

Chapitre 1 : Contexte général du projet

Cette première partie du rapport donne une vision globale du projet, sa problématique ainsi que son cadre général

1. Problématique générale du projet

Une des tendances les plus en vue et qui concerne tous les secteurs de développement, est l'informatisation. Depuis l'apparition de l'informatique et son introduction dans le monde économique, les entreprises et les entités publiques aspirent à optimiser et à rendre fiable la gestion de leurs structures internes.

Un projet est un processus dynamique, qui suppose l'existence d'une volonté partagée, non seulement par le prescripteur sponsor, mais aussi par le réalisateur (chef de projet et équipe projet) et les bénéficiaires de celui-ci. La prise en compte, dès l'amont, de la dimension «conduite du changement» dans le projet est un élément important pour l'appropriation et l'acceptation de ses objectifs.

Le succès du projet se mesure, en effet, à la satisfaction du client et à la qualité du résultat, c'est-à-dire à la conformité du produit, à ce qui est attendu, livré dans le respect du délai imparti et du budget alloué.

Les projets sont susceptibles de rencontrer plusieurs risques :

- Risque de dépassement de budget.
- Risque de dépassement des délais.
- Risque d'abandon du projet.

Et le problème le plus courant dans la gestion de projet se situe souvent au niveau de la communication entre les différents intervenants (parties prenantes). Le rôle de chaque acteur doit être précisé, afin que toutes les énergies se traduisent en valeur ajoutée.

Un projet consomme des ressources humaines, techniques et financières... Ces ressources doivent être gérées, afin d'éviter tout gaspillage. A l'instar de toute entreprise, une SSII veut améliorer ses procédures de gestion des projets, pour disposer d'une vision globale des différentes études et leur état d'avancement respectif.

Avant d'entamer une étude détaillée du projet, il nous a fallu réaliser une étude de l'existant pour déterminer la démarche préconisée par l'entreprise pour gérer les projets et limiter l'univers de l'étude.

Les problématiques sont nombreuses et couvrent plusieurs préoccupations :

- Préoccupation en terme de rapidité : Comment accélérer le processus de façon à gagner le maximum de temps.
- Préoccupation en termes d'accessibilité: Comment fournir aux utilisateurs un environnement simple et facile à exploiter.
- Préoccupation en terme d'efficacité: Comment obtenir des résultats à satisfaisants à l'utilisateur.

Pour y remédier nous devrions se doter d'une solution informatique à la fois simple, pratique et robuste.

2. Cadre générale du projet

Le présent travail s'inscrit dans le cadre de la mise en pratique des connaissances acquises lors de notre cursus à l'EMSI (Ecole Marocaine des Sciences de l'Ingénieur). Il ambitionne de parfaire ces connaissances au regard des contraintes réelles du monde de la gestion au sein d'une organisation.

Dans cette perspective, ce document se propose de relater la genèse de mise en place d'une application de gestion de projets et rappeler les principes conducteurs l'ayant jalonné. Il est structuré autour des sections suivantes:

- Objectifs assignés au projet d'informatisation en question ;
- Principes méthodologiques qui ont régi la conduite de ce programme ;

- Présentation de l'étude conceptuelle du système en termes de règles de gestion, dictionnaire de données et diagramme de classes;
- Description des outils de travail et interfaces de l'application.

En guise de conclusion, ce document applique la méthode de gestion de projets, modèle qui se prête à une réutilisation pour des applications similaires.

Chapitre 2 : Etude générale du projet

Dans ce chapitre nous allons présenter le travail d'identification, la planification et l'organisation du projet, en détaillant ainsi les choix méthodologiques du projet

1. Identification du projet

1.1 But du projet

Le présent projet a pour but de réaliser un système de suivi de projets pour une entreprise qui désire avoir une vision bien détaillée de ses ressources humaines, pouvoir initier un projet, le suivre dans le temps en respectant les coûts et le délai et affecter à chaque utilisateur un rôle bien précis.

1.2 Objectifs

L'objectif de l'application sera de constituer un menu permettant d'effectuer des tâches spécifiques selon le profil.

Chaque tâche devra répondre aux besoins de l'organisme comme par exemple établir un suivi afin d'avoir une vision globale de l'avancement du projet, il est aussi recommandé de mettre en place des recherches simples et multicritères pour ensuite effectuer des opérations d'ajout, de suppressions et de modifications sur les informations présentes dans la base de données définit préalablement.

Les principales missions du projet sont :

- Collecte des besoins
- Etude fonctionnelle du projet
- Conception
- Implémentation
- Documentation

Les principaux modules à développer dans ce projet sont :

- Gestion des projets
- Gestion des ressources

- Gestion des taches
- Gestion des suivis

1.3 Livrables du projet

Un livrable est tout résultat, document, mesurable, tangible ou vérifiable, qui résulte de l'achèvement d'une partie de projet ou du projet. Les principales livrables de notre application sont : l'application elle-même et le rapport final du projet.

Nom du livrable	Responsable	Date de remise
Spécifications fonctionnelles	Manager	20/03/10
Spécification techniques	Manager	20/03/10
Code source	Chef de projet	29/04/10
Maquettes du projet	Chef de projet	10/05/10
Rapport de fin d'année	Chef de projet	25/06/10

Tableau 2 : Information sur les livrables du projet

Les délais de validation des différents livrables du projet ne devront pas dépasser une semaine, si aucune remarque n'a été reçue, le document se considère comme accepté manuellement.

2. Planification et suivi

2.1 Cycle de vie du projet

2.1.1 Définition

Le cycle de vie du projet définit les phases qui relient le début d'un projet à sa fin. Par exemple, lorsqu'une organisation identifie une opportunité à laquelle elle voudrait répondre, elle autorise souvent une étude de faisabilité avant de décider si elle doit entreprendre le projet. La définition du cycle de vie du projet peut aider le chef de projet à déterminer s'il doit traiter l'étude de faisabilité en tant que première phase du projet ou en tant que projet séparé et indépendant.

En général, les cycles de vie du projet définissent :

- Le travail technique à exécuter dans chaque phase
- le moment où les livrables doivent être générés dans chaque phase et la manière de les passer en revue, de les vérifier et de les valider
- les participants à chaque phase
- les modalités de maîtrise et d'approbation de chaque phase

Notre projet se compose de phases, elles mêmes découpées en tâches, ou lots de travaux. Chaque lot de travail se caractérise par la production d'un «livrable». Il en est de même pour chaque phase pour laquelle le livrable final validé.

2.1.2 Description des phases

2.1.2.1 Phase d'étude

Le succès de la phase d'étude passe par la réalisation de trois étapes successives :

- Expression du besoin
- Etude d'opportunité
- Etude de faisabilité

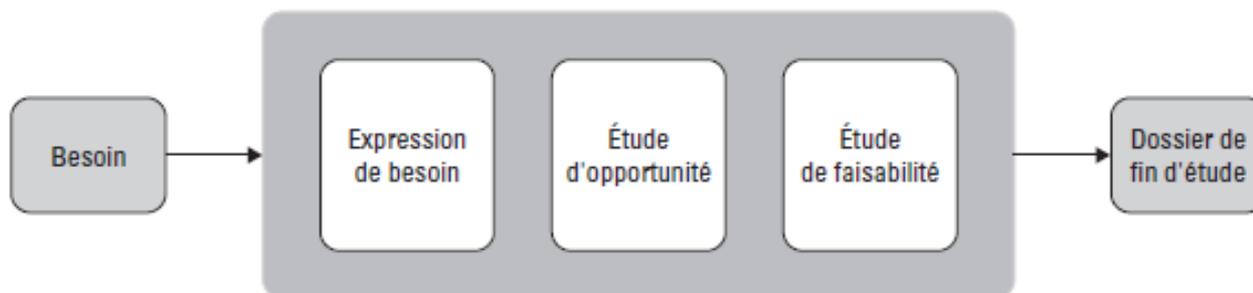


Figure 1 : Les entrées et sorties de la phase d'étude

Pour exprimer les besoins de notre projet, nous allons utiliser le formalisme UML (Unified Modeling Language) des cas d'utilisation. Ces derniers ont été développés par Ivar Jacobson bien avant l'apparition d'UML. Ils ont été intégrés à ce langage de modélisation pour représenter les fonctionnalités du système du point de vue utilisateur. Ils permettent de modéliser des processus métier en les découpant en scénarii. Les cas d'utilisation sont normalement représentés par un schéma, puis enrichis par un document décrivant plus précisément chaque cas ainsi que d'une maquette de l'interface graphique et/ou d'un diagramme d'activités

Le diagramme de cas d'utilisation se compose :

- d'acteurs : ce sont les entités externes (personne humaine ou robot) qui utilisent le système.
- de cas d'utilisation : ce sont les fonctionnalités proposées par le système.

Le succès de la phase d'étude passe par la réalisation de trois étapes successives : L'expression du besoin, l'étude d'opportunité et l'étude de faisabilité ainsi que par l'utilisation d'outils et de livrables appropriés.

2.1.2.2 Phase de conception

Après avoir effectué l'analyse des besoins. Cette partie traite les phases les plus importantes et les plus critiques du cycle de développement. Ce sont l'analyse et la conception.

Il ne s'agit pas de se jeter sur l'écriture du code en omettant de formaliser les besoins des utilisateurs et d'élaborer une architecture robuste et évolutive. D'un autre côté, le but n'est pas de faire de la modélisation pour le plaisir, mais bien de produire le plus rapidement possible un intranet qui satisfasse au mieux ses utilisateurs.

Dans un premier temps, nous identifions les fonctionnalités auxquelles les systèmes de gestion des projets doivent répondre ainsi que les acteurs qui entrent en interaction avec l'application, en déterminant les cas d'utilisation et les diagrammes de séquences système associés. Nous présentons aussi les autres diagrammes qui nous aident au développement de notre application à savoir, les diagrammes des classes participantes, les diagrammes de classes et aussi les diagrammes de séquence. Ensuite on présente la génération de la base de données.

2.1.2.3 Phase de réalisation

Après avoir mené les phases de l'étude des besoins, l'analyse des spécifications et la conception, nous avons entamé la phase de la réalisation. Cette phase porte sur la mise en œuvre de l'intranet. Ainsi il est composé de trois parties : La première passe en revue les différentes technologies et outils de développement utilisés pour la réalisation. La deuxième présente la mise en œuvre effective de l'application en expliquant son architecture technique et met l'accent sur les différents modules réalisés. Enfin la troisième partie résume l'état d'avancement du projet et conclut par quelques perspectives.

2.1.2.4 Phase de tests et mise au point

Pour garantir la crédibilité de notre intranet, nous avons effectué un certain nombre de tests sur deux phases : d'abord des tests de chaque composant de l'intranet (tests unitaires), ensuite des tests d'intégration ont été effectuées pour mesurer l'interopérabilité entre ces composants

2.2 Planning initial du projet

Tache	Durée	Date		Prédécesseur(s)
		Début	Date fin	
Identification du Projet	24 Jours	05/01/2010 08:00	05/02/2010 17:00	
Elaborer Charte du Projet	10 Jours	05/01/2010 08:00	18/01/2010 17:00	
Estimer Budget après Etude	3 Jours	19/01/2010 08:00	21/01/2010 17:00	2
Définition des objectifs	5 Jours	22/01/2010 08:00	28/01/2010 17:00	3
Suivi	6 Jours	29/01/2010 08:00	05/02/2010 17:00	4
Analyse et Conception	42 Jours	08/02/2010 08:00	06/04/2010 17:00	1
Elaborer Cahier de charge	10 Jours	08/02/2010 08:00	19/02/2010 17:00	
Elaborer Règle de gestion	6 Jours	22/02/2010 08:00	01/03/2010 17:00	7
Elaborer Diagramme de use case	5 Jours	02/03/2010 08:00	08/03/2010 17:00	8
Elaborer Diagramme de séquences	10 Jours	09/03/2010 08:00	22/03/2010 17:00	9
Elaborer diagramme de classes	8 Jours	23/03/2010 08:00	01/04/2010 17:00	7;8;9;10
Suivi	3 Jours	02/04/2010 08:00	06/04/2010 17:00	11
Planification du Projet	12 Jours	07/04/2010 08:00	22/04/2010 17:00	6
Découpage du Projet en tache et sous tache	5 Jours	07/04/2010 08:00	13/04/2010 17:00	
Affectation Ressources/Tache	5 Jours	14/04/2010 08:00	20/04/2010 17:00	14
Suivi	2 Jours	21/04/2010 08:00	22/04/2010 17:00	15
Codage des Modules et Réalisation des documents	48 Jours	07/04/2010 08:00	10/06/2010 17:00	6
Conception des pages web	20 Jours	07/04/2010 08:00	04/05/2010 17:00	
Codage module d'ajout	20 Jours	05/05/2010 08:00	01/06/2010 17:00	18
Codage module suppression	3 Jours	02/06/2010 08:00	04/06/2010 17:00	19
Codage module modification	3 Jours	02/06/2010 08:00	04/06/2010 17:00	20SS
Codage module recherche	3 Jours	02/06/2010 08:00	04/06/2010 17:00	21SS
Création des documents	5 Jours	05/06/2010 08:00	10/06/2010 17:00	22
suivi	2 Jours	05/06/2010 08:00	07/06/2010 17:00	23SS;22
Test et validation	3 Jours	08/06/2010 08:00	10/06/2010 17:00	24

Tableau 3 : Planning initial du projet

2.3 Planning réel du projet

Tache	Durée	Date		Prédécesseur(s)
		Début	Date fin	
Identification du Projet	24 Jours	05/01/2010 08:00	05/02/2010 17:00	
Elaborer Charte du Projet	10 Jours	05/01/2010 08:00	18/01/2010 17:00	
Estimer Budget après Etude	3 Jours	19/01/2010 08:00	21/01/2010 17:00	2
Définition des objectifs	5 Jours	22/01/2010 08:00	28/01/2010 17:00	3
Suivi	6 Jours	29/01/2010 08:00	05/02/2010 17:00	4
Analyse et Conception	42 Jours	08/02/2010 08:00	06/04/2010 17:00	1
Elaborer Cahier de charge	10 Jours	08/02/2010 08:00	19/02/2010 17:00	
Elaborer Règle de gestion	6 Jours	22/02/2010 08:00	01/03/2010 17:00	7
Elaborer Diagramme de use case	5 Jours	02/03/2010 08:00	08/03/2010 17:00	8
Elaborer Diagramme de séquences	10 Jours	09/03/2010 08:00	22/03/2010 17:00	9
Elaborer diagramme de classes	8 Jours	23/03/2010 08:00	01/04/2010 17:00	7;8;9;10
Suivi	3 Jours	02/04/2010 08:00	06/04/2010 17:00	11
Planification du Projet	12 Jours	07/04/2010 08:00	22/04/2010 17:00	6
Découpage du Projet en tache et sous tache	5 Jours	07/04/2010 08:00	13/04/2010 17:00	
Affectation Ressources/Tache	5 Jours	14/04/2010 08:00	20/04/2010 17:00	14
Suivi	2 Jours	21/04/2010 08:00	22/04/2010 17:00	15
Codage des Modules et Réalisation des documents	59 Jours	07/04/2010 08:00	25/06/2010 17:00	6
Conception des pages web	20 Jours	07/04/2010 08:00	04/05/2010 17:00	
Codage module d'ajout	20 Jours	05/05/2010 08:00	01/06/2010 17:00	18
Codage module suppression	3 Jours	19/06/2010 08:00	22/06/2010 17:00	19
Codage module modification	3 Jours	19/06/2010 08:00	22/06/2010 17:00	20SS
Codage module recherche	3 Jours	19/06/2010 08:00	22/06/2010 17:00	21SS
Création des documents	3 Jours	23/06/2010 08:00	25/06/2010 17:00	22
suivi	2 Jours	23/06/2010 08:00	24/06/2010 17:00	23SS;22
Test et validation	1 Jour	25/06/2010 08:00	25/06/2010 17:00	24

Tableau 4 : Planning réel du projet

2.4 Analyse des écarts

Parmi les problèmes qui ont causé des retards dans le planning de notre projet :

- Des difficultés dans gestion du temps qui était réparti entre l'élaboration du projet et les examens de fin d'année
- Des difficultés au niveau de l'apprentissage du langage JSP/Servlet
- Incompatibilité de certains logiciels avec notre système d'exploitation Windows 7

3. Organisation du projet

3.1 Maitre d'ouvrage

Nom et prénom	Post
El mehdi BELASLA	Maitre d'ouvrage du projet

- Garantit la conformité des livrables, le respect du périmètre et des délais
- Valide les spécifications techniques
- Valide les actions de l'ensemble des autres intervenants du projet
- Valide le rapport de fin de stage.

- Valide l'ensemble des livrables.
- Définit les objectifs fonctionnels du projet

3.2 Equipe maitrise d'œuvre

Nom et prénom	Post
Mohamed Jacem ENNACIRI	Ingénieur méthode et développement
Soufiane EL OUFIR	Ingénieur méthode et développement

- Responsable de la réalisation du projet
- Responsable des livrables du projet
- Identifie la solution technique
- Teste et livre l'application.
- Planifie et effectue le suivi du projet

Chapitre 3 : Etude des besoins

Dans ce chapitre nous allons présenter l'essentiel du travail de collecte et de spécification fonctionnelle du projet, ainsi que l'architecture logicielle du logiciel cible.

1. Modules du projet

Dans notre application, le module de base est le module de gestion des projets qui consiste à ajouter, modifier ou bien supprimer un projet et aussi désigner un chef de projet pour un projet donné. Ensuite vient le module de tâche qui permet d'ajouter, modifier ou bien supprimer une tâche et aussi affecter les ressources pour cette tâche donnée. Et enfin le module de suivi qui permet de visualiser l'avancement du projet mais aussi les différentes valeurs de charges dont la charge initiale, la charge consommée et la charge réelle.

2. Quelques règles de gestions

Il s'avère toujours nécessaire de faire une étude de l'existant avant d'entamer l'analyse et la conception d'un système, afin de comprendre le problème.

La conception préconise de rédiger le contenu de ce problème sous forme d'un ensemble de règles de gestion.

Les règles de gestion sont fondées sur les exigences du monde réel. Elles sont généralement appliquées par l'intermédiaire des contraintes de vérification, des types de données définis par l'utilisateur et l'utilisation adéquate des transactions.

Pour réaliser notre application il faut suivre les règles de gestion suivantes :

RG1 : Un projet a un seul chef de projet.

RG2 : Un chef de projet peut avoir plusieurs projets.

RG3 : Un projet peut avoir plusieurs personnes qui travaillent dessus.

RG4 : Une personne travaille sur un seul projet à un moment donné.

RG5 : Un projet est décomposé en plusieurs phases.

RG6 : Une phase fait partie d'un seul projet.

RG7 : Une phase est réalisée par plusieurs personnes.

RG8 : Une personne peut participer à plusieurs phases.

3. Dictionnaire de données

Nom attribut	Type	Description
Budget_Projet	double	Budget associer à un projet
Budget_Tache	float	Budget associer à une tache
budgetDepense	double	Budget dépensé à la date de contrôle
Charge_Consumé	varchar(50)	Charge d'heure(s) consommée(s) pour le projet ou une tache
Charge_Initial	varchar(50)	Charge estimée pour le projet ou la tache
dateD_affectation	datetime	Date d'affectation de la recherche
dateDebutInitial	varchar(20)	Date de début initiale d'une tache
dateDebutReelle	varchar(20)	Date de début réelle d'une tache
dateEmbauche	varchar(20)	Date d'embauche d'une ressource humaine
dateFinInitiale	date	Date de fin initiale d'une tache
dateFinReelle	date	Date de fin réelle d'une tache
dateLivvable	int(11)	Date de remise d'un livrable
description_Projet	varchar(254)	Description d'un projet
description_Tache	varchar(254)	Description d'une tache
disponibilite_Ressource	varchar(254)	Disponibilité d'une ressource
login	varchar(254)	Identifiant d'un utilisateur
matricule	varchar(254)	Matricule d'une ressource
nbrHeureEcoule	int(11)	Nombre d'heure(s) travaillée(s)
nomPhase	varchar(50)	Nom d'une phase
nomProjet	varchar(254)	Nom d'un projet
nomRessource	varchar(254)	Nom d'une ressource
nomTache	varchar(254)	Nom d'une tache
numAffectTache	varchar(254)	Numéro d'affectation d'une tache
numAv	varchar(254)	Numéro d'avancement
numLivvable	varchar(254)	Numéro d'un livrable
numPersonne	varchar(254)	Numéro d'une personne
numPhase	varchar(254)	Numéro d'une phase
numPr	varchar(254)	Numéro d'un projet
numRessource	varchar(254)	Numéro d'une ressource
numTache	varchar(254)	Numéro d'une tache
password	varchar(254)	Mot de passe d'un utilisateur
phase	varchar(254)	Numéro d'une phase en cours de réalisation
poste	varchar(254)	Le poste occupé par une ressource humaine
prixAchat	decimal(10,0)	Prix d'achat de la ressource matérielle
salaire	double	Salaires d'une ressource humaine
taux_affectation	int(11)	Taux d'affectation d'une ressource à une tache

Tableau 5 : Dictionnaire de données

4. Spécification générale

4.1 Identification des acteurs

L'application est séparée en deux groupes d'utilisateurs : le chef de projet qui veille sur l'organisation du projet et un manager qui gère tout l'ensemble.

- Chef de projet : Sa tâche consiste à définir les données du projet, le décomposer en phases, désigner les membres de personnel qui travaillent sur le projet, répartir les tâches, affecter chaque équipe à une phase selon les compétences et la formation tout en respectant les critères de délai, de coût et de qualité.
- Manager : Sa tâche consiste à ajouter, supprimer et modifier des projets et aussi ajouter des ressources.
- Ressource : Est un acteur qui effectue la saisie de l'avancement des tâches.

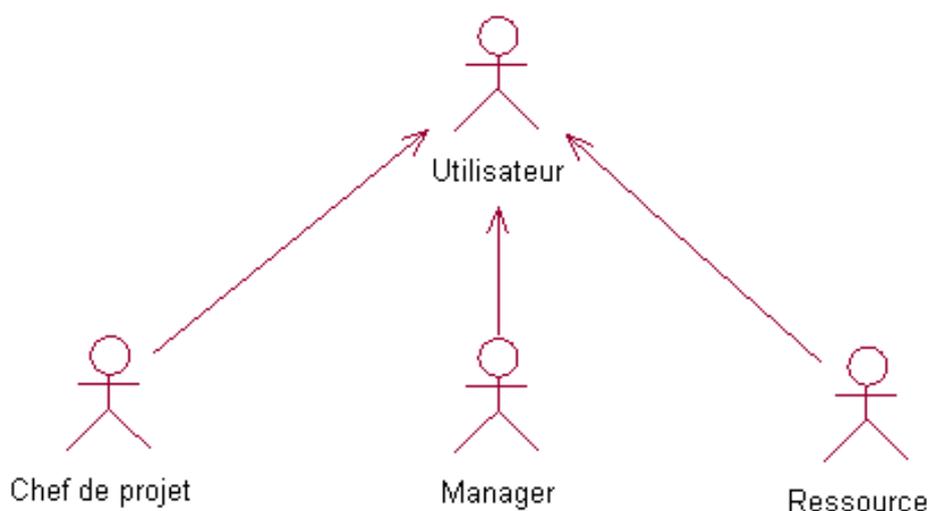


Figure 2 : Les différents acteurs du projet

4.2 Diagramme de cas d'utilisation général

Dans notre projet, nous avons opté pour 4 blocs de packages. Le premier représente le bloc de gestion des projets manipulé par un manager ainsi que par le chef de projet, le manager gère l'ajout, la modification et la suppression d'un projet et le chef de projet gère l'affectation des ressources à un projet ainsi que la décomposition d'un projet en tâches. Le deuxième représente le bloc de gestion des ressources manipulé aussi par le manager. En troisième lieu, le bloc de gestion des tâches manipulé par le chef de projet. Et finalement le quatrième bloc qui représente le bloc de gestion de suivi des projets qui est manipulé par le chef de projet et aussi les ressources selon le schéma suivant :

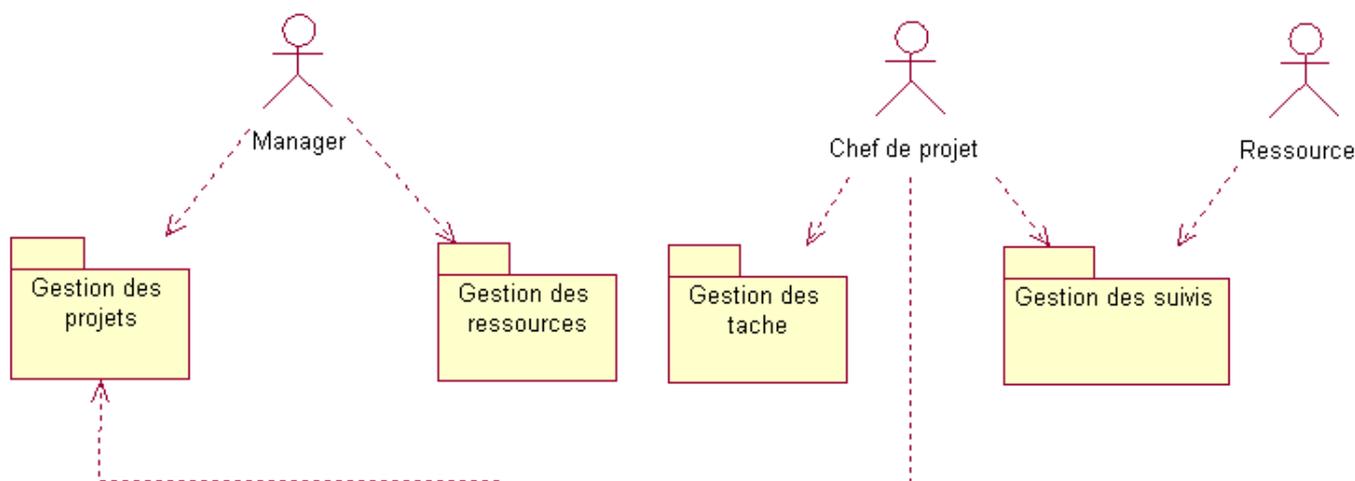


Figure 3 : Les 4 blocs de packages

4.2.1 Bloc de gestion des projets

Dans ce bloc est inclus deux diagrammes de cas d'utilisation de gestion des projets, le premier qui est manipulé par le manager et qui permet l'ajout, la suppression et la modification d'un projet, et le deuxième manipulé par le chef de projet qui permet l'affectation des ressources à un projet, la gestion des tâches et la clôture d'un projet. Ainsi que les scénarios de gestion des projets, couvrant les scénarios suivants :

- Ajouter projet
- Supprimer projet
- Modifier projet
- Affecter des ressources
- Gérer les tâches
- Clôturer un projet

Premier cas d'utilisation	Gestion des projets
Acteurs	Manager
Pré-conditions	Authentification
Post-conditions	-

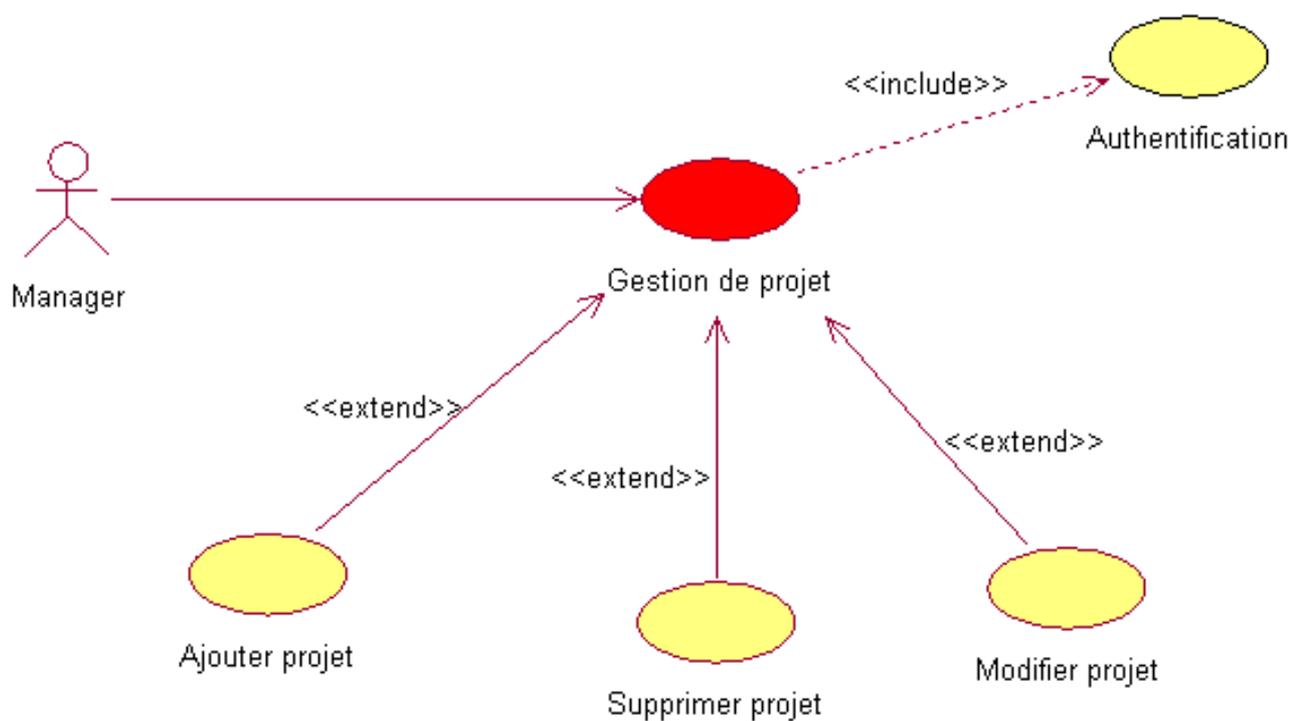


Figure 4 : Diagramme de cas d'utilisation de gestion des projets manipulé par le manager

Deuxième cas d'utilisation	Gestion des projets
Acteurs	Chef de projet
Pré-conditions	Authentification
Post-conditions	-

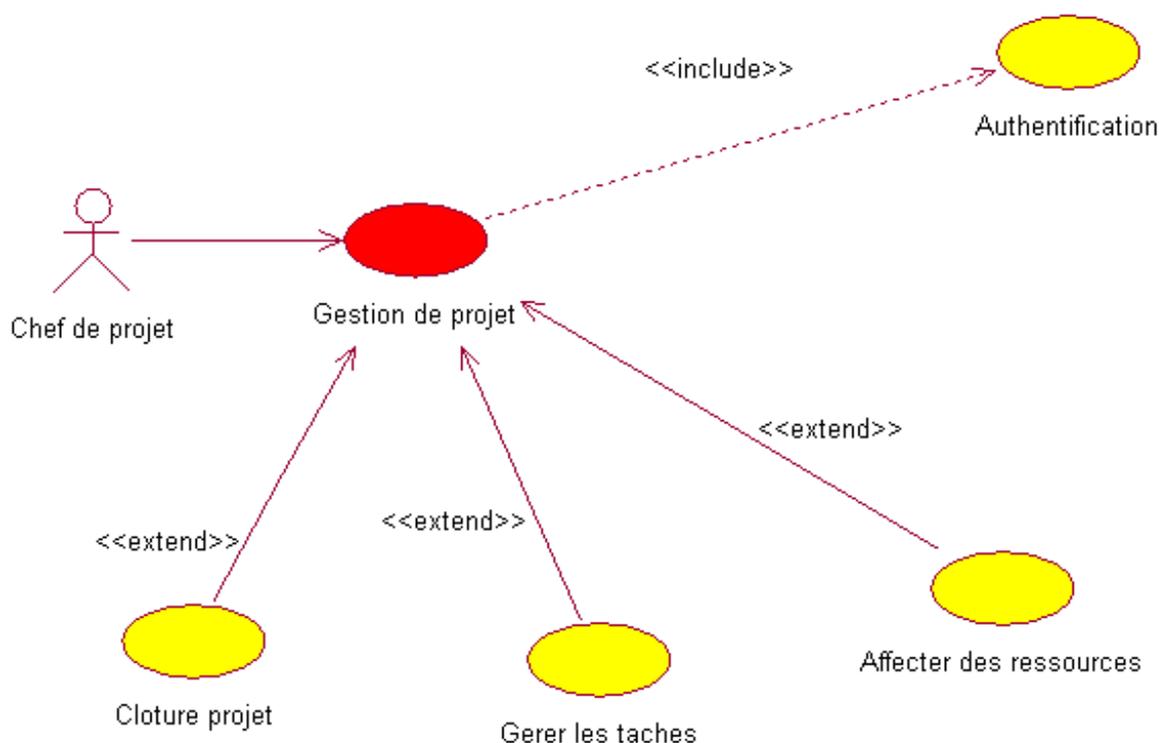


Figure 5 : Diagramme de cas d'utilisation de gestion des projets manipulé par le chef de projet

Le diagramme de séquence principal de ce bloc est le scénario d'ajout d'un projet qui permet l'ajout d'un projet à la base de données en donnant quelques informations sur le projet comme le nom, le budget, la date de début et fin ...

Scénario principal	Ajouter un projet
Acteurs	Manager
Pré-conditions	Authentification
Post-conditions	-

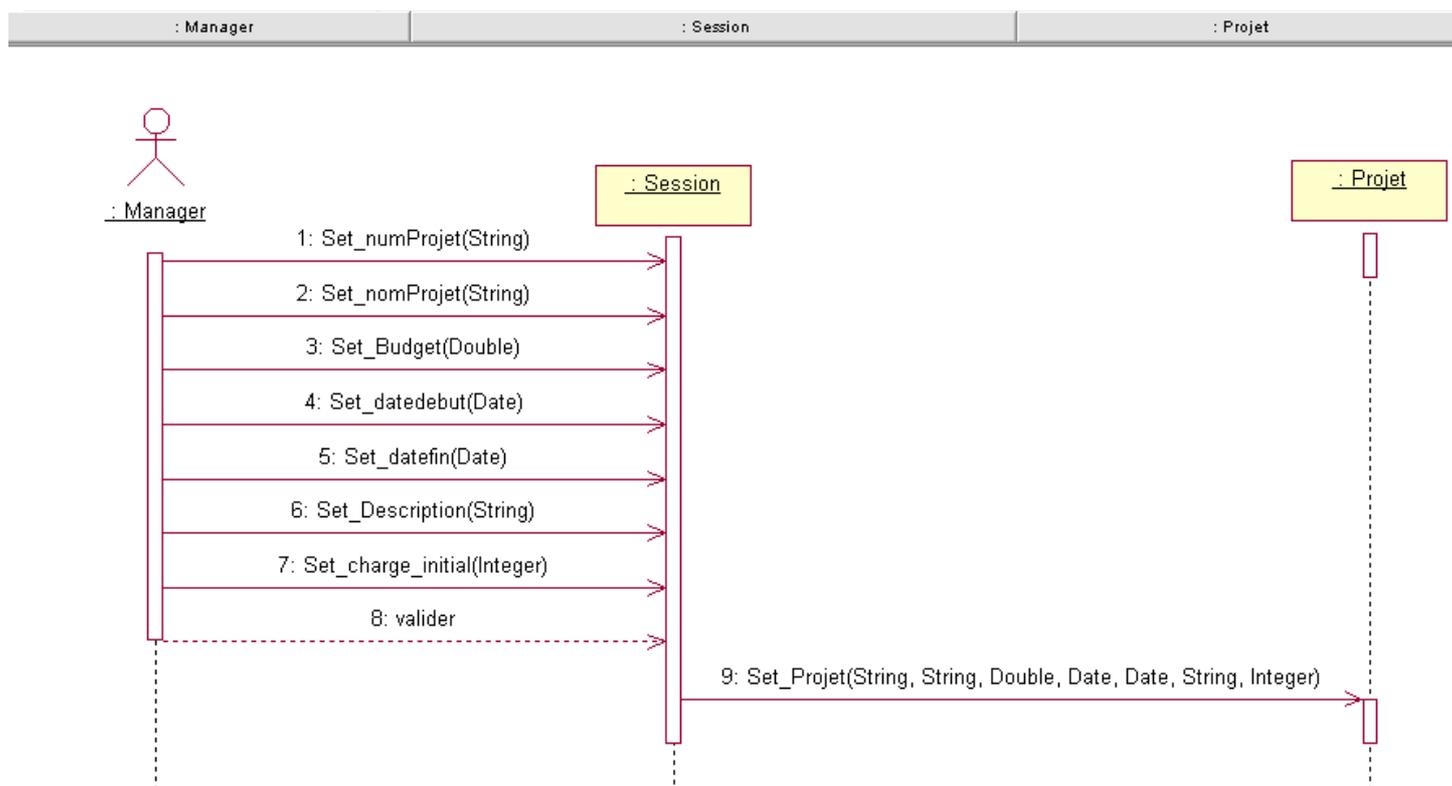


Figure 6 : Diagramme de séquence d'ajout d'un projet

4.2.2 Bloc de gestion des ressources

Dans ce bloc est inclus le diagramme de cas d'utilisation de gestion des ressources, manipulé par le manager et qui permet l'ajout, la suppression et la modification d'une ressource. Ainsi que les scénarios de gestion des ressources, couvrant les scénarios suivants :

- Ajouter ressource

- Supprimer ressource
- Modifier ressource

Cas d'utilisation	Gestion des ressources
Acteurs	Manager
Pré-conditions	Authentification
Post-conditions	Disponibilité des ressources

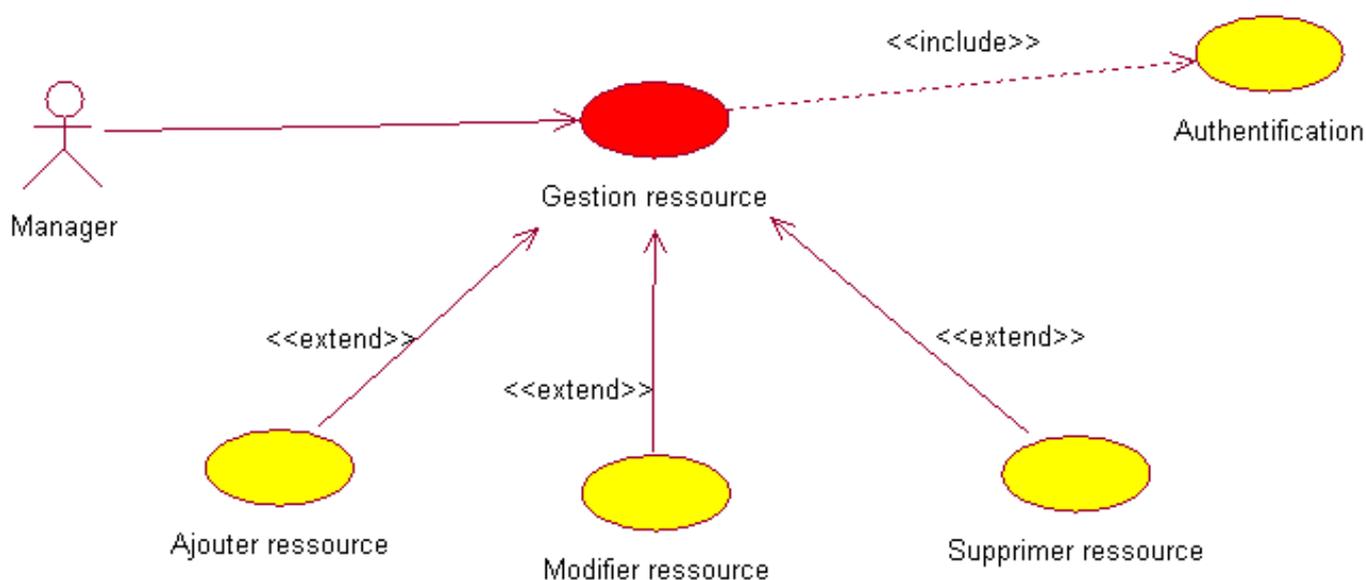


Figure 7 : Diagramme de cas d'utilisation de gestion des ressources

Le diagramme de séquence principal de ce bloc est le scénario de modification d'une ressource qui permet la modification d'une ressource.

Scénario principal	Modifier une ressource
Acteurs	Manager
Pré-conditions	Existence d'une ressource
Post-conditions	-

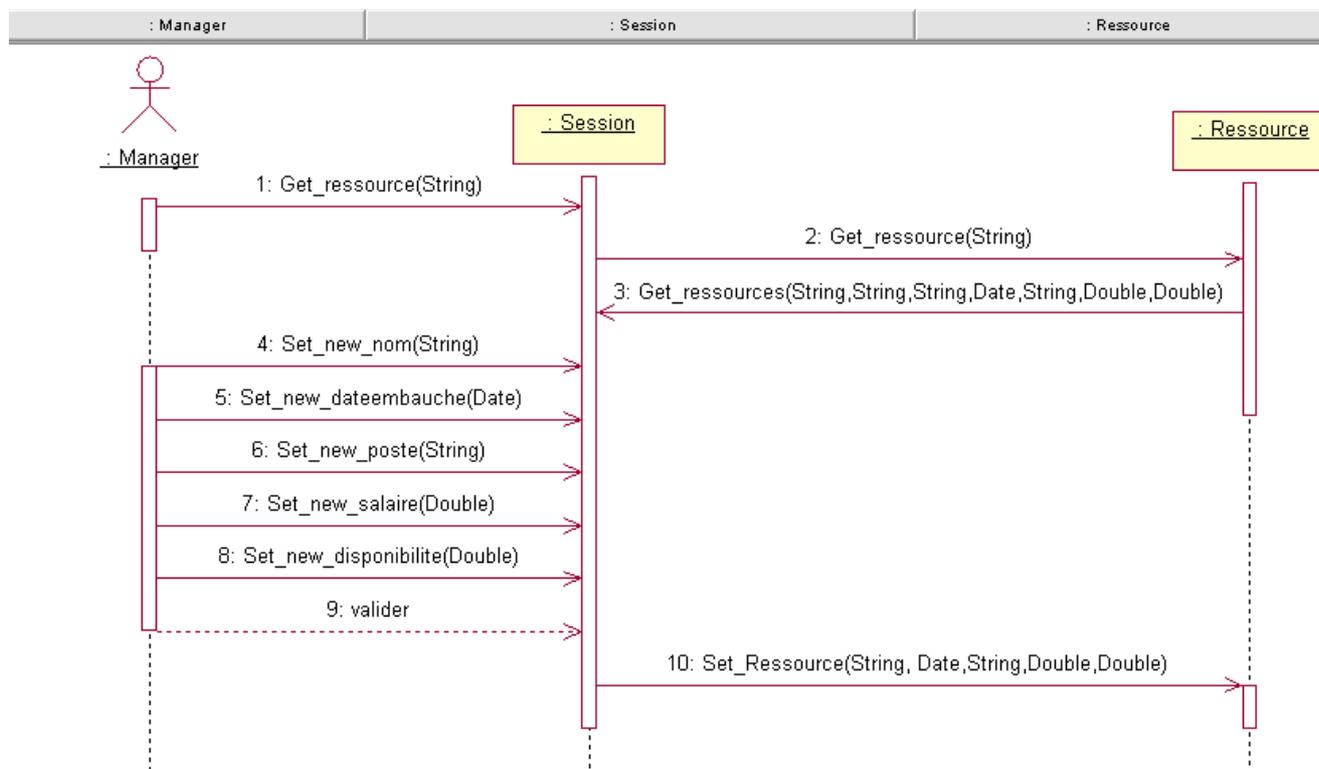


Figure 8 : Diagramme de séquence de modification d'une ressource

4.2.3 Bloc de gestion des taches

Dans ce bloc est inclus le diagramme de cas d'utilisation de gestion des taches, manipulé par le chef de projet et qui permet l'ajout, la suppression et la modification d'une tache. Ainsi que les scénarios de gestion des taches, couvrant les scénarios suivants :

- Ajouter tache
- Supprimer tache

- Modifier tache

Cas d'utilisation	Gestion des taches
Acteurs	Chef de projet
Pré-conditions	Authentification
Post-conditions	-

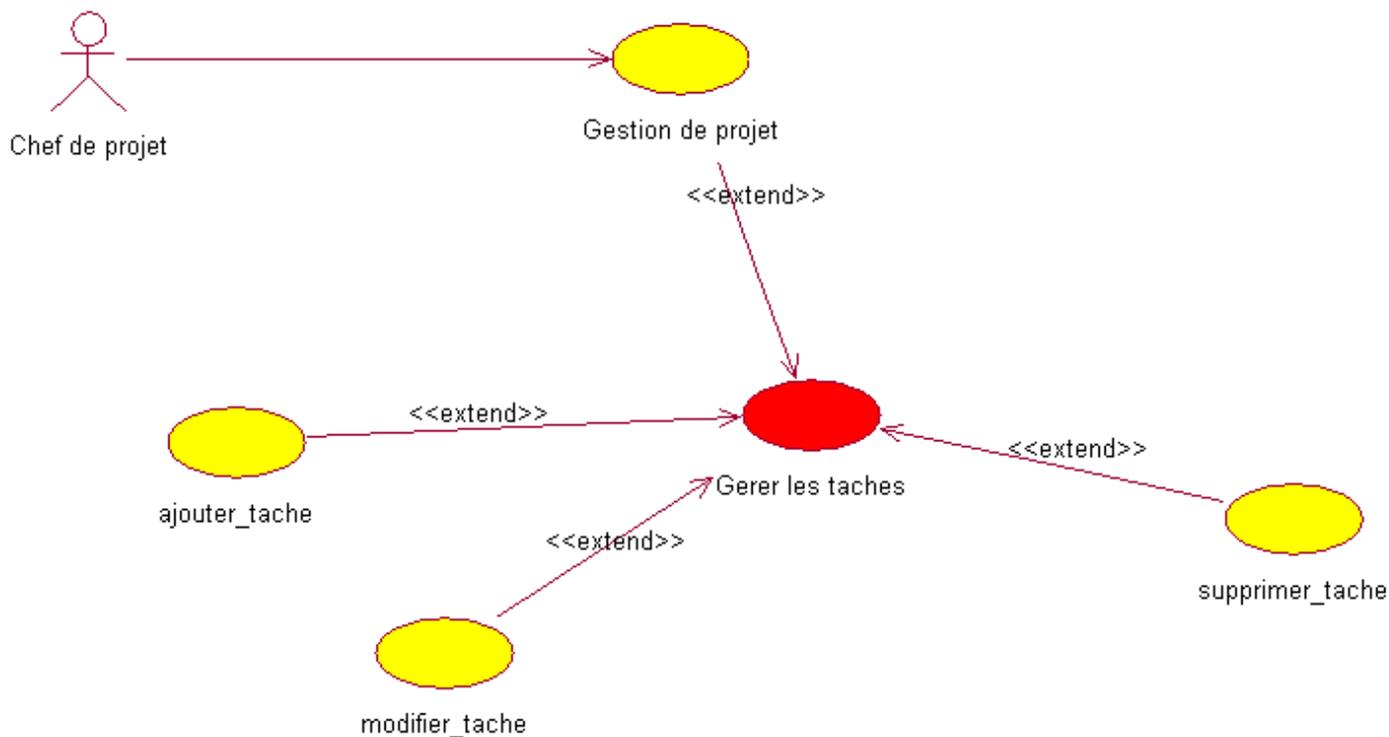


Figure 9 : Diagramme de cas d'utilisation de gestion des taches

Le diagramme de séquence principal de ce bloc est le scénario de suppression d'une tache qui permet la suppression d'une tache.

Scénario principal	Supprimer une tache
Acteurs	Chef de projet
Pré-conditions	Existence d'une tache
Post-conditions	-

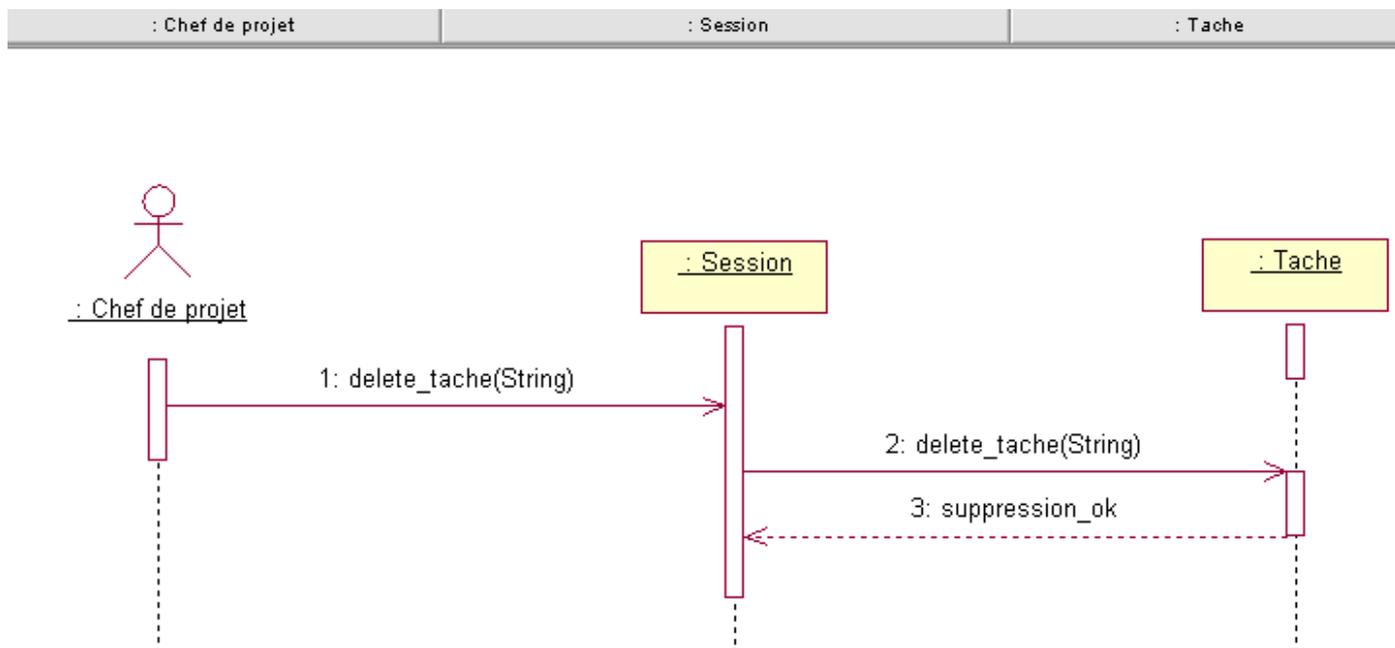


Figure 10 : Diagramme de séquence de suppression d'une tache

4.2.4 Bloc de gestion des suivis

Dans ce bloc est inclus le diagramme de cas d'utilisation de gestion des suivis, qui permet aux ressources l'ajout d'un suivi d'une tache ou d'un projet à un projet, et permet aussi au chef de projet la consultation d'un suivi d'une tache ou d'un projet d'un projet. Ainsi que les scénarios de gestion des suivis, couvrant les scénarios suivants :

- Ajouter suivi d'une tache ou d'un projet
- Consulter suivi d'une tache ou d'un projet

Cas d'utilisation	Gestion des suivis
Acteurs	Chef de projet, ressource
Pré-conditions	Existence d'un projet ou d'une tache
Post-conditions	-

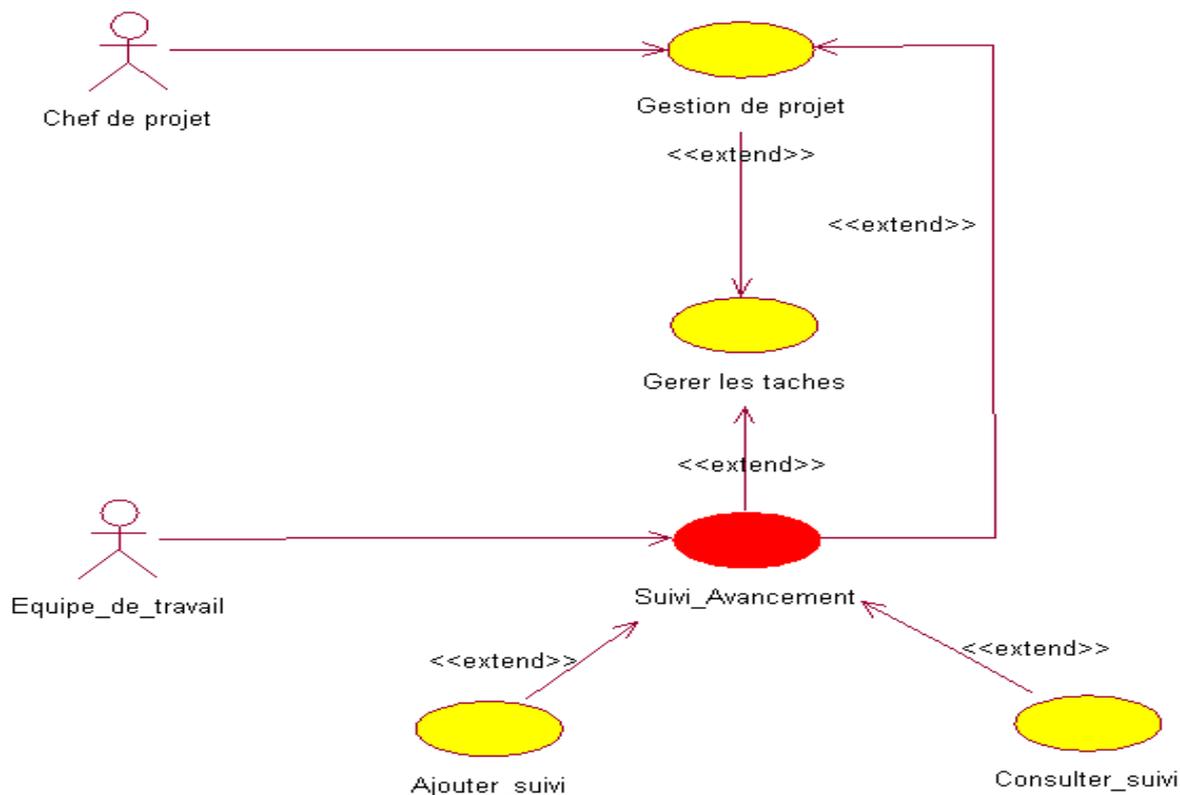


Figure 11 : Diagramme de cas d'utilisation des suivis des taches

Dans la partie du suivi, nous avons essayé de développer 2 scénarios de diagramme de séquences. Le premier concerne la saisie du budget dépensé selon le nombre des heures travaillées par l'équipe de travail qui se présente comme suit :

Scénario principal 1	Saisie de l'état de l'avancement
Acteurs	Ressource
Pré-conditions	Existence d'une tache
Post-conditions	-

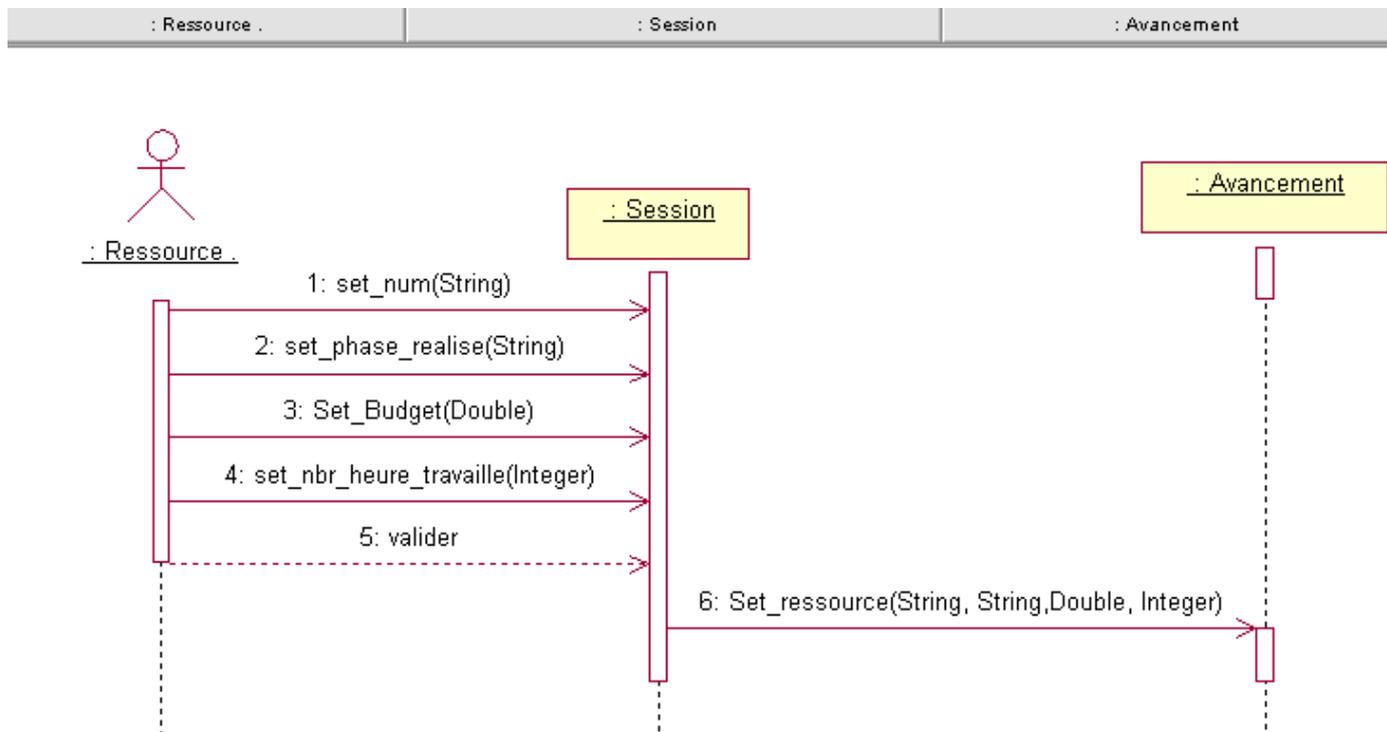


Figure 12 : Diagramme de séquence de la saisie de l'état de l'avancement

Le deuxième scénario de diagramme de séquence, concerne l'exploitation des données saisies par l'équipe de travail envers le chef de projet qui a la possibilité d'augmenter la durée du projet ou d'une tâche si le pourcentage d'avancement est supérieur à 90% et le nombre d'heures de travail restant est inférieur à 16h. Se diagramme se présente comme suit :

Scénario principal 2	Suivi de l'avancement
Acteurs	Ressource
Pré-conditions	Existence d'un suivi
Post-conditions	-

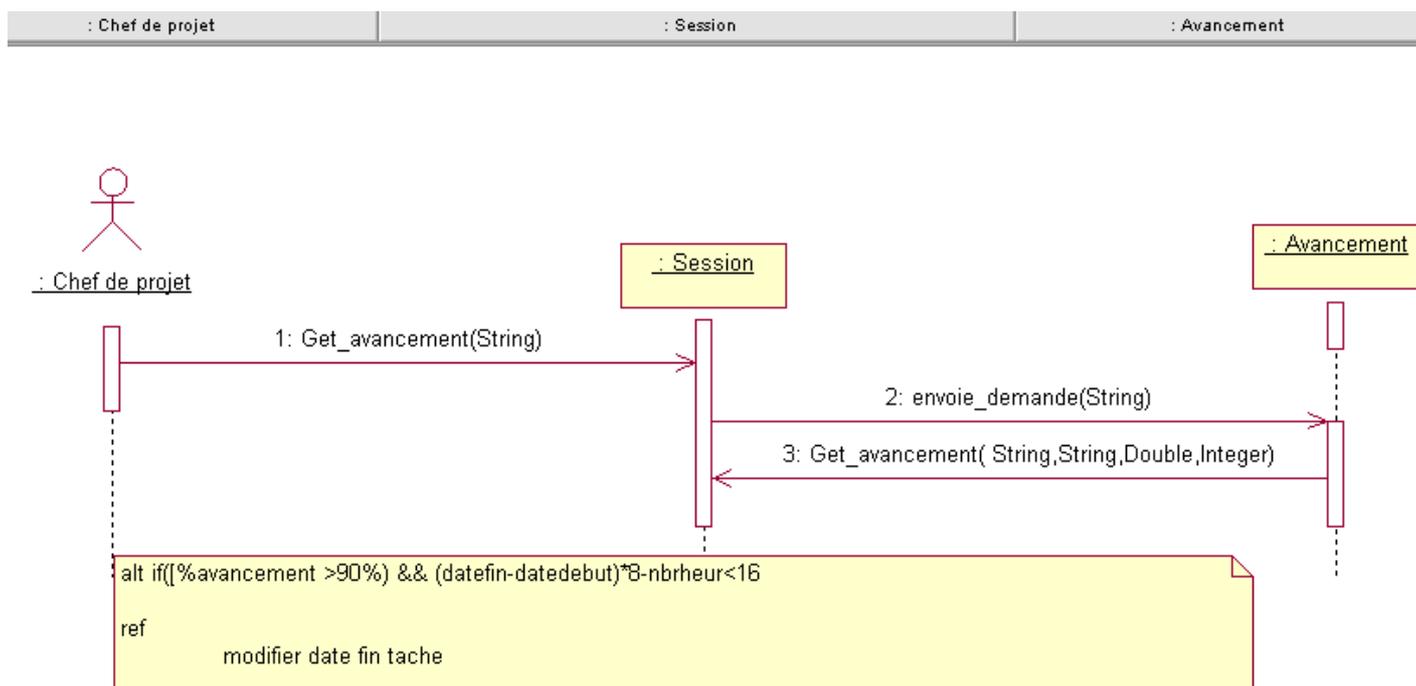


Figure 13 : Diagramme de séquence du suivi de l'avancement

4.3 Diagramme de classes

La modélisation statique des classes permet de représenter la vue statique d'un système grâce à un ensemble de diagrammes constitués des concepts tels que les classes (propriétés et opérations), les relations entre elles, les interfaces ou les collaborations.

La notion de classe est la description d'un ensemble de types d'objets qui partagent les mêmes attributs, les mêmes opérations, les mêmes relations et la même sémantique.

Une classe est définie par :

- Un nom : chaîne de caractères commençant par une majuscule,
- Un ensemble d'attributs,
- Un ensemble d'opérations et
- Un ensemble de responsabilités.

Le diagramme de classes de notre application est le suivant :

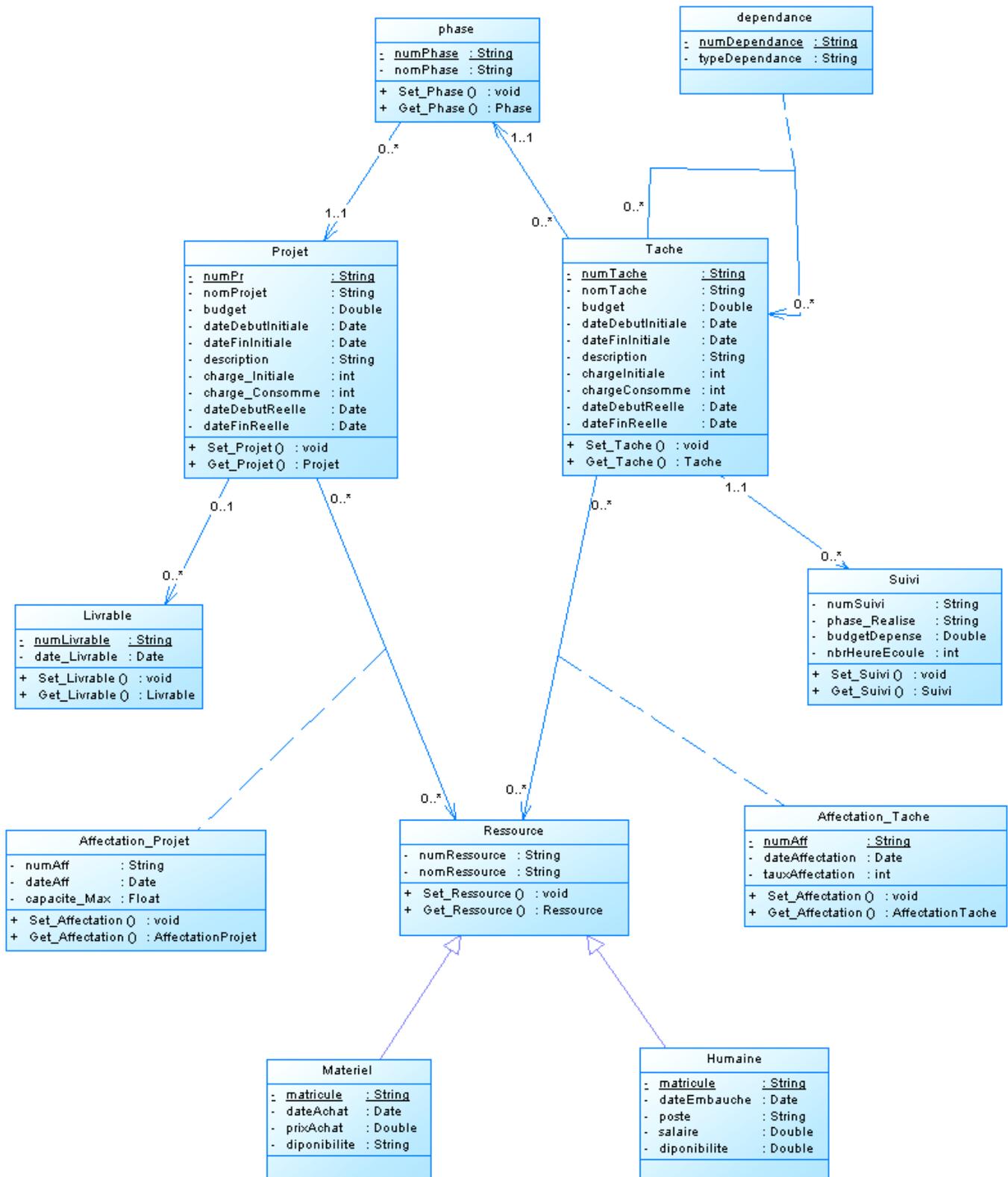


Figure 14 : Diagramme de classes de l'application

5. Architecture de navigation de l'application

L'application réalisée comporte plusieurs pages qui sont enchaînées, et permet le passage d'une page à une autre selon un ordre donné comme ce qui est présenté par les graphes ci-dessous :

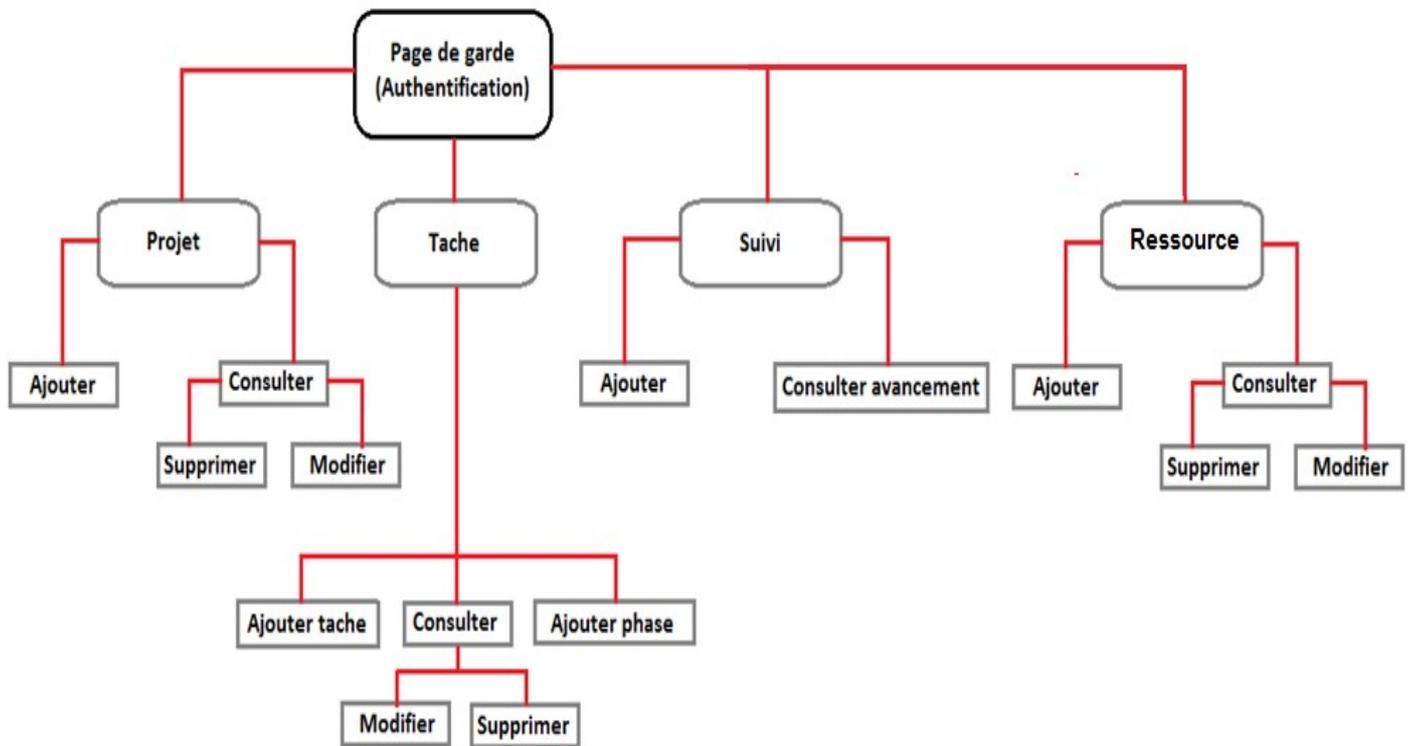


Figure 15 : Architecture du site

Chapitre 5 : Réalisation et test

Dans ce chapitre nous allons présenter le détail de l'implémentation, commençant par une présentation de différentes technologies utilisées dans le développement, et terminant par une présentation de quelques interfaces de l'application réalisée

1. Architecture du traitement

Comme toute application réalisée en J2EE, notre projet doit être décomposé en deux parties :

- La partie « Client » contenant les interfaces et le contenu WEB sous forme de pages .JSP stockées dans le dossier « WebRoot ».
- La partie « Serveur » invoquée par une JSP suite à une requête du client, cette partie est constituée d'un ensemble de Servlets qui sont stockées dans un package dans le dossier source « src ».

2. Architecture 3 tiers

L'architecture 3-tiers est basée généralement sur l'utilisation de navigateurs HTTP. Pour rendre interactif l'utilisation des bases de données, il est nécessaire d'avoir une application serveur capable :

- d'y accéder et d'exécuter les instructions voulues
- de générer les pages HTML permettant à l'utilisateur de visualiser le résultat de sa demande.

Pour cela nous avons utilisé JSP qui est un langage libre et qui fournit une excellente prestation dans ce domaine.

JSP fonctionne généralement sur un serveur apache; comme nous avons déjà cité. Ce dernier reconnaît les pages HTML comportant une extension particulière et les envoie aux servlets. Ces pages sont en fait des pages HTML comportant entre les balises `<%jsp` et `%>` un langage de script qui est interprété et permet de forger dynamiquement la page HTML à renvoyer. Le client ne voit que le résultat de sa requête.

D'un point de vue général quelques points importants sont à souligner pour l'architecture 3-tiers :

- Le client qui n'a donc que des fonctions d'affichage ne fait que des requêtes vers le serveur, aucun calcul n'est effectué par le client. Les résultats de ses requêtes sont ensuite affichés.
- C'est le serveur qui va effectuer tous les calculs ou faire des requêtes vers d'autres serveurs additionnels (eg vers des SGBD).

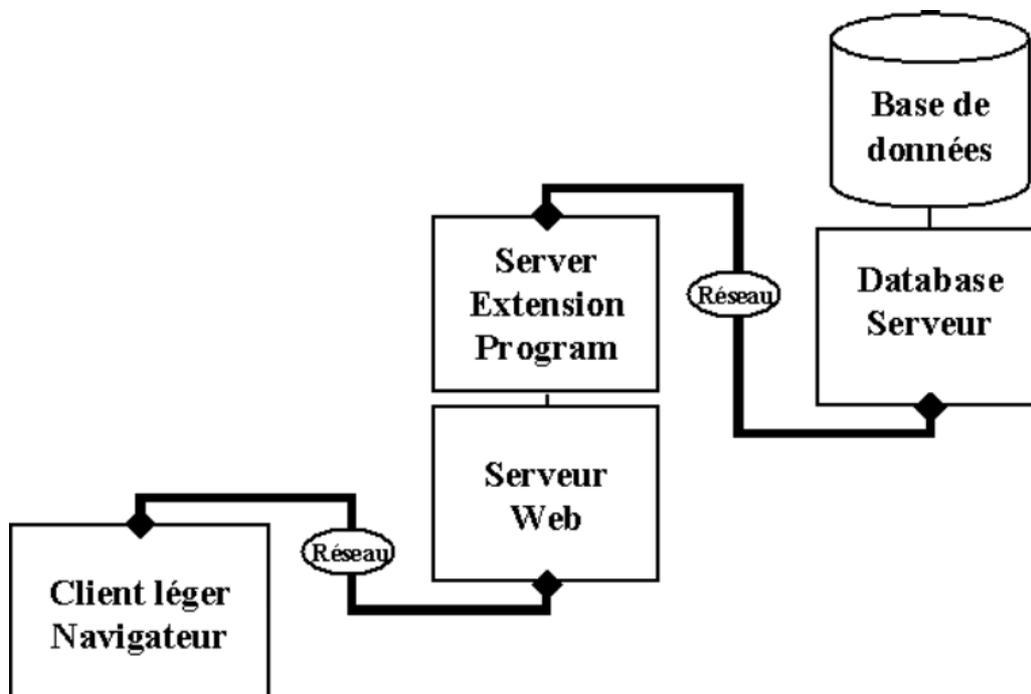


Figure 16 : Représentation de l'architecture 3 tiers

3. Outils et frameworks de développement

3.1 Logiciels

3.1.1 Power AMC

PowerAMC est un [logiciel](#) de modélisation. Il permet de modéliser les traitements informatiques et leurs [bases de données](#) associées.

Il nous a permis de concevoir notre MCD pour fournir une représentation formelle des données nécessaires à la gestion de notre site, et de le transformer par la suite en un modèle physique directement exploitable par un [SGBDR](#) (qui est Mysql dans notre cas).

3.1.2 DreamWeaver

Adobe Dreamweaver est un [éditeur de site web](#) de type [WYSIWYG](#) (What you see is what you get), il fut l'un des premiers éditeurs [HTML](#) de type tel écrit tel écran.

Dreamweaver a évolué avec les technologies de l'internet. Il offre aujourd'hui la possibilité de concevoir des feuilles de style. Les liaisons avec des bases de données ont également été améliorées ainsi que le chargement des fichiers sur les [serveurs](#) d'hébergement.

Il propose en outre l'utilisation de modèles imbriqués de pages web, selon un format propriétaire.

Il est utilisé avec des langages web dynamiques ([ASP](#), [PHP](#)), et permet ainsi de développer des applications dynamiques sans connaissance préalable des langages de programmation.

Il nous a été utile pour composer et ajuster visuellement le résultat voulu, puisque c'est un logiciel de mise en page qui affiche directement à l'écran à quoi ressemblera le résultat final.

3.2 Langages de programmation

3.2.1 HTML

Langage de description de document et non pas de programmation il sert à éditer des pages virtuelles c'est un langage universel pour communiquer sur le web .l'information sera par la suite transportée ainsi par la toile de réseau local de l'organisme grâce à un programme appelé navigateur ou browser

Pour transiter le plus rapidement possible sur les lignes téléphoniques on a adopté un format de texte très compact mais aussi plus sophistiqué est le bon vieux format de texte pur et dur, sans fioritures du Bloc notes par exemple ;en plus du texte ;il faudra inclure des instructions pour le browser .ces instructions seront différenciées du texte par les signes sup et inf .ces instructions s'appellent des tags ou des balises.

3.2.2 JSP/SERVLET

Les JSP (Java Server Pages) sont un standard permettant de développer des applications Web interactives, c'est-à-dire dont le contenu est dynamique. C'est-à-dire qu'une page web JSP (repérable par l'extension .jsp) aura un contenu pouvant être différent selon certains paramètres (des informations stockées dans une base de données, les préférences de l'utilisateur,...) tandis que page web "classique" (dont l'extension est .htm ou .html) affichera continuellement la même information.

Il s'agit en réalité d'un langage de script puissant (un langage interprété) exécuté du côté du [serveur](#) (au même titre que les scripts [CGI](#),[PHP](#),[ASP](#),...) et non du côté client (les scripts écrits en [JavaScript](#) ou les applets [Java](#) s'exécutent dans le navigateur de la personne connectée à un site).

Les JSP sont intégrables au sein d'une page Web en [HTML](#) à l'aide de balises spéciales permettant au serveur Web de savoir que le code compris à l'intérieur de ces balises doit être interprété afin de renvoyer du code HTML au navigateur du client.

Ainsi, les Java Server Pages s'inscrivent dans une [architecture 3-tier](#), ce terme compliqué signifie qu'un serveur supportant les Java Server Pages peut servir d'intermédiaire

(on parle généralement de serveur applicatif) entre le navigateur du client et une base de données (on parle généralement de serveur de données) en permettant un accès transparent à celle-ci. JSP fournit ainsi les éléments nécessaires à la connexion au système de gestion de bases de données, à la manipulation des données grâce au [langage SQL](#).

4. Quelques interfaces de l'application

4.1 Interface d'authentification

Gestion de projets

Bienvenue, veuillez vous authentifier



Login

Mot de passe

[About Us](#) | [Site Map](#) | [Privacy Policy](#) | [Contact Us](#) | ©2010 EL OUFIR && ENNACIRI Entreprises

Cette interface permet à l'utilisateur de s'authentifier pour accéder aux différentes ressources disponibles sur l'application Gestion de projets.

4.2 Interface d'ajout d'un projet

Ajout projet Accueil | Log-out

Projet | Tache | Suivi | Ressource

Ajouter | Consulter

Num Projet	<input type="text" value="Pr1"/>
Nom du projet	<input type="text" value="Projet1"/>
Budget Projet	<input type="text" value="20000.667"/>
Date début Initial	<input type="text"/>
Date fin Initial	<< < [Fermer] > >> Juin 2010 Dim Lun Mar Me Jeu Ven Sam 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30
Description	
Charge Initiale	



[About Us](#) | [Site Map](#) | [Privacy Policy](#) | [Contact Us](#) | ©2010 EL OUFIR && ENNACIRI Entreprises

Cette interface permet au manager de créer un nouveau projet avec les différentes informations sur un projet

4.3 Interface de consultation d'un projet

Consulter Projet Accueil | Log-out

Projet Tache Suivi Ressource

NumProjet	NomProjet	Budget Projet	DateDebutInitial	DateFinInitial	Description	ChargeInitial	
Pr1	Projet1	20000.667	04-06-2010	11-06-2010	1er Projet	80	 
pr3	Gestion3	0	01-10-2010	01-12-2010	Film	52	 

[About Us](#) | [Site Map](#) | [Privacy Policy](#) | [Contact Us](#) | ©2010 EL OUFIR && ENNACIRI Entreprises

Cette interface montre les détails d'informations sur les différents projets existant ainsi que la possibilité de modification ou de suppression

Conclusion générale

Pour mener à bien ce projet, nous avons dû enrichir et approfondir nos connaissances autant du point de vue de la conception, et en matière de base de données ainsi du point de vue programmation en utilisant les technologies WEB comme JSP, HTML et Servlet.

Nous étions donc amenées à créer une ou plusieurs page(s) Web pour chaque utilisateur en tenant compte de son profil et de sa fonction. La programmation de ces fonctions a rendu aisé l'accomplissement des tâches des différents acteurs.

Par ailleurs, reste quelques modules à finaliser, tel que le suivi des projets, et l'affectation des ressources aux projets.

En somme, la réalisation de ce projet nous était bénéfique et intéressante puisque nous avons eu l'opportunité d'employer nos connaissances et progresser notre savoir.