettre en pratique les connaissances théoriques acquises durant notre cursus universitaire notamment nous familiariser avec l'environnement de développement IDE NetBeans et le langage de programmation java, et d'enrichir nos connaissances sur les outils de modélisation d'un système d'information en particulier le langage UML.

Enfin, nous souhaitons que ce travail réponde aux attentes de l'organisme d'accueil.

Bibliographie

[1] : UML pour l'analyse d'un SI « 2^{ème} édition » de : **Chantal Morley**.

Maison d'édition : **DUNOD** en **2000**.

[2] : Manuel de gestion de l'entreprise **Cevital Agro_industriel**.

[3] : Manuel de gestion de l'unité de traitement d'eau minéral **Lalla Khedidja**.

[4] : mémoire de fin d'étude : Développement d'une application workflow en utilisant un moteur open source. Cas : « **suivi de l'octroi d'un crédit bancaire aux particuliers** » Au niveau de la CNEP Banque de Tizi-Ouzou

Réalisé par : Katia TOUBAL et Soraya DJENNADI., année : 2015 /2016

[5]: mémoire de fin d'étude « **Développement d'une application web pour la société algérienne (SAA)** » .

Réalisé par : Belkacem Ali et Ammar Marzouk, année : 2009/2010

Autres memoires :

- Mémoire de fin d'étude de l'UMMT0 « **Conception et réalisation d'une plateforme de messagerie** »

Réalisé par : M^{elle} CHAMEK Lynda et M^{elle} CHAOU Samira, année : 2006/2007

- Mémoire de fin d'étude de l'UMMT0 « **Conception et réalisation d'une application client/ serveur 3- tiers sous oracle** » cas : **gestion des ressources humaines de la direction générale de SONATRACH** ».

- Sites web :

Réalisé par : TEBANI Hakim, année :2006/2007.

- <https://www.developpez.net/forums/u1244244/>

- Comment ça marche encyclopédie informatique, site web www.commentcamarche.net

Annexes

1. Notion sur l'orienté objet :

Introduction :

Aujourd'hui, il existe deux principaux modèles de représentations des systèmes. Le modèle fonctionnel et le modèle objet (ou de « l'orienté-objet », l'OO). Dans l'approche fonctionnelle les programmes sont composés d'une série de fonctions qui assurent certains services. Il s'agit d'une approche logique, cohérente et intuitive de la programmation. Cette approche a un avantage que certain appelle factorisation des comportements (c'est-à-dire que pour créer une fonction d'une application, rien ne vous empêche d'utiliser un autre ensemble de fonctions déjà écrites).

Mais cette approche a ses défauts comme par exemple une maintenance complexe en cas d'évolution d'une application (une simple mise à jour de l'application à un point donné peut impacter en cascade sur d'autres fonctions de l'application). Donc la mise à jour sera alors retouchée l'application dans sa globalité.

L'approche fonctionnelle n'est pas adaptée au développement d'applications qui évoluent sans cesse et dont la complexité croît continuellement. Donc nous allons voir l'approche objet qui a été inventée pour faciliter l'évolution d'applications complexes.

1.1. Les concepts de base de l'approche objet :

a) Objet :

Un objet est une abstraction d'un élément du monde réel. Il possède des informations, par exemple nom, prénom, adresse etc., et se comporte suivant un ensemble d'opérations qui lui sont applicables.

De plus, un ensemble d'attributs caractérisent l'état d'un objet, et l'on dispose d'un ensemble d'opérations (les méthodes) qui permettent d'agir sur le comportement de l'objet. Ce dernier est le cœur de cette approche. Tout objet donné possède deux caractéristiques : Son état courant (attributs) et Son comportement (méthodes).

b) Classe :

On appelle classe la structure d'un objet, c'est-à-dire la déclaration de l'ensemble des entités qui composeront un objet. Les objets de même nature ont en général la même structure et le même comportement. La classe factorise les caractéristiques communes de ses objets et permet de les classer. Un objet est l'instance d'une classe, et une classe est un type de données abstrait, caractérisé par des propriétés (ses attributs et ses méthodes) communes à ses objets et un mécanisme permettant de créer des objets ayant ces propriétés. Le concept de classe permet donc de regrouper des objets de même nature

c) Héritage :

L'héritage est un principe propre à la programmation orientée objet, permettant de créer une nouvelle classe à partir d'une classe existante. Le nom d'"héritage" (ou parfois dérivation de classe) provient du fait que la classe dérivée (la classe nouvellement créée) contient les attributs et les méthodes de sa superclasse (la classe dont elle dérive).

L'intérêt majeur de l'héritage est de pouvoir définir de nouveaux attributs et de nouvelles méthodes pour la classe dérivée, qui viennent s'ajouter à ceux et celles héritées. Par ce moyen on crée une hiérarchie de classes de plus en plus spécialisées

1.2. Les avantages de l'approche objet :

En fait, on peut dire que la POO est une façon de développer une application qui consiste à représenter (on dit également « modéliser ») une application informatique sous la forme d'objets, ayant des propriétés et pouvant interagir entre eux. La modélisation orientée objet est proche de la réalité ce qui fait qu'il sera relativement facile de modéliser une application de cette façon. De plus, les personnes non-techniques pourront comprendre et éventuellement participer à cette modélisation.

Cette façon de modéliser les choses permet également de découper une grosse application, généralement floue, en une multitude d'objets interagissant entre eux. Cela permet de découper un gros problème en plus petits afin de le résoudre plus facilement.

Utiliser une approche orientée objet facilite également de faire la maintenance du logiciel. Plus le temps passe et plus une application est difficile à maintenir. Il devient difficile de corriger un code sans tout casser ou d'ajouter des fonctionnalités sans provoquer une régression par ailleurs. L'orienté objet nous aide ici à limiter la casse en proposant une approche où les modifications internes à un objet n'affectent pas tout le reste du code, grâce notamment à l'encapsulation.

Un autre avantage de la POO est la réutilisabilité. Des objets peuvent être réutilisés ou même étendus grâce à l'héritage. C'est le cas par exemple de la bibliothèque de classes du Framework .NET. Cette bibliothèque nous fournit par exemple tous les objets permettant de construire des applications graphiques. Pas besoin de réinventer toute la mécanique pour gérer des fenêtres dans une application, le Framework .NET sait déjà faire tout ça. Nous avons juste besoin d'utiliser un objet « fenêtre », dans lequel nous pourrions mettre un objet « bouton » et un objet « zone de texte ». Ces objets héritent tous des mêmes comportements, comme le fait d'être cliquable ou sélectionnable.

Conclusion

L'approche orientée objet permet de modéliser une application sous la forme d'interactions entre objets.

- Les objets ont des propriétés et peuvent faire des actions.
- Ils masquent la complexité d'une implémentation grâce à l'encapsulation.
- Les objets peuvent hériter de fonctionnalités d'autres objets s'il y a une relation d'héritage entre eux.