

# RAPPORT DE STAGE

Projet de fin de deuxième année  
Année Universitaire 2012-2013



Période du stage

Du 01/07/2013 au 30/09/2013

Type : Stage Ingénieur

Durée : 3 mois

Filière : Génie informatique

## Sujet

Interaction entre Microsoft Projets Server 2010 et Google Apps



Réalisé par : Elmetni Hamza

Encadré par : Ms. Mohcin Dria  
Ms. Abdellah Yassine

ENTREPRISE :



Atlas Bottling  
Company  
(Coca Cola)  
Tanger

ECOLE :



L'Ecole Nationale  
des Sciences  
Appliquées  
de Tétouan



# Remerciements

Je tiens à remercier mon maître de stage, Monsieur **Abdellah yassine** , qui m'a si bien accueilli et s'est montré disponible. Merci à lui d'avoir répondu sans limites aux questions que j'ai pu lui poser. La rencontre avec un futur collègue compte pour beaucoup dans le parcours de formation qui est le mien.

Il me semble évident, de remercier aussi Monsieur **Mohcin Dria**, qui a fourni un travail extraordinaire pour m'aider et me soutenir dans la préparation et le bon déroulement de ce stage ingénieur ,Il m'a permis d'effectuer ce stage dans d'excellentes conditions, tout en étant toujours disponible et à mon écoute.

Merci également à l'équipe de System d'information **d'Atlas boltting company tanger** , qui m'ont aidé avec leur expériences, leur temps et leur informations et me faire sentir comme je suis un membre de leur équipe

Finalement, j'adresse mes vifs remerciements à tous les cadres et les personnelles **d'Atlas boltting company tanger** qui m'ont beaucoup aidés et pour ses remarques, critiques et suggestions qui ont formulé sur mon travail.

Je profite de cette occasion pour adresser mes remerciements les plus spéciaux et distingués à l'ensemble du corps **enseignant de l'ENSA Tétouan** pour leur efforts remarquables afin de m'offrir une formation solide dans un cadre de respect et de complicité.

Enfin, mon profonde gratitude et mon respect à toute personne qui a contribué de près ou de loin à l'élaboration de ce travail.



# Sommaire

<b>Remerciements .....</b>	<b>3</b>
Résumé du projet .....	7
Introduction générale.....	8
<b>Chapitre 1 : Contexte générale du projet .....</b>	<b>9</b>
1- Introduction générale.....	11
A) Diana holding .....	11
B) Atlas Bottling Company .....	11
2- Problématique.....	16
3- Des concepts liés au projet.....	16
A) Microsoft projet 2010 (application de bureau) .....	16
B) Microsoft projet server 2010.....	17
C) Google apps for business .....	19
D) Google Calendar .....	20
E) Google api .....	21
F) Google Apis Console .....	21
4- Méthodologie du travail.....	23
A) Cycle de vie.....	23
B) Organisation de travail au sein d'entreprise .....	24
<b>Chapitre 2 : Description et analyse du projet .....</b>	<b>25</b>
1- Charte du projet .....	27
A) Introduction de charte de projet .....	27
B) Description du projet .....	27
C) Organisation du .projet .....	33
2- Les cas d'utilisation .....	35
3- Analyse des solutions .....	37
A) La première solution .....	37
B) La deuxième solution .....	38
C) La troisième solution .....	39
D) Conclusion .....	40
4- Analyse de la solution .....	41
A) La forme générale de la solution .....	41
B) Les avantages .....	41
C) Les technologies utilisées .....	41
5- La planification du projet .....	43

## **Chapitre 3 : Mise en oeuvre du projet .....45**

<b>1- L'architecture de la solution .....</b>	<b>47</b>
A) Vue contexte .....	47
B) Vue cas d'utilisation .....	48
C) Vue logique .....	49
D) Vue physique .....	52
<b>2- Structure des bases de données .....</b>	<b>53</b>
A) La base de données de Microsoft project server .....	53
B) La base de données de la solution .....	55
C) Les curseurs .....	61
<b>3- Les applications Java de la solution .....</b>	<b>65</b>
A) L'application de La génération du fichier config.xml ....	65
B) L'application principale .....	65
<b>4- La sécurité de la solution .....</b>	<b>69</b>
A) l'authentification de Google Apps .....	69
B) La communication SMTP .....	72
C) La base de données.....	73
D) La sécurité de fichier « config.xml » .....	74
E) La gestion des erreurs .....	74
<b>5- L'interface graphique .....</b>	<b>76</b>
A) l'authentification de Google Apps .....	76
B) La communication SMTP .....	77
C) La base de données .....	79
D) La sécurité de fichier « config.xml » .....	80
E) La gestion des erreurs .....	81
F) L'application des critères ergonomiques .....	81
<b>6- Optimisation de base de données .....</b>	<b>82</b>
A) Les techniques d'optimisation de base de données.....	83
B) Le résultat d'optimisation .....	83
<b>Conclusion .....</b>	<b>85</b>
<b>Listes des figures .....</b>	<b>87</b>
<b>Listes des tableaux .....</b>	<b>89</b>



# Résumé du projet

Le présent document constitue une synthèse de mon stage de la fin d'année (**PFA**) , effectué au sein d' **Atlas Bottling Company de tanger** , une société marocaine filiale du groupe Diana Holding, spécialisée dans l'embouteillage et la distribution des produits de « **The Coca Cola Company** » , Ce stage a pour objectif de développer d'un système pour réaliser l'interaction entre « **Ms project Server 2010** » et « **Google Apps for business** ».

Ce projet entre dans le cadre de migration d'entreprise ABC-tanger de « Microsoft exchange » vers « Google apps » afin de bénéficier tous les avantages proposés par les applications de Google , la migration qui a commencé au début de mois février 2013.

Le stage a été élaboré en cinq grandes étapes, à savoir :

1. L'analyse des besoins et l'étude de faisabilité du projet afin d'établir un cahier de charge.
2. Documentation, le choix de technologie, Détermination de la forme de la solution et finalement planification du projet.
3. Le développement de la solution.
4. Optimisation de la solution.
5. L'intégration de la solution dans le système d'information d'entreprise

Pour le développement du projet, j'ai suivi le cycle de développement en cascade. Pour la modélisation du système, j'ai utilisé le langage UML et merise.

La réalisation du projet a fait appel à l'utilisation java comme le langage de programmation principale, Google api pour communiquer avec les serveurs de Google , SQL server comme le système de gestion de base de données , des notions de Cryptographie , XML et Transact-SQL .

Le présent rapport permet de présenter les différentes étapes par lesquelles le projet a passé dans le but de réaliser le travail qui a été demandé.



# Introduction générale

Dans le cadre des études d'ingénieur, le stage de la fin d'année (PFA) est souvent Synonyme de première expérience réel d'un élève ingénieur dans le monde d'entreprise. Découvrir et vivre la réalité du terrain, prendre conscience de l'organisation et des Contraintes liées à l'entreprise.

Une entreprise aujourd'hui ne peut pas s'en passer de l'informatique comme outil de traitement, de gestion et de manipulation des données, celle-ci grâce à ses outils divers, puissants et innovants assure à l'organisme la performance, la fiabilité, une optimisation de temps et des ressources et une bonne gestion de ces services.

ce projet entre dans le cadre de migration d'entreprise ABC de « Microsoft exchange » vers « Google apps for business » afin de bénéficier tous les avantages proposé par les applications de Google .

Le sujet de projet est de trouver une solution pour réaliser une interaction entre « Microsoft project Server 2010» et « Google Apps for business » .

Cette solution s'agit d'une application qui va être utilisé par le service informatique d'ABC, elle va toucher tous les utilisateur de « Microsoft project Server» d'entreprise ABC , que soit des chefs de projets ou bien des membres des équipes qu'il vont réaliser certain taches .

Ce rapport décrira en détails les différentes étapes du développement de l'application, il s'étendra sur 3 chapitres illustrant le déroulement de mon travail. Le premier chapitre sera consacré à la présentation de l'entreprise Atlas Bottling Company de tanger, l'organisme m'ayant accueilli, ainsi que le contexte général du projet et la définition de certain mot clé du projet . La présentation des études fonctionnelles et l'analyse du projet fera l'objet du deuxième chapitre. Le troisième et le dernier chapitre aborde la conception détaillée et la réalisation du projet. En conclusion, je vais repréciser le travail réalisé et les perspectives envisagées.

# CHAPITRE 1

## Contexte générale du projet

Ce premier chapitre présente la société d'accueil, Atlas Boltting company est une entreprise qui appartient au groupe « diana holding », premièrement on va donner donnons un bref aperçu sur l'organisme diana holding , ensuit une description générale d'ABC tanger et ainsi l'organigramme de cette dernière , ainsi que le contexte général de notre projet , ensuit une description du problématique , puis on va présenter des concepts fondamentaux liées au projet , et on va finir par la méthodologie de travaille.



# 1- Introduction générale

## A) Diana holding



Figure 1 : Logo de Diana Holding

Acteur incontournable de l'industrie agroalimentaire, le groupe Diana Holding ou le groupe Zniber consolide et développe sa présence au Maroc depuis plus de 50 ans. Aujourd'hui le groupe représente un modèle de référence dans l'ensemble de ses métiers (spiritueux, boissons gazeuses, aviculture et agriculture).

Le groupe Diana Holding, qui emploie plus de 6 500 personnes, opère principalement dans la viticulture (les celliers de Meknès), l'importation et la distribution de boissons alcoolisées (Group Ebertec), l'embouteillage des boissons non alcoolisées (ABC), les loisirs (Magic Park à Salé)

## B) Atlas Bottling Company



Figure 2 : Logo d'Atlas Bottling Company

Atlas Bottling Company est une société marocaine filiale du groupe Diana Holding, spécialisée dans l'embouteillage et la distribution des produits de « The Coca Cola Company » (Coca-Cola, Fanta, Sprite, Pom's, Hawaï, Schweppes, Ciel, ...). ABC assure aussi la distribution d'autres produits de Coca Cola dont Aquarius, Burn, Miami, Pulpy, etc.

La société dispose de deux sites de production à Tanger et à Oujda et sa production concerne deux grandes familles de produits : le Verre et le PET (polyéthylène téréphthalate ou conditionnements en plastique)

Elle a actuellement plus de 10000 clients ; les clients sont les personnes susceptibles de revendre les produits Coca à partir de différents points de vente (épicerie, café, restaurant et autres).

## ➤ Historique

En 1947, « THE COCA COLA BOTTLING PLANT OF TANGIER » premier embouteilleur de Coca Cola au Maroc est fondé par l'américain Mr.James Hall. Il installe au Dock Monopolio, premier quartier industriel de Tanger situé sur la baie, une unité d'embouteillage débarquée d'un navire américain. La compagnie s'installe à Casablanca en 1950 et crée la première usine de concentré du Maroc et d'Afrique.

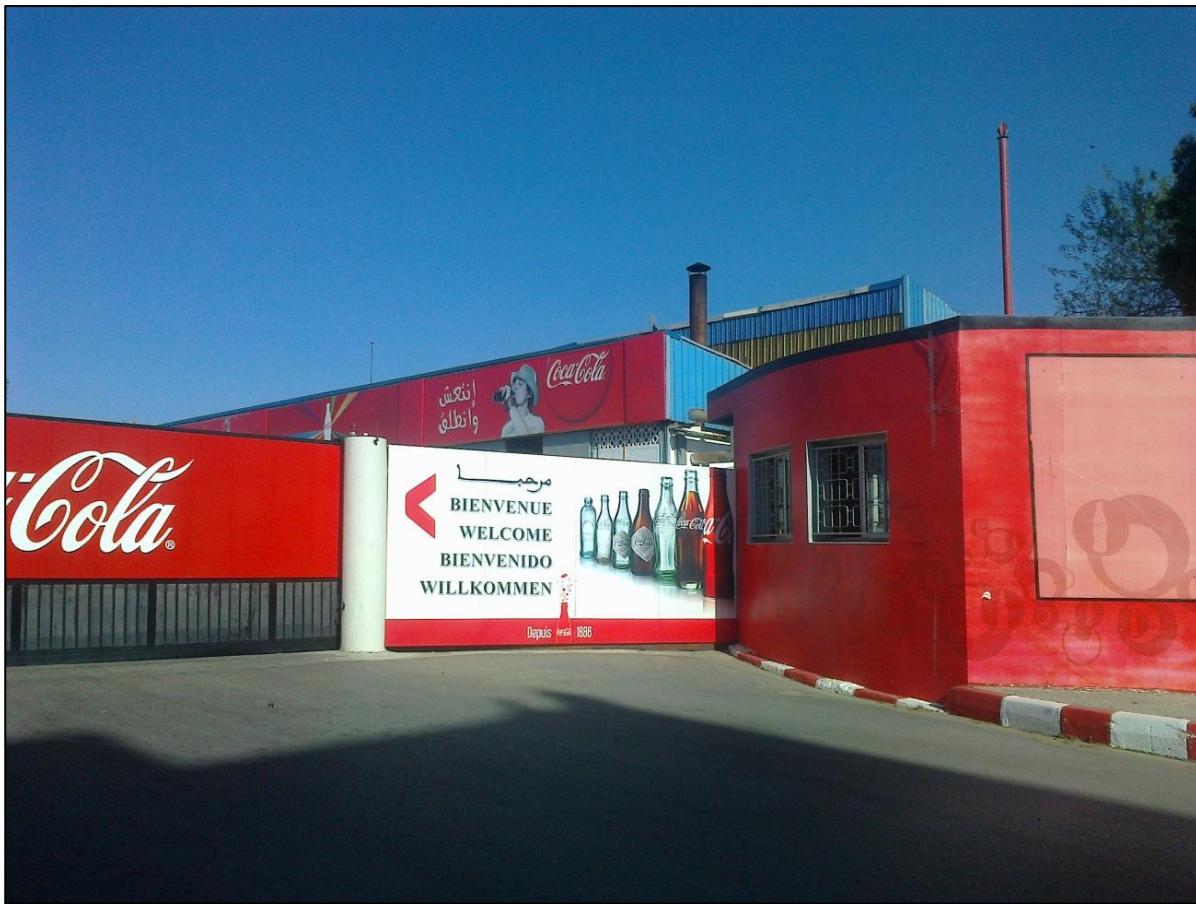


Figure 3 : Une photo de l'extérieur de l'usine d'ABC tanger (2013)

### **Voici les dates saillantes de Atlas Bottling Company-tanger**

- En 1953 Mr. Hall crée la compagnie des boissons Atlas SARL fabricant d'Atlas orange pour l'ensemble du Maroc et fonde ATLAS BOTTLING COMPANY OF TANGIER
- 1974 : ATLAS BOTTLING intègre le groupe ZNIBER qui lance l'extension de la capacité de production en aménageant une nouvelle usine d'embouteillage.
- 1987 : Devant le développement du marché, l'accroissement des conditionnements et le lancement de nouveaux parfums, une nouvelle usine est créée à Oujda avec une ligne verre.

- 1991 à 1995, le marché des boissons gazeuses connaît des changements profonds avec le lancement des conditionnements en plastique ou P.E.T (Polyéthylène Téréphthalate) ATLAS BOTTLING réagit rapidement en mettant en place des investissements conséquents dans de nouvelles lignes spécifiques en plastique. Elle investit dans la fabrication des bouteilles plastiques et des lignes spécifiques au P.E.T
- 1997 ATLAS BOTTLING rachète au groupe SIM ses deux usines Seven-Up de Tanger et Les Boissons de l'oriental à Oujda.
- 1998 L'unité de soufflage d'Oujda est renouvelée.
- 2000 Lancement du système de pré- vente généralisé ensuite en 2001
- 2003 Lancement d'un programme de Management de la Relation Client sur Tanger, étendu ensuite à la zone d'Oujda.

### ➤ **Atlas bottling company en chiffres**

De la création de la société jusqu'aux années 70 les ventes étaient réalisées uniquement dans 2 formats de bouteilles : le standard et la familiale avec une distribution cantonnée aux villes de tanger et téouan exclusivement.

La figure suivante représente Répartition géographique du marché des boissons au maroc .

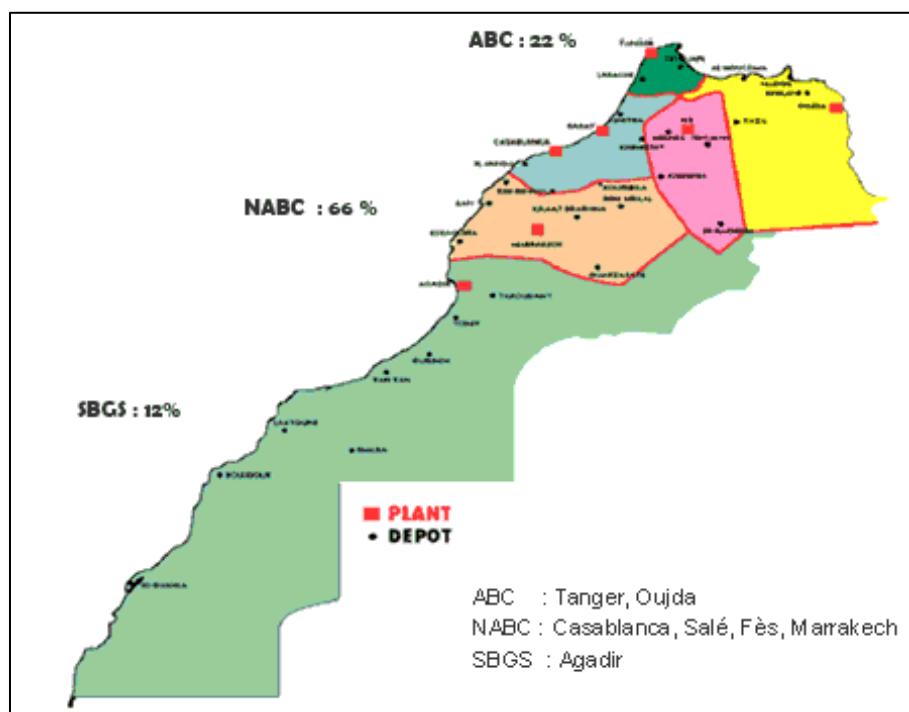


Figure 4 : Répartition géographique du marché des boissons au maroc

Même si Abc tanger couvre une zone géographique assez petit en comparant avec les autres concurrents mais elle prend un pourcentage important de marché des boissons au maroc , il atteint aujourd'hui plus de 22% du marché national .

Le graphe suivant représente la structure du marché national

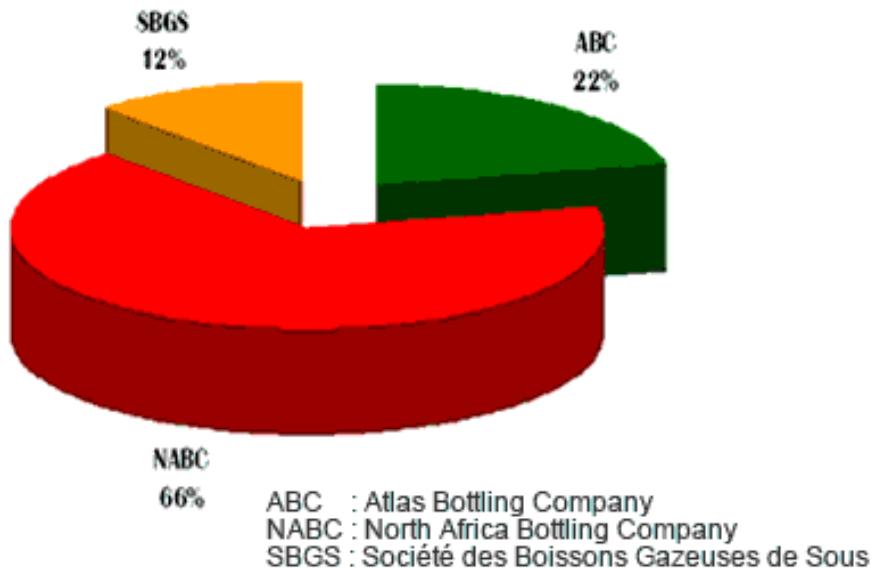


Figure 5 : la structure du marché national du boissons

Le chiffre d'affaires d'Atlas bolting company a connu une évolution rapide ces derniers 5ans jusqu'il atteint aujourd'hui plus de **30.000.000 Dh**

Le graphe suivant représente Evolution du chiffre d'affaires d'ABC en millions de dirhams

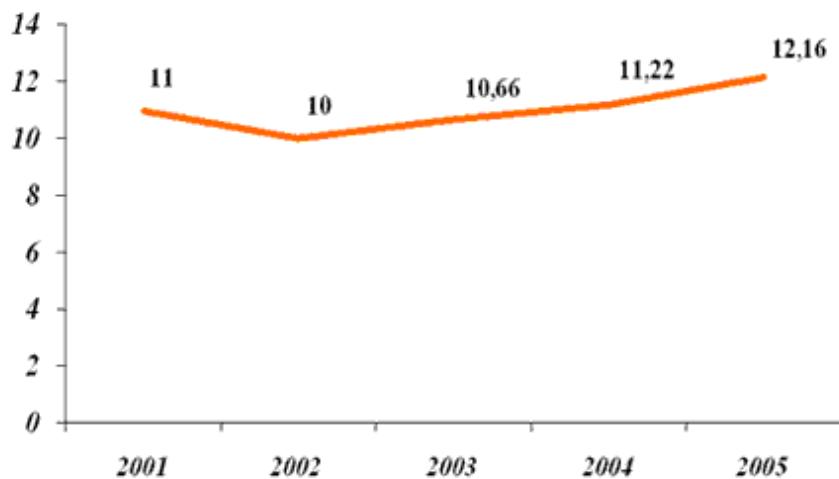


Figure 6 : l'Evolution du chiffre d'affaires d'ABC

➤ **Fiche Signalétique :**

Dénomination sociale	Atlas Bottling Company
Alias	<b>ABC</b>
Forme Juridique :	<b>Société Anonyme</b>
Siège Social :	<b>Lotissement PANAMARIBO, Branes Tanger</b>
Capital Social :	<b>30.000.000 Dh</b>
Détenteur du Capital :	<b>DIANA HOLDING (GROUPE ZNIBER)</b>
Activités de la société :	<b>Fabrication et distribution des boissons gazeuses.</b>
Effectif :	<b>Plus de 1300 personnes</b>

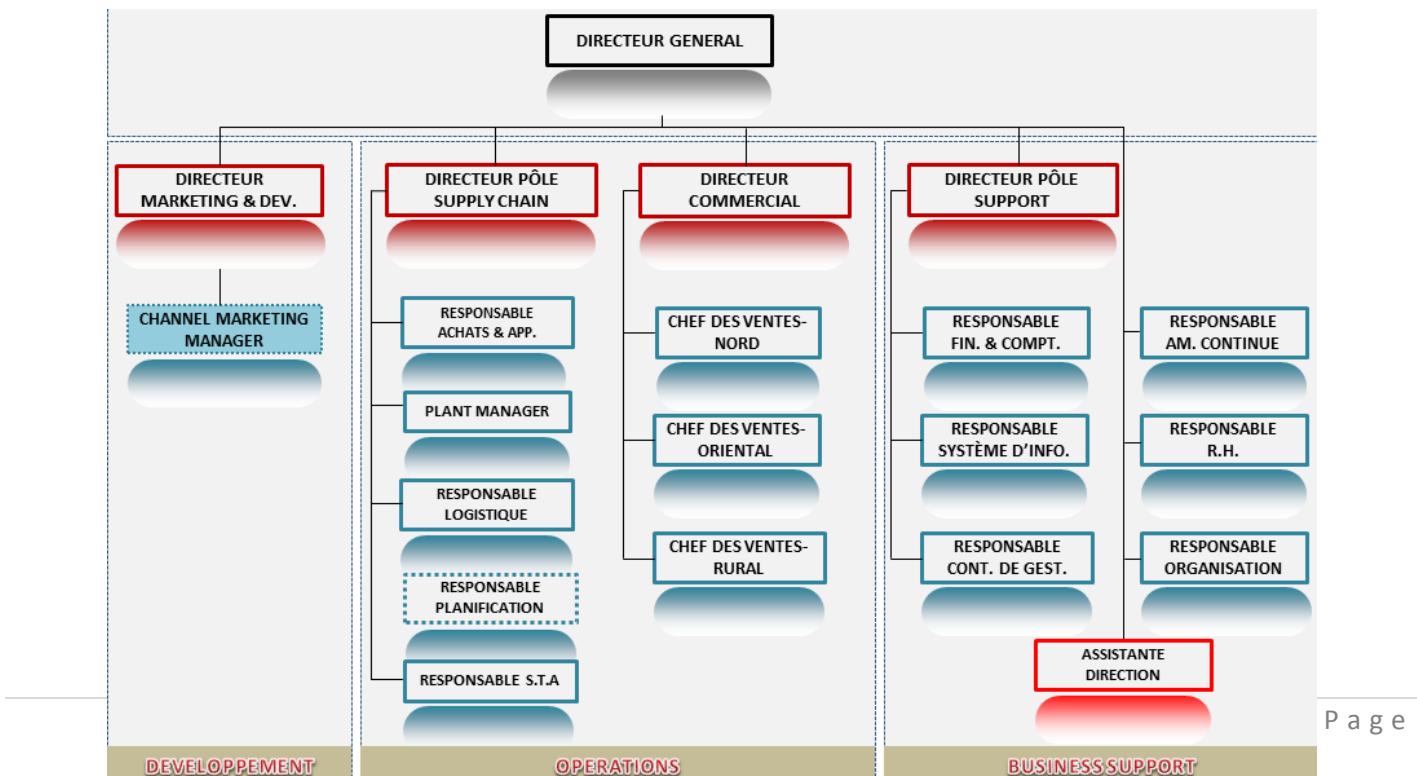
*Tableau 1 : Fiche Signalétique d'ABC*

➤ **Organisation**

ABC est structuré suivant une logique d'exploitations. Au fait, les deux concessions fonctionnent sous la houlette du Directeur Général et à la tête de chacune un Directeur d'exploitation. Les Directions fonctionnelles sont basées au siège à Tanger et coiffent à la fois les deux zones.

Le développement des ressources humaines se trouve au cœur des préoccupations de la Direction Générale. Dans ce sens, un plan de formation annuel est élaboré systématiquement dont l'objectif est de développer les potentialités humaines de tous les niveaux hiérarchiques.

Cette structure est représentée par l'organigramme suivant :



*Figure 7 : l'organigramme d'ABC*

## 2- Problématique

Au début de février 2013 , ABC tanger commence à quitter « Microsoft exchange » et migrer vers « Google apps for business » , pour bénéficier les différent services et applications de « google apps » .

En réalisant cette migration en donnant chaque personnel d'ABC un compte de google apps for business contient ensembles des applications coffrées par google ,

Dans le cadre de cette migration vient ce projet pour réaliser une interaction avec l'outil de gestion des projets « Microsoft Project server 2010 » et « Google apps » (spécialement les deux application « Google calendar » et « Gmail ») .

L'idée générale est : on veut de synchroniser tous les opérations effectuée dans « Microsoft Project server 2010 » avec « Google calendar » ( sous forme des taches ) et Gmail (en forme des alertes au cas d'une retard par exemple ) , cela va permettre les personnels d'ABC d'utiliser leurs comptes de « Google Apps » dans tous les opérations quotidiennes , et par conséquence pousser les procédures d'intégration entière de « Google apps » avec le System de ABC .

vous trouvez plus de détails à propos de cahier de charges du projet dans le **chapitre 2** .

## 3- Des concepts liés au projet

Dans cette section, on va présenter les concepts suivants qui sont directement liés au projet et comment ils fonctionnent :

- a. Microsoft projet 2010 ( l'application de bureau)
- b. Microsoft projet server 2010
- c. Google apps for business
- d. Google Calendar
- e. Google Api
- f. Google Api Console

### A) Microsoft projet 2010 (application de bureau)



Microsoft Project (ou MS Project ou MSP) est un logiciel de gestion de projets édité par Microsoft. Il permet aux chefs de projet et aux planificateurs de planifier et piloter les projets, de gérer les ressources et le budget, ainsi que d'analyser et communiquer les données des projets.

Utilisé aujourd'hui (2011) par plus de 20 millions de chefs de projet, Microsoft Project est le logiciel de gestion de projet le plus utilisé au monde, Plus de 10 000 entreprises

Figure 8 : logo de Microsoft projet 2010 (application de bureau)

ont aussi déployé la version serveur de Microsoft Project, nommée Microsoft Project Server.

## B) Microsoft projet server 2010



Figure 9 : logo de Microsoft projet server 2010

Microsoft Office Project Server est une solution de serveur de gestion de projet faite par Microsoft. Il utilise Microsoft SharePoint comme sa fondation, et prend en charge l'interface de Microsoft Project comme une application client ou via un navigateur Web se connecter à son Project Web App (PWA) composant. Microsoft étend les fonctionnalités de Microsoft Project (application de bureau)

Avec Project Server et Project Web App (PWA, anciennement Project Web Access). Magasins Serveur d'informations de projet Microsoft Project dans une **base de données SQL Server centrale**, protégée contre tout accès non autorisé et la corruption. Un administrateur de projet peut contrôler la sécurité définissant les utilisateurs et les droits d'accès.

Le Centre de projets soutient des rapports à travers une organisation au niveau du projet. Les gestionnaires peuvent descendre dans les détails du projet. Tâches, projets et ressources peuvent avoir des champs personnalisés d'entreprise définis (similaire aux champs personnalisés dans l'application de bureau Microsoft Project).

Le schéma suivant montre comment Microsoft project server fonctionne.

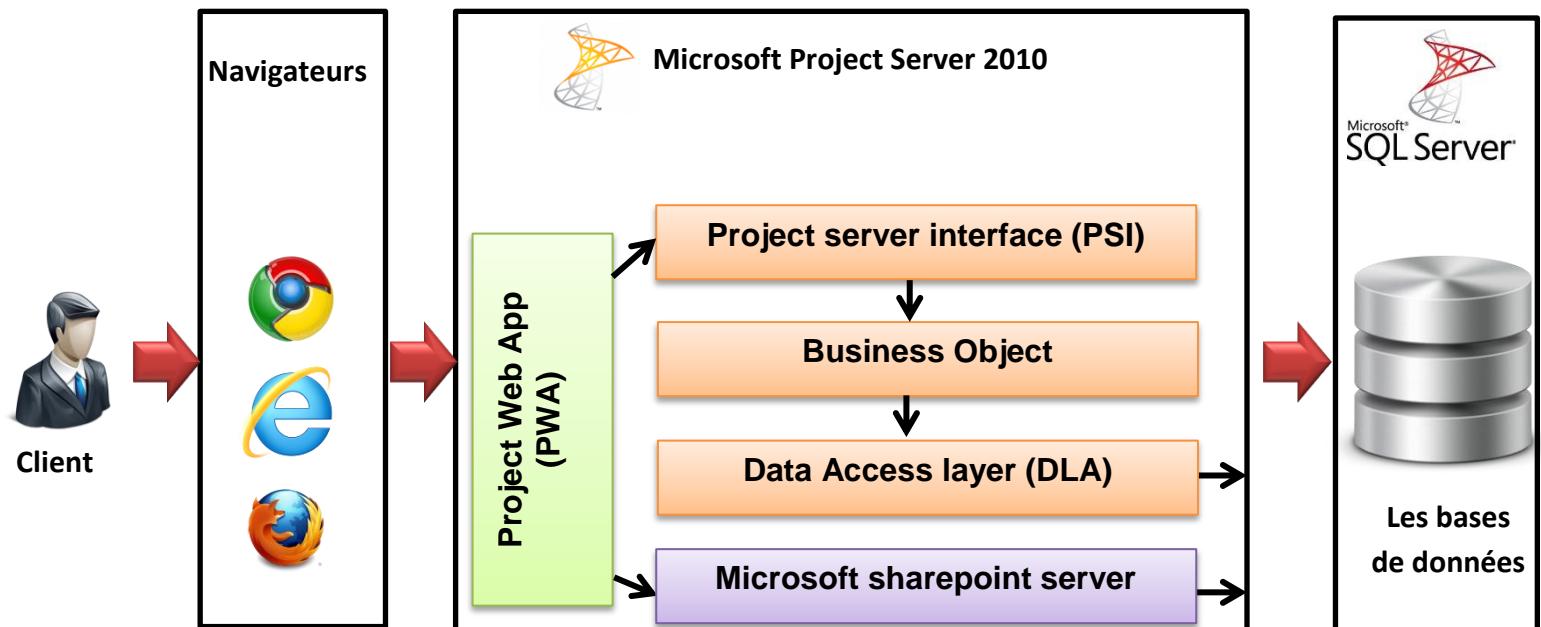


Figure 10 : Le fonctionnement de Microsoft project server 2010

## ➤ Fonctionnalités de Microsoft project server 2010:

### Planification et pilotage des projets

Microsoft Project permet la planification des projets, c'est-à-dire la création d'un plan. Il permet la création de tâches et de jalons, leur hiérarchisation, et de définir des liens entre les tâches. Une estimation de la durée et de la charge (ou travail) nécessaire à la réalisation de chaque tâche peut ensuite être réalisée.

Des modèles peuvent être créés afin de proposer des plans types aux utilisateurs (par exemple, si les projets nécessitent de suivre une méthodologie ou un processus précis).

Microsoft Project propose différentes représentations graphiques du projet : diagramme de Gantt, réseau des tâches...

Les possibilités de pilotage des projets sont nombreuses : définition de la planification initiale (c'est-à-dire l'enregistrement d'une photographie de référence du plan de projet), saisie de l'avancement des tâches, planification.

Le chef de projet peut mettre à jour l'avancement du projet :

- via la saisie d'un pourcentage d'avancement (délai ou charge) ;
- via la mise à jour de la durée réelle et de la durée restante ;
- via la mise à jour du travail réel (réalisé) et du travail restant (reste à faire).

### Gestion des ressources

Microsoft Project permet la gestion des ressources de chaque projet, c'est-à-dire la création de l'équipe projet puis l'affectation des ressources définies.

Il est possible de créer différents types de ressources :

Les ressources, en particulier celles de type travail, peuvent être typées comme génériques : elles correspondent alors à des profils de compétences.

Chaque ressource peut ensuite être affectée aux tâches du projet. Via le planificateur d'équipe, nouveauté de la version 2010, il est possible de gérer un plan de capacité des ressources et d'allouer ainsi les tâches du projet aux ressources.

Lorsque les ressources ont été affectées, Microsoft Project rend possible l'analyse des plans de charge des ressources affectées, via par exemple l'affichage d'un graphe des ressources. Une fonctionnalité, l'audit des ressources, peut être utilisée pour résoudre les problèmes de surutilisation des ressources détectés, en réorganisation des tâches du projet.

### Gestion des coûts

Chaque ressource peut avoir un coût : taux journalier pour une ressource de type travail, coût unitaire pour une ressource matérielle... Ainsi, en fonction des affectations des ressources réalisées sur les tâches du projet et de coûts fixes pouvant être définis pour chaque tâche, Microsoft Project calcule le coût du projet. Pour les individus, il est également possible de gérer les heures supplémentaires.

## Importation des Projets depuis Ms project (l'application du bureau)

L'une des fonctionnalités la plus importante de Ms project server 2010 c'est d'importer les projets depuis l'application du bureau Microsoft project , c.-à-d. créer tous les tâches ,les assignements ,les cout ...etc localement puis l'importer au Microsoft project server 2010 pour les partager avec l'équipe de travail .

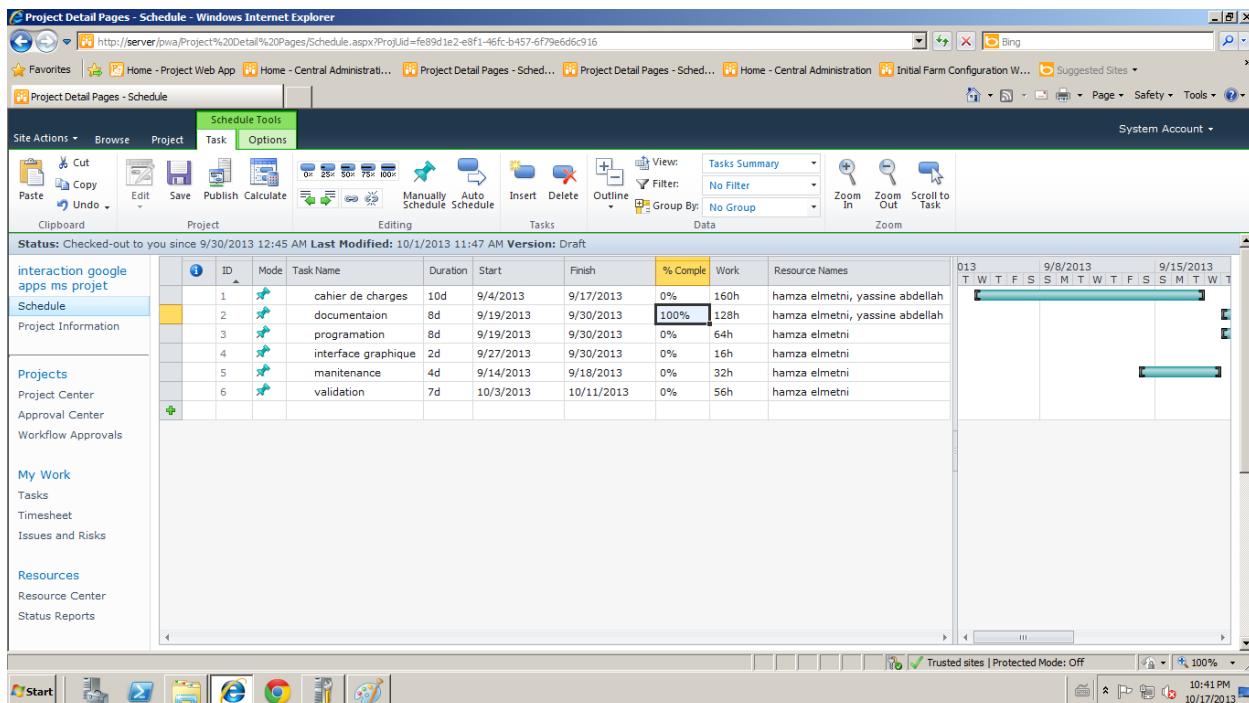


Figure 11 : une capture d'écran du Microsoft project server 2010

## C) Google apps for business



Figure 12 : Logo de Google Apps

Google Apps est un service externalisé de messagerie collaborative comprenant un serveur de messagerie, une suite bureautique et une Gestion électronique des documents de la société Google. Ce service en mode Logiciel en tant que service (SAAS) est proposé par Google à tout type d'organisation : entreprises, établissements d'enseignement, collectivités, etc.

Les principales applications en ligne incluses dans Google Apps sont Gmail, **Google Calendar**, Google Drive, Google Documents, Google Sites...etc.

## Les Principaux avantages du SAAS

- Aucun serveur à maintenir : tout est géré par Google.

- Les utilisateurs ont accès à leurs données de partout (toutes leurs données sont hébergées sur le Cloud avec un espace de stockage vaste), tant qu'une connexion internet est disponible.
- Les applications (e-mails, agendas, contacts, documents, visioconférences, etc.) sont accessibles à partir d'un navigateur internet : plus besoin de maintenir de multiples logiciels.
- Les applications se mettent à jour automatiquement : les utilisateurs bénéficient alors de toutes nouvelles fonctionnalités

## D) Google Calendar

Google Calendar est une application Google qui permet de partager des événements et des agendas et de les publier sur internet ou sur un site Web. Son utilisation nécessite d'ouvrir un compte Google.

La Fonctionnalités la plus importante de « Google calendar » liée directement à mon projet c'est **la gestion des tâches** .

Cette fonctionnalités appelle aussi « Google tasks » est inclus dans « Google calendar », c'est un service « to-do list », a l'aide de ce service on peut :

- Créer une nouvelle tâche avec une date de début, titre de tâche et remarque sur la tâche.
- Modifier et supprimer les tâches déjà existantes.
- Marquer une tâche comme achevée.
- Regrouper des tâches dans des listes.
- Les tâches s'affichent sur l'interface de « Google calendar »
- Les rappels des tâches seront envoyés par courriel, pop-up fenêtre ou bien par SMS

Voici un exemple des listes des tâches dans « Google calendar » :

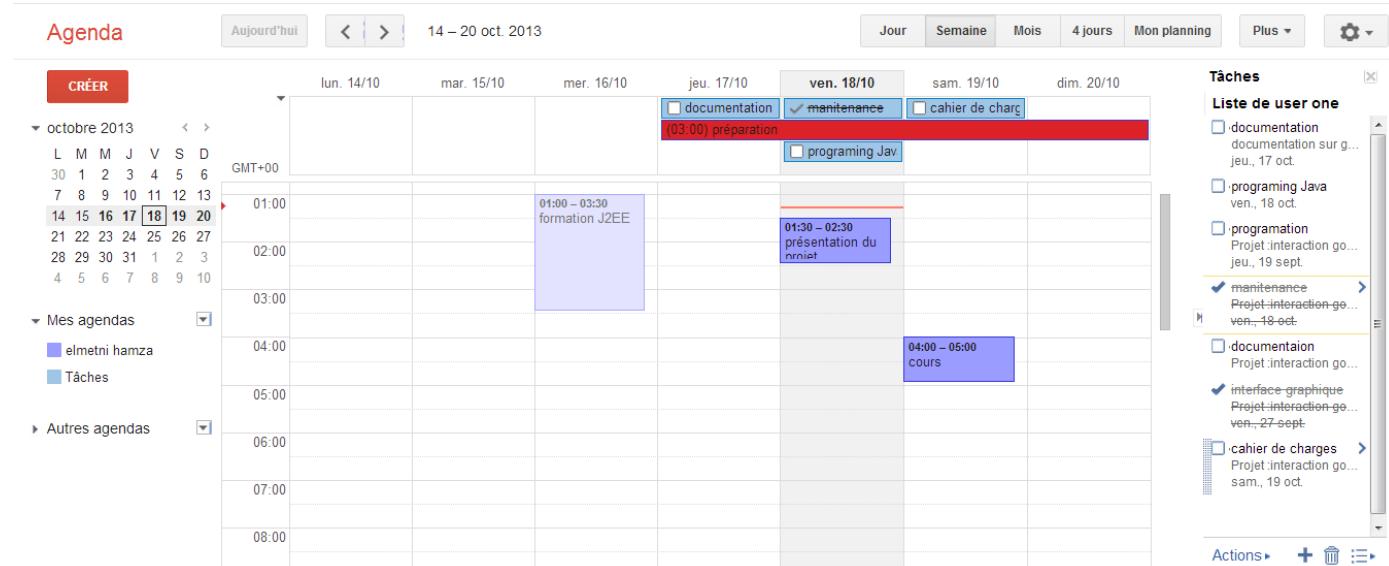


Figure 13 : une capture d'écran du Google Calendar

## E) Google api :

L'interface de programmation de Google (abr. Google API pour Google Application Programming Interface) est un ensemble normalisé de classes, des méthodes ou des fonctions qui permet aux développeurs d'utiliser les services et les application de Google .

Elle est offerte par une bibliothèque logicielle ou un service web, et elle est accompagnée d'une description qui spécifie comment des programmes consommateurs peuvent se servir des fonctionnalités du programme fournisseur.

L'objectif est de fournir une porte d'accès à une fonctionnalité en cachant les détails de la mise en œuvre.

Il exist plusieur interfaces de programmation de Google par exemple :

- Google plus Api
- Google Map Api
- Youtube Api
- Google drive Api

Dans ce projet on a utilisé « Google task Api v1 » .

## F) Google Apis Console



Figure 15 : Logo du Google Apis Console

Google Apis Console est une interface web pour la gestion et la visualisation des données de trafic, l'authentification et les informations de facturation pour l'API Google qui sont utilisés par votre projet. Dans la Console API, un projet est une collection nommée de l'information au sujet de votre demande: informations d'authentification, les adresses de courriel des membres de l'équipe de l'API Google que votre application utilise, et ainsi de suite. Vous pouvez créer vos propres projets, ou être ajoutés en tant que spectateur ou de l'éditeur de projets existants créés par d'autres développeurs. Au sommet de la console est une liste déroulante qui affiche les projets que vous avez accès, ainsi que des commandes pour la création et la gestion de projets.

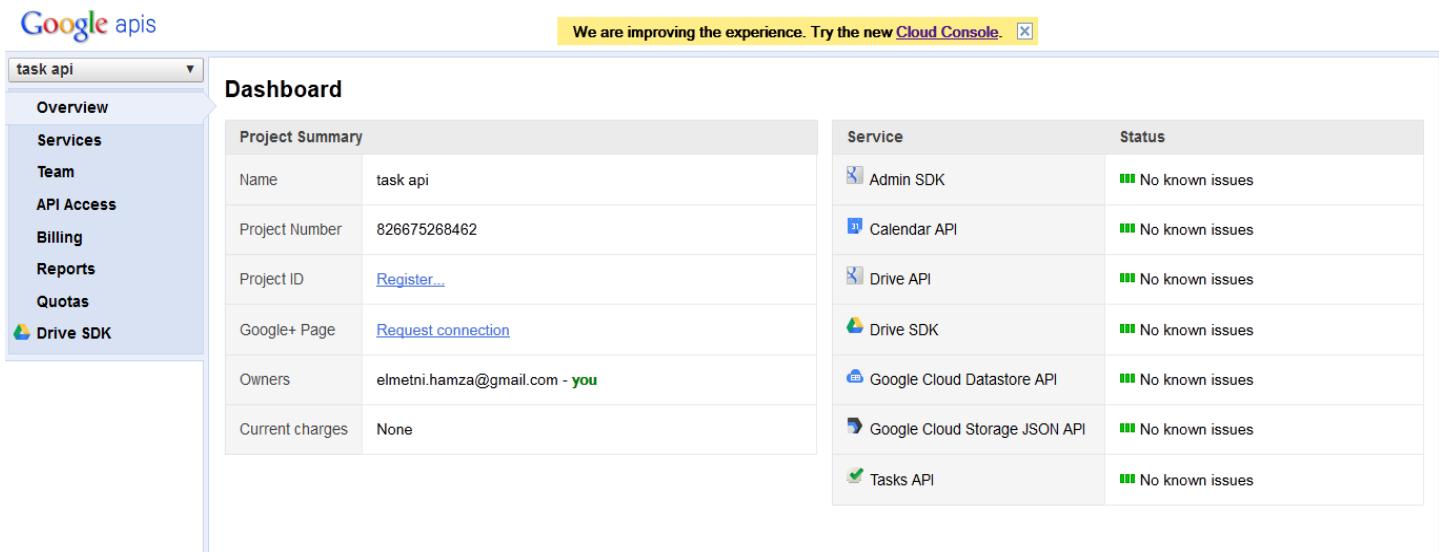


Figure 14 : page d'accueil de Google Apis Console

La page d'accueil de Google Apis Console affiche le dashboard d'un projet nommé "task api".

**Project Summary**

Name	task api
Project Number	826675268462
Project ID	<a href="#">Register...</a>
Google+ Page	<a href="#">Request connection</a>
Owners	elmetni.hamza@gmail.com - <b>you</b>
Current charges	None

**Service Status**

Service	Status
Admin SDK	No known issues
Calendar API	No known issues
Drive API	No known issues
Drive SDK	No known issues
Google Cloud Datastore API	No known issues
Google Cloud Storage JSON API	No known issues
Tasks API	No known issues

## ➤ Les fonctionnalités Google apis console

**Activated API :** Activer une ou plusieurs des API pour permettre la surveillance du trafic , de filtrage et de facturation , et les pages API spécifiques pour votre projet.

**Les informations de trafic :** La console enregistre les informations de trafic pour chaque API activée. En outre, vous pouvez limiter ou filtrer l'utilisation par les API .

**Informations de facturation :** Après avoir activé la facturation, votre API activé sont autorisés à dépasser la limite d'utilisation de courtoisie. Les frais d'utilisation sont facturés à vous en utilisant le mode de paiement que vous spécifiez.

**Clés du projet :** Chaque projet est identifié soit par une clé API ou un jeton OAuth 2.0 . Utilisez cette clé / token dans vos requêtes API pour identifier le projet , afin d'enregistrer les données d'utilisation , respecter vos restrictions de filtrage, et l'utilisation du projet de loi au projet approprié . Vous pouvez utiliser la console de générer ou de révoquer les clés de l'API ou OAuth 2.0 certificats à utiliser dans votre application.

**Membres de l'équipe :** Vous pouvez spécifier les membres supplémentaires à lire, à écrire, ou l'accès à la propriété à la page de la console de ce projet

**Gestion des rapports d'API :** l'une de plus important fonctionnalités de Google apis console, à l'aide de cette fonctionnalités on peut afficher des statistiques d'utilisation de chaque API sous la forme des Graphe:

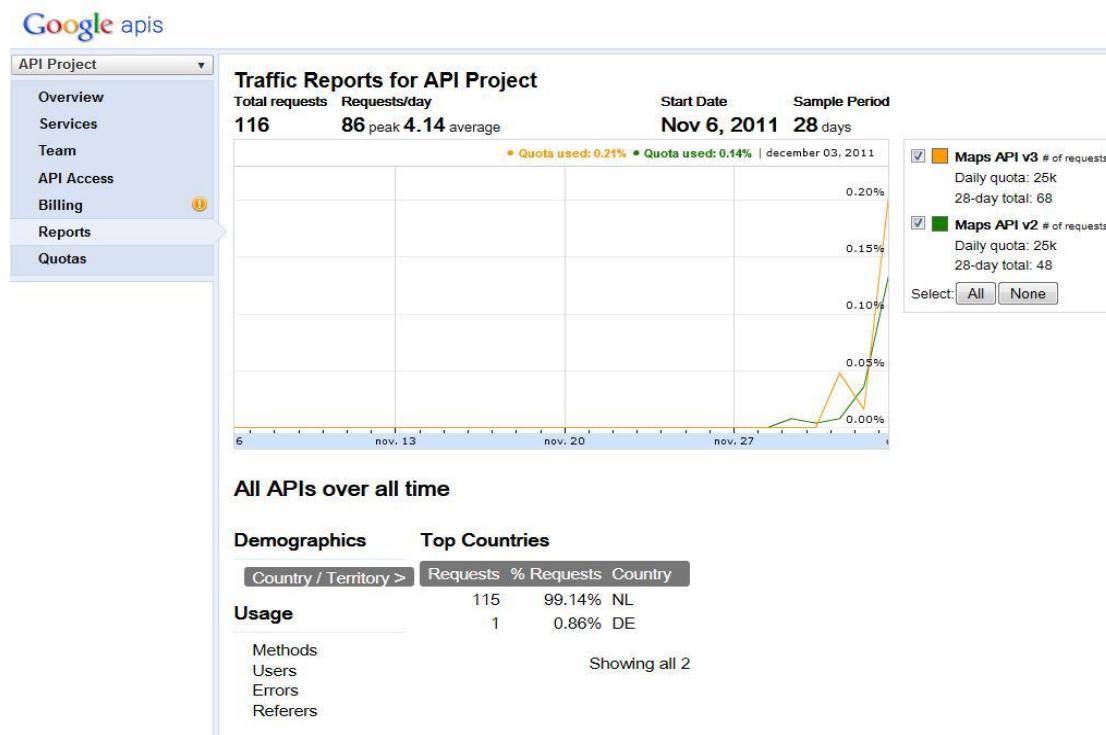


Figure 16 : Exemple d'un rapport de Google Apis console

# 4- Méthodologie du travail

## A) Cycle de vie

Dans ce projet on a utilisé cycle de vie en cascade, le principe de modèle en cascade est de découper le projet en phases distinctes sur le principe du non-retour. Lorsqu'une phase est achevée, son résultat sert de point d'entrée à la phase suivante.

Les raisons pour lesquelles j'ai utilisé ce modèle sont :

- Les besoins sont bien précis, claires et stables et tous les besoins et les objectifs sont connus avant la réalisation du projet .
- L'équipe de travail se compose seulement d'un seul membre (moi même) donc on ne peut pas parler de modèle en Y ou en V.
- la technologie utilisée est maîtrisée.
- Le modèle en cascade est adapté aux projets de durée inférieure à l'année, sur des projets à forte composante réglementaire (ça de ce projet)

La modèle en cascade ce compose des activités suivantes :

- **Spécification** : se compose de :

**Définition des objectifs**, consistant à définir la finalité du projet et son inscription dans une stratégie globale.

**Analyse des besoins et faisabilité**, c'est-à-dire l'expression, le recueil et la formalisation des besoins du demandeur (le client) et de l'ensemble des contraintes.

- **Conception générale**. Il s'agit de l'élaboration des spécifications de l'architecture générale du logiciel.
- **Conception détaillée**, consistant à définir précisément chaque sous-ensemble du logiciel.
- **Codage** (Implémentation ou programmation), soit la traduction dans un langage de programmation des fonctionnalités définies lors de phases de conception.
- **Intégration**, dont l'objectif est de s'assurer de l'interfaçage des différents éléments (modules) du logiciel. Elle fait l'objet de *tests d'intégration* consignés dans un document.
- **Mise en production**,

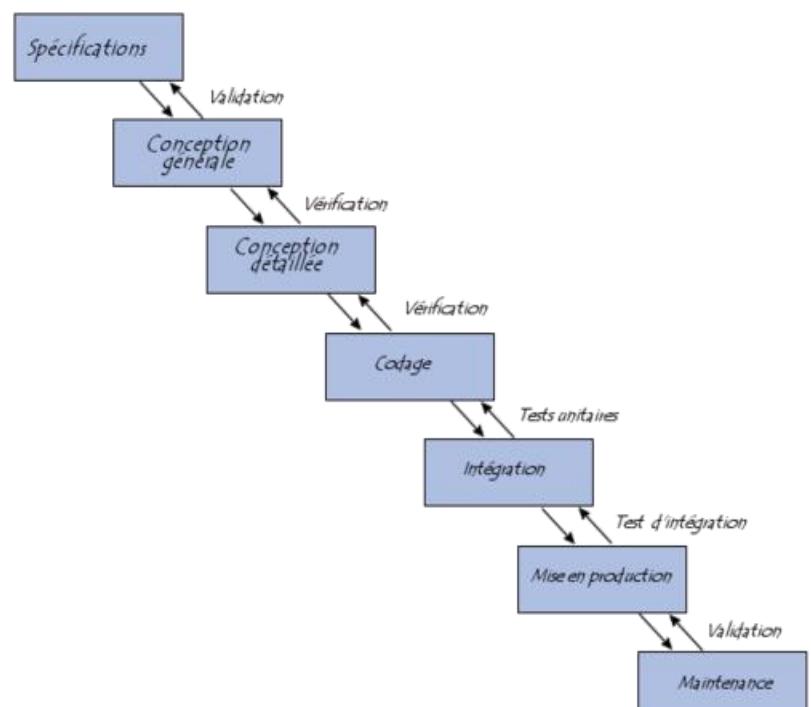


Figure 17 : Modèle en cascade

- **Maintenance**, comprenant toutes les actions correctives (maintenance corrective) et évolutives (maintenance évolutive) sur le logiciel.  
Vous trouvez plus de détails dans le chapitre suivant.

## B)Organisation de travaille au sein d'entreprise

La première semaine de stage a été réservée pour déterminer les besoins et les objectifs de stage, après la phase de planification de projet et détermination des tâches et leurs durée et l'étude de projet, on a entré dans la phase de « conception générale », pour organiser la réalisation du projet et assurer la bien suivi des tâches. J'ai organisé une réunion chaque 3 jours avec mon encadrant monsieur Abdellah yassine Ou bien avec Directeur de service information monsieur mohcine deria, pour discuter :

- L'état d'avancement du projet
- Les tâches réalisées
- Les retards
- Les problèmes
- Les tâches suivantes

Et à la fin de chaque 15 jours j'ai préparé un rapport qui contient :

- La date et le titre de rapport
- Les tâches réalisées durant toute la semaine
- Les points bloquants
- Les solutions proposées
- Etat d'avancement ou bien de retard par rapport la planification initiale du projet
- Les changements appliqués sur la planification initiale du projet
- Les tâches suivantes
- L'estimation de temps des tâches réalisées
- Si j'ai besoin d'accès à certaine donnée je les mentionne dans ce rapport

Et chaque lundi après ce rapport, je discute avec mon encadrant de stage s'il y a des commentaires ou bien des changements à propos des solutions proposées ou bien de validations des changements réalisées sur la planification initiale du projet

# CHAPITRE 2

## Description et analyse du projet

Le présent chapitre sera consacré, premièrement une description fonctionnelle du projet sous forme de la « charte du projet », où nous prêtons attention à la capture des besoins fonctionnels et le cahier de charges, que nous traduisons en spécifications, projetées ensuite en langage UML à travers les diagrammes des cas d'utilisation.

Puis je vais présenter les solutions possibles du problématique avec les avantages et les inconvénients de chaque solution, en suis une analyse et description de la solution adaptée et finalement une présentation de la planification réel du projet au cours de stage.





# 1- Charte du projet

La **charte projet** est un document délivré par l'initiateur du projet ou le commanditaire qui autorise formellement l'existence d'un projet et fournit au chef de projet le pouvoir d'utiliser les ressources organisationnelles aux activités du projet.

Elle comprend les exigences initiales qui répondent aux besoins des parties prenantes ainsi qu'à leurs attentes.

## A) Introduction de charte de projet

### ➤ Sujet de projet

Le développement d'un système pour réaliser l'interaction entre « Ms project Server 2010 » et « Google Apps for business » ( Google tasks )

### ➤ Equipe de projet :

Nom et prénom	Fonction	Fonction dans le projet
M.Yassine Abdellah	Responsable Systèmes Opérationnels	Encadrant
M.Dria Mohcin	Chef département SI	Encadrant
Elmetni Hamza	Stagiaire	Gestionnaire de projet

*Tableau 2: Equipe de projet*

## B) Description du projet

### ➤ Résumé

Ce projet entre dans le cadre de migration d'entreprise ABC de « Microsoft exchange » vers « Google apps » afin de bénéficier tous les avantages proposé par les applications de Google .

Le sujet de projet est de trouver une solution pour réaliser une interaction entre « Microsoft project Server» et « Google Apps for business « Google tasks » .

Cette solution s'agit d'une application qui va être utilisé par le System d'information d'ABC, elle va toucher tous les utilisateur de « Microsoft project Server» d'entreprise ABC , que soit des chefs de projets ou bien des membres des équipes qu'il vont réaliser certain taches .

## ➤ **Les finalités du projet**

- Mettre en évidence l'utilisation de « Google apps » pour réaliser des opérations quotidienne et se familiariser avec le fonctionnement de « Google apps »
- Avancer les procédures d'intégration entière de « Google apps » avec le System de ABC

## ➤ **Parties prenantes:**

Les utilisateurs ciblés de ce projet sont tous les utilisateurs de « Ms project server 2010 » Dans l'entreprise ABC qui ont effectué ou qui ont été chargé des taches.

## ➤ **Cahier de charges :**

Dans cette partie on va présenter le cahier de charges de projet en détails :

### **Gestion des taches :**

- **Synchronisation du la création des taches :**  
à chaque création des taches par un chef de projet en "Microsoft project server 2010 ", il faut créer ces taches dans "Google task" de membres d'équipe assignée à elles .
- **Synchronisation du la suppression des taches :**  
à chaque suppression des taches par un chef de projet "Microsoft project server 2010 ", il faut supprimer ces taches de "Google task" de membres d'équipe assignée à elles .
- **Synchronisation du la modification des taches :**  
à chaque mise à jours des taches par un chef de projet "Microsoft server project 2010 ", il faut mettre à jours ces taches dans "Google task" de membres d'équipe assignée à elles .
- **Synchronisation du la complétude des taches :**  
à chaque fois un utilisateur de "Microsoft project server 2010 " marquer une tache comme « complet » , il faut que cette tache soit marquer comme complète en « Google tasks» , et à chaque fois un utilisateur de " Google tasks" marquer une tache comme « complet » , il faut que cette tache soit marquer comme complète en "Microsoft project server 2010 " .
- **Les informations des taches :**  
Chaque tache créer par notre solution doit contenir les informations suivantes :
  2. Nom de la tâche.
  3. Nom du projet concerné
  4. la date de début de la tâche
  5. la date de fin de la tâche
  6. le nom et l'email de la personne assignée à la tâche

7. le nom et l'email du chef de projet concerné
8. la description de la tâche
9. la date d'envoi d'alerte.

#### **Gestion des messages des alertes :**

- condition d'envoie des alertes :

Une alerte doit être envoyée au chef du projet si une tâche n'a pas été validée après un délai déterminé de sa date de fin.  
Ce délai défini par l'administrateur d'application.

- Les informations des alertes :

Chaque alerte envoyée par notre solution doit contenir les informations suivantes :

10. Nom de la tâche.
11. Nom du projet concerné
12. la date de début de la tâche
13. la date de fin de la tâche
14. le nom et l'email de la personne assignée à la tâche
15. la description de la tâche
16. la date d'envoi d'alerte.
17. Le cadre d'envoie le message d'alerte

#### **Gestion des assignments :**

Une tâche peut être assignée à une ou plusieurs personnes

- L'ajout d'un assignment :  
à chaque ajout d'un assignment d'une tâche en « Microsoft project 2010 », il faut créer cette tâche en « Google task » des utilisateurs assignés à elle .
- La suppression d'un assignment :  
à chaque suppression d'un assignment d'une tâche en « Microsoft project 2010 », il faut suppression cette tâche de « Google task » des utilisateurs assignés à elle .
- La validation d'une tâche :  
si une tâche a assigné au plusieurs personnes , seulement la validation de tous les personnes assignées à cette tâche peut la valider en « Microsoft project server 2010 »

#### **Historisation :**

il faut enregistrer tous les opérations effectuées par la solution

#### **Sécurité :**

il faut sécuriser des données d'accès au cas d'utilisation d'un fichier externe pour les stocker .

#### **Interface graphique :**

ou cas du développement d'une solution avec une interface graphique , il faut respecter les Règles ergonomiques ( interface homme machine )

## ➤ La portée du projet

Pour réaliser les objectifs de ce projet on va créer un système (ensemble des applications) qui a les rôles de :

- Communiquer avec les servers de Google pour gérer « Google task et calendar » .
- modifier des valeurs dans la base de données de « Ms project »
- synchroniser la modification dans la base de donnée de Ms project avec la gestion des « Google Task et Calendar » dans les server de Google .
- recevoir des données à partir des serveurs de Google et appliquer des modifications sur les bases de données de « Ms project »

## ➤ Les Milestones

Identifier les points ou les événements importants du projet (tels que les phases, étapes, points de décision et l'approbation d'un livrable) :

Milestone de projet	Description
1. création de charte de projet et la planification.	Etudier des concepts de management de projets qu'on va utiliser. Réaliser une étude globale de projet Etudier la faisabilité de projet. Discuter les détails de projet avec l'encadrant de stage.
2. la recherche de solution et documentation.	Etudier la faisabilité de projet et les solutions possible et la technologie qu'on va utiliser dans le projet
3. Réalisation de la solution avec une base de données local et des comptes de Google gratuits.	Développer les ensembles des applications et les tester avec des bases de données locales et de comptes de Google gratuit.
4. intégrer la solution avec le System d'ABC	Intégrer la solution en utilisant des compte de « Google apps for business » et des bases de données de « Ms project server »

*Tableau 3: Milestone de projet*

## ➤ Les livrables

Un livrable désigne tout composant matérialisant le résultat d'une prestation de réalisation

<b>livrable 1:</b>	<b>Charte de projet</b>
Description:	C'est un document qui donne une description de tous les détails de projet
Critères d'acceptation	Il faut qu'il donne une claire idée sur le projet, et ne laisse pas d'ambiguïté
Date de livraison	7 juillet 203
<b>livrable 2:</b>	<b>Etat d'avancement</b>
Description:	C'est un document qui donne une idée sur l'état d'avancement de projet
Critères d'acceptation	Il faut qu'il contienne : <ol style="list-style-type: none"> <li>1. Toutes les tâches réalisées.</li> <li>2. Les problèmes et points bloquants</li> <li>3. Les tâches suivantes</li> </ol>
Date de livraison	Chaque semaine.
<b>livrable 3:</b>	<b>La solution :</b>
Description:	
Critères d'acceptation	Il faut qu'elle respecte le cahier de charges
Date de livraison	15 septembre 2013
<b>livrable 4:</b>	<b>Rapport de projet</b>
Description:	le rapport de l'ensemble du projet du début jusqu'à fin
Critères d'acceptation	Il faut qu'il accepte les normes d'un rapport de stage
Date de livraison	La fin de durée de stage
<b>livrable 5:</b>	<b>Guide d'utilisation</b>
Description:	Ce document aide à utiliser et configurer la solution .
Critères d'acceptation	Contient une description détaillée sur : <ul style="list-style-type: none"> <li>- La configuration de la solution</li> <li>- Tous les éléments d'interface graphique</li> </ul>
Date de livraison	La fin de durée de stage

Tableau 4:les livrables de projet

## ➤ Les risques

Cette évaluation initiale des risques ne remplace pas l'évaluation complète des risques réalisée au cours de la phase de planification et documenté dans le projet de Plan-les deux sont des activités qui viennent après la charte de projet est établie.

No .	Description du risque	Probabilité	incidence	mesures d'atténuation prévues
1.	Risque de retard : au niveau technique on ne connaît pas vraiment des solutions et ça peut causer des retards de projets	Haute	faible	On peut accélérer dans les phases quotidiennes.
2.	Risque de n'avoir pas une solution pour ajouter des taches dans « Google task »	moyenne	Haute	C'est un risque bloquant de projet
3.	Risque de n'avoir pas le niveau technique approprié pour réaliser certain taches	Faible	Moyenne	On peut réserver plus de temps pour la recherche et on peut demander d'aide des encadrants
4.	Risque de ne pas avoir l'accès au compte administratif de « Google business »	faible	haute	on peut demander ca au chef de département.
5.	Risque de ne pas avoir l'accès à la base de données de Ms project (au niveau technique)	moyenne	Haute	C'est un risque bloquant de projet
6.	Risque de bloquer le système en modifiant quelque donnée de base de données de MS Project	Trop faible	haute	On peut demander de surveillance au hors de modification de donnée

Tableau 5:les risques de projet

## ➤ Les contraintes

Identifier les contraintes et restrictions que les conditions aux limites ou lieu spécifiques sur le projet, en particulier ceux qui sont associés au projet

No.	catégorie	contrainte
1)	Expérience	Le Gestionnaire de projet n'a pas d'expérience dans le management de projet
2)	Technique	Il n'existe pas des « Google task samples » officiel pour Java
3)	Technique	On peut avoir des problèmes d'autorisation des modifications au niveau de « Google apps »

No.	catégorie	contrainte
4)	Technique	<i>Le Gestionnaire de projet n'est pas familier avec PL/SQL pour créer des triggers</i>
5)	Temps	On peut changer des délais car au niveau technique on ne connaît pas vraiment des solutions
6)	légal	On n'a pas d'autorisation au compte administratif de « Google apps business »
7)	légal	On ne peut pas accès toujours à la base de données de Ms projet à la raison de l'autorisation.
8)	Temps	Ce projet peut prendre plus de temps que la durée de stage de le Gestionnaire de projet

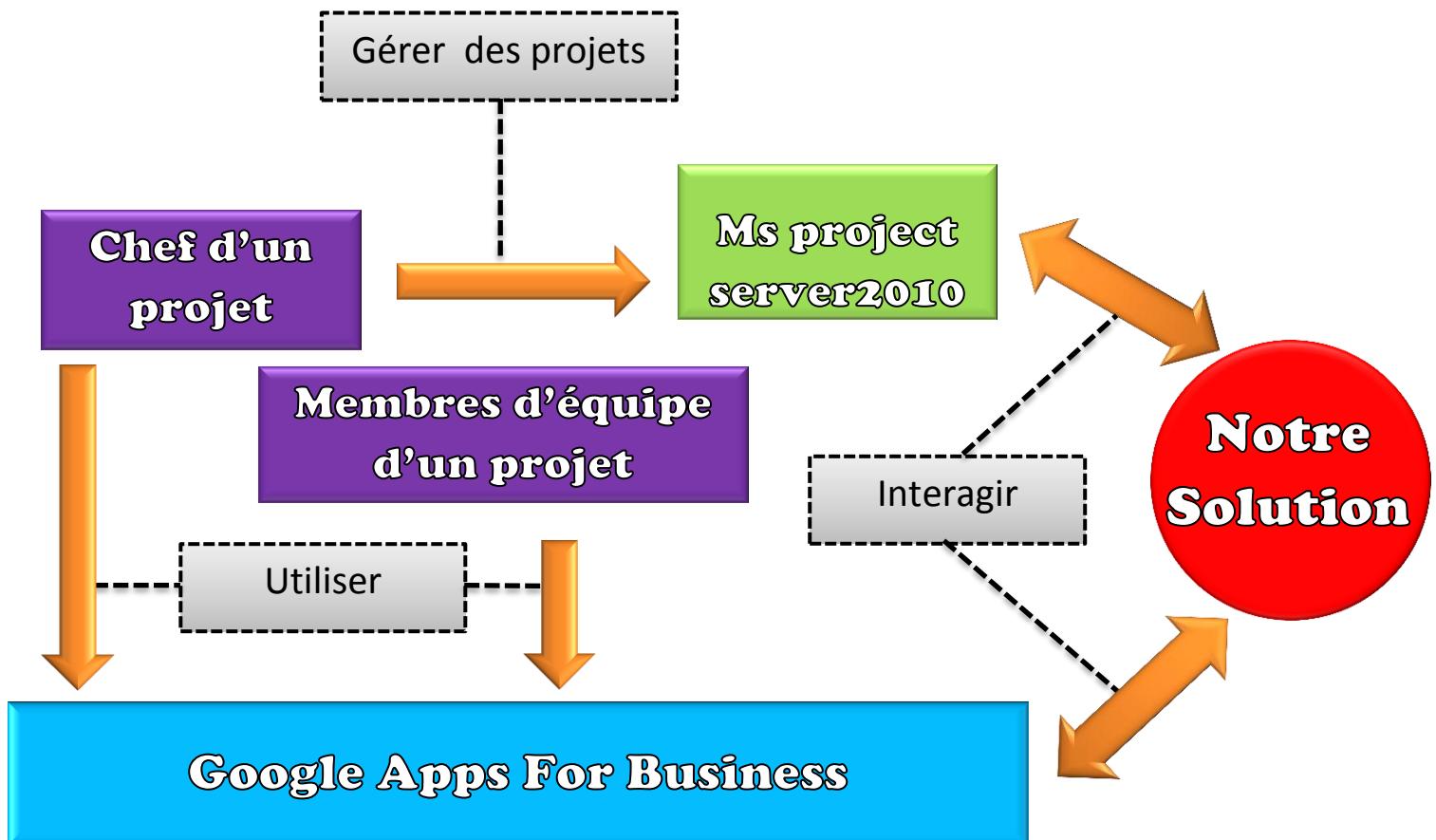
*Tableau 6:les contraintes de projet*

## C)Organisation du projet

### ➤ L'utilité de projet

Le schéma suivant représente ou on va appliquer notre solution :

Le mot « projet » dans le schéma représente n'importe quel projet d'ABC .



*Figure 18: un schéma représente L'utilité de projet*

## ➤ Structure de l'équipe de projet

Le schéma suivant représente la Structure de l'équipe de projet :

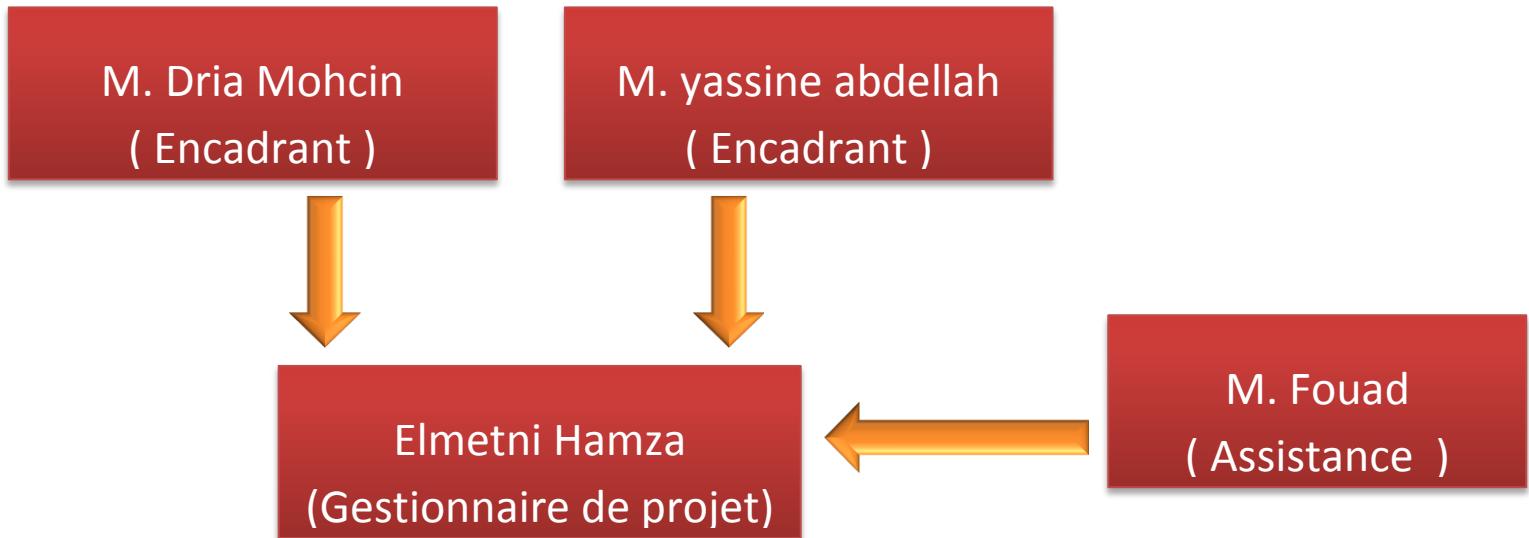


Figure 19:Structure de l'équipe de projet

## ➤ Rôles et responsabilités

Dans cette partie on va définir les rôles et les responsabilités assignées à chaque membre de l'équipe de projet ainsi que ceux de tous les intervenants et les groupes de travail qui ont une influence significative sur le projet

Rôle de projet	responsabilités	assigné à
Encadrant	Encadrer , surveiller , donner des indications , proposer des amélioration , fournir des documentation , suivre la réalisation des taches	M. yassine abdellah M. Dria mohcin
Assistance	Donner d'assistance avec « Google apps business » en donnant l'accès a des comptes administratif	M. fouad
Gestionnaire de projet	Réaliser les taches, développer des solutions, gérer de projet , chercher des améliorations , développer les application .	Elmetni hamza

Tableau 7:Rôles et responsabilités d'équipe de projet

## ➤ **Les installations et les ressources de projet**

On va décrire, les exigences du projet pour les installations et ressources, tels que des bureaux, des installations spéciales, le matériel informatique, matériel de bureau et des outils de soutien, ainsi qu'identifier la personne ou l'équipe responsable de l'obtention des éléments précis permettant de soutenir l'environnement de développement du projet.

<b>La ressource</b>	<b>L'utilisation</b>	<b>assigné à</b>
Exemple d'un projet avec Ms projet	Pour tester les bases données et l'application	M. yassine Abdellah
L'Accès a un compte administratif de « Google apps business »	Pour effectuer des configurations	M. fouad
Des comptes de « Google apps business »	Pour tester l'application avec les comptes de Google d'ABC	M. fouad
Avoir de documentations sur : <ul style="list-style-type: none"><li>• La production dans ABC</li><li>• Le system d'information d'ABC</li><li>• Des Documentation générale sur l'entreprise ABC</li></ul>	Pour rédiger le rapport de stage	M. Dria mohcin

*Tableau 8:Les installations et les ressources de projet*



## **2- Les cas d'utilisation**

Les cas d'utilisation permettent de représenter le fonctionnement du système vis-à-vis de l'utilisateur : c'est donc une vue du système dans son environnement extérieur.

Les fonctionnalités principales, que le système doit assurer, ainsi raffinées, on passe à leur modélisation en langage UML à travers du diagramme des cas d'utilisation.Dans cette partie on va prendre le système : Ms project server 2010 , Google apps , notre solution et Gmail pour étudier les cas d'utilisation de notre projet car on ne peut pas séparer ces éléments de projet , on va appeler ces cas d'utilisation Vue global , vous trouvez dans la page 49 les cas d'utilisation de vue solution.

Le schéma suivant présente le diagramme des cas d'utilisation de notre projet (vue global) :

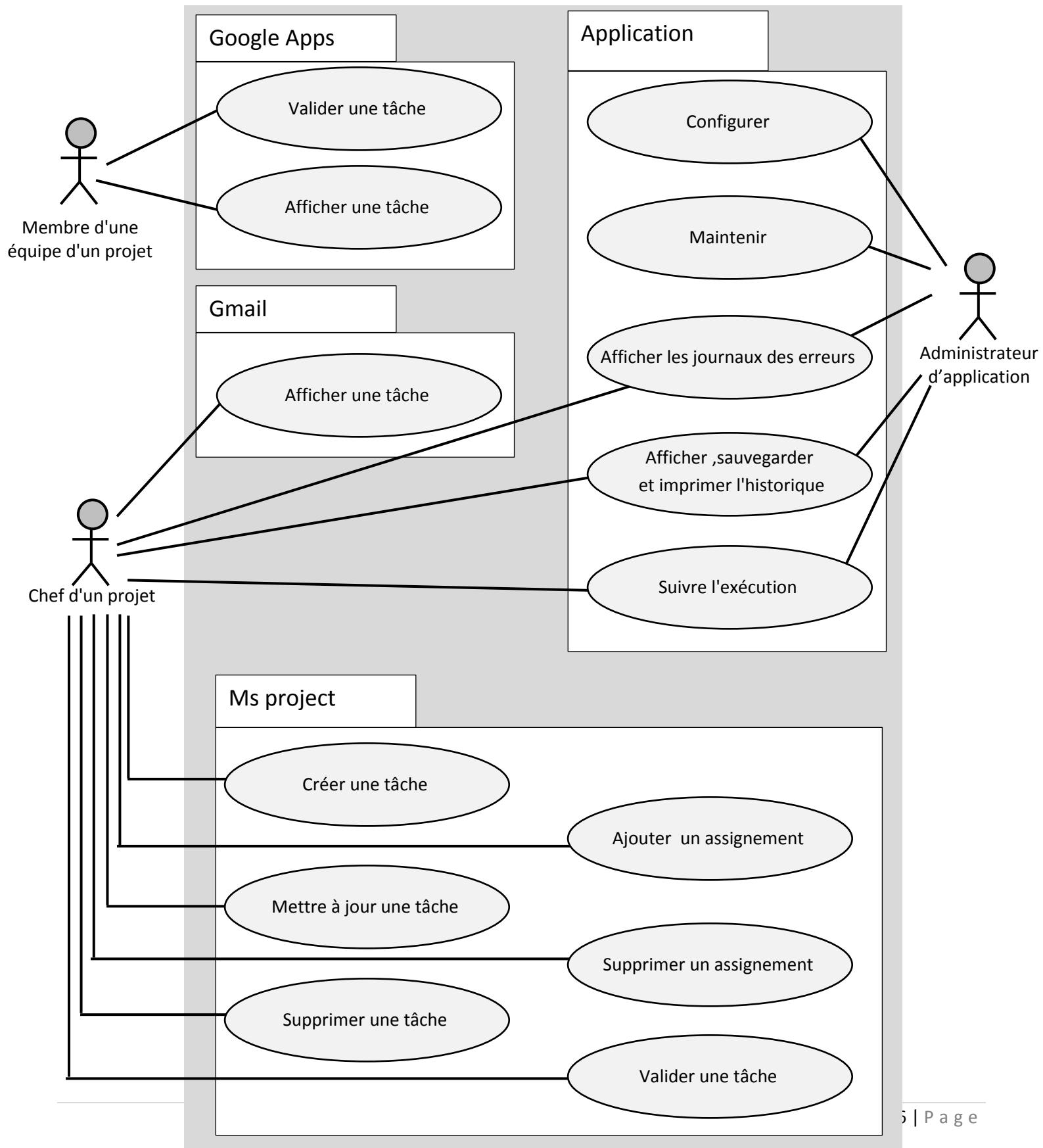


Figure 20 : le diagramme des cas d'utilisation de projet(vue global)

### 3- Analyse des solutions

Dans cette section on va analyser tous les solutions possibles de notre projet avant de parler de la solution adaptée.

On va réalisation une analyse des possibilités puis une évaluation de chaque solution , puis on va finir par une conclusion .

#### A) La première solution :

L'idée générale de cette solution est d'ajouter un déclencheur (Trigger) qui va exécuter notre application (à partir des commandes par exemple) chaque fois que l'utilisateur effectue un changement dans la base de données de « Microsoft Project Server 2010 », cette application va interagir avec Google apps pour effectuer les opérations nécessaires (Google Tasks pour les tâches et Gmail pour les alertes) :

Le schéma suivant présente la forme de cette solution :

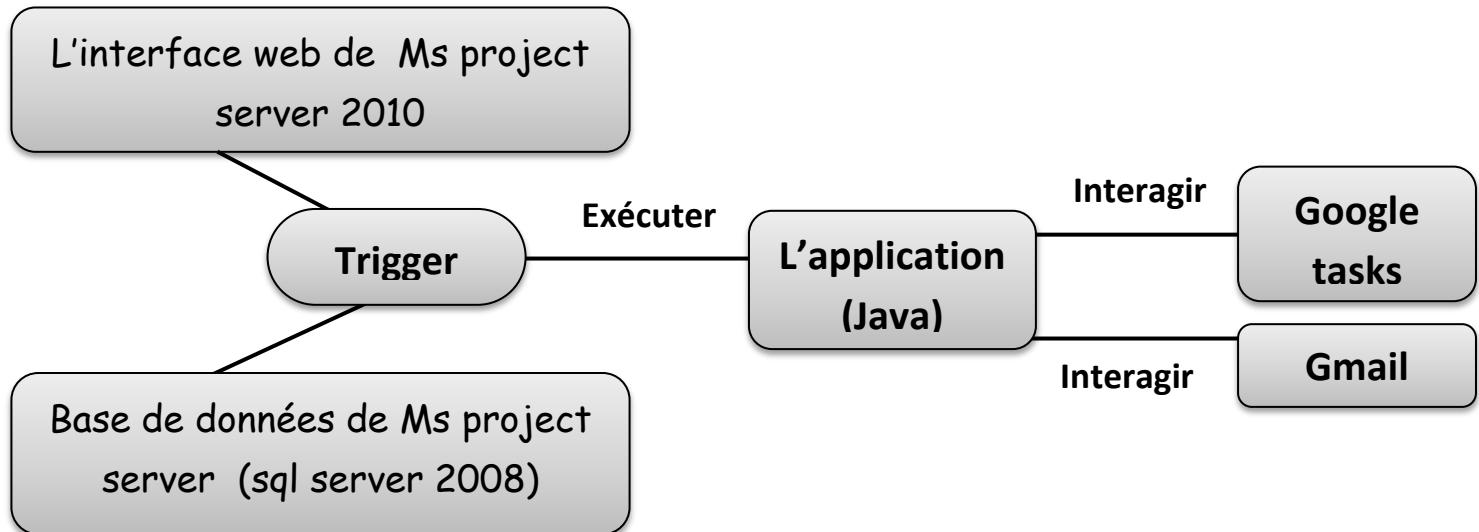


Figure 21: La forme de première solution possible

#### ➤ Avantage :

- Les notifications de « Google tasks » se font en temps réel après la création des tâches en « Ms project »
- On n'a pas besoin d'une base donnée pour l'application java (base donnée local)
- L'exécution est très rapide

#### ➤ Inconvénient :

- On peut avoir des problèmes si il existe plusieurs utilisateurs qui font des changements en même temps (a vérifié)
- C'est un peu lourd pour le serveur et risqué.

- Elle est plus compliquée au niveau technique et demande plus des ressources au cours d'exécution

## B) La deuxième solution :

Dans cette solution on ne va pas utiliser les déclencheurs on va laisser tous le travail pour une application java , l'application va lire tous les données de base de données de « Ms project server » et réaliser les opérations nécessaires , dans cette solution on va utiliser design pattern « Observateur » pour détecter s'il y a des changements au niveau de base de données

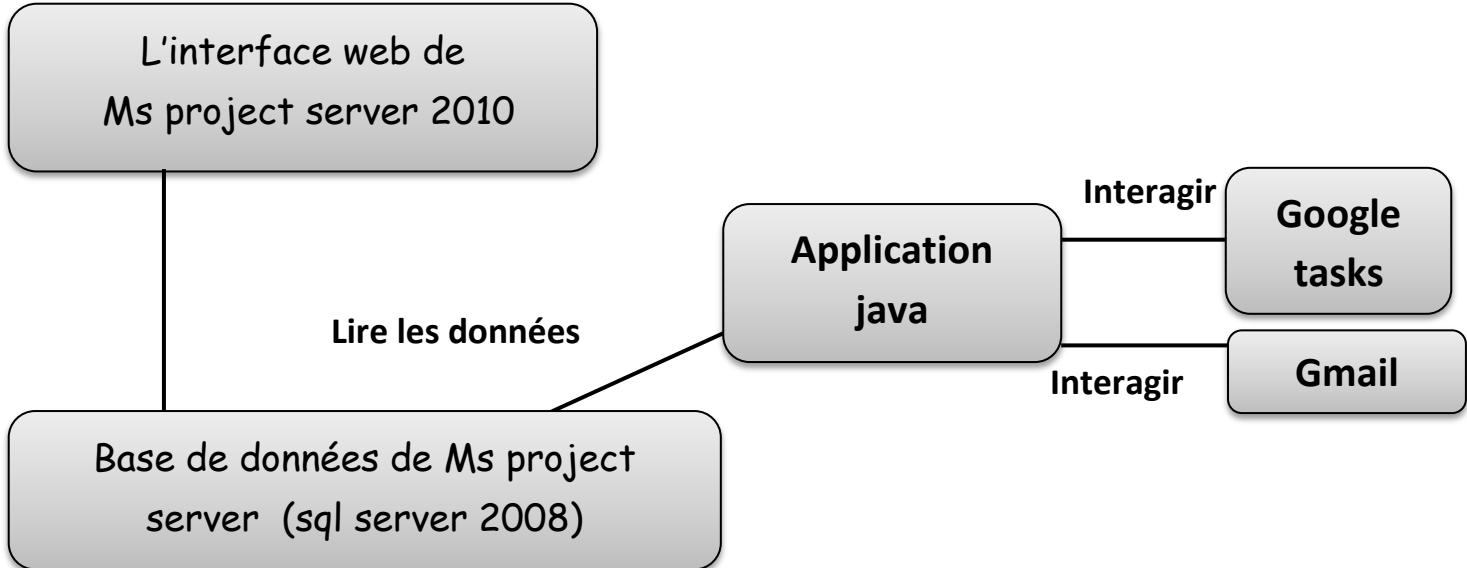


Figure 22: La forme de la deuxième solution possible

### ➤ Les avantages :

- Il est facile au niveau de la réalisation technique.
- On n'a pas besoin d'une base donnée pour l'application java (base de données locale)

### ➤ Les inconvénients :

- Il est très lent car chaque fois il faut lire tous les données de « Ms project » .
- On peut avoir de redondance des notifications de « Google tasks » au cas d'un redémarrage du système, car on ne sait pas les utilisateurs qu'on a déjà notifié.
- Il est lourd pour le serveur.
- les notifications ne se font pas en temps réel (dépend à la période de temps )

### C) La troisième solution :

A chaque fois on ajoute, modifie, supprime quelque chose de base de données de « Ms project » un trigger va insérer ces données dans la base de données locale de la solution .

L'application va s'exécuter une fois chaque période de temps pour lire les données de sa base de données et interagir avec Google Apps .

Une fois l'application Java termine la lecture des données il vide la base pour éviter la redondance des notifications.

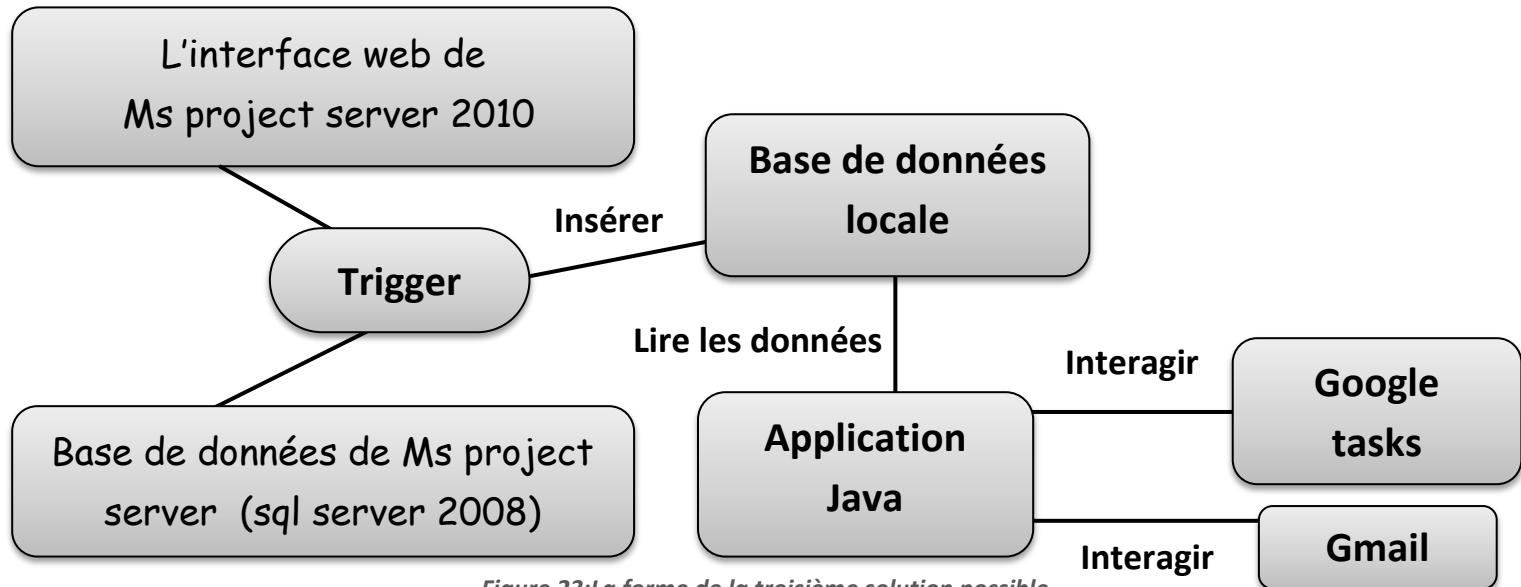


Figure 23:La forme de la troisième solution possible

#### ➤ Avantage :

- Elle est facile au niveau technique par rapport au première solution .
- Elle est rapide car il ne demande pas des traitements lourds pour le serveur et on n'est pas obligé de lire tous la base de donnée de ms project server
- On évite la redondance des notifications.
- Facile à maintenir

#### ➤ Inconvénient :

- les notifications ne se font pas en temps réel (dépend a la période de réexécution d'application java )
- on a besoin d'une base de données locale (plus de ressources )
- On peut avoir des problèmes s'il existe plusieurs utilisateur qui font des changements ont même temps .

## D) Conclusion :

En conclusion, nous pouvons dire que chacune de ces trois solution a ses avantages et ses inconvénients, mais la troisième solution est la meilleure parmi eux même si elle demande une base de donnée locale (c.-à-d. plus de ressource) par contre elle facile à intégrer et elle ne touche pas la base de données de Microsoft project server, ainsi elle est très rapide car on parle seulement de traitement des opérations qu'elles ont effectuées dans une intervalle de temps (période de réexécution).

La seule problème risquée avec ce genre de solution c'est l'utilisation des déclencheurs (triger), les déclencheurs ne pose pas seulement des problèmes avec le traitement de plusieurs opération au même temps mais aussi la complexité de les développer si on prend en considération que la base de donnée de Microsoft project server est composé de plusieurs tables et plusieurs base de donnée (plus de 200 tables divisées sur 7 ou 8 base de donnée).

Pas seulement ça, après une recherche concernant la mécanisme de traitement de données de Microsoft projet server, j'ai trouvé lorsqu'on import un projet à partir de « Microsoft projet application de bureau » au Microsoft project server, ce dernier importe les données en utilisant « importation en Masse » c.-à-d. l'exécution de plusieurs requêtes SQL en même temps (au moins 100 requêtes), et si on utilise les déclencheurs dans ce cas l'importation va prendre plus de temps car chaque requête exécutée va faire l'appelle au déclencheur, et « importation en Masse » est lié à une condition de « time-out » ça veut dire si l'importation ne se fait pas dans durée déterminé toute la procédure sera annulée.

Finalement on peut dire que l'utilisation de déclencheurs va arrêter l'importation des projets à partir de « Microsoft projet -application de bureau- », et l'arrêt de cette opération est une erreur critique si on prend que plus de 70 % des projets réalisés en « ABC tanger » utiliser cette importation.

Sur la base de tout cela, on est obligé de chercher une nouvelle solution qui va prendre les principaux avantages de ces trois solutions et éviter leurs inconvénients (spécialement l'utilisation des déclencheurs).

La section suivant présente une étude générale de la forme de cette solution.



## 4- Analyse de la solution

Dans cette section on va réaliser une description et étude générale de la solution adaptée, on va laisser des détails techniques pour le chapitre 3.

### A) La forme générale de la solution

Dans cette solution on a remplacé les déclencheurs par des « curseurs » qui lisent les données à partir de les bases de donnée de « Ms project server » et les comparer avec la base donnée locale de la solution.

Une application Java lit les donnée stockés dans la base de donnée locale et effectue les opérations nécessaires sans toucher les bases de données de « Ms project server » (**sauf dans une seule cas on va la détailler en chapitre 3**).

Comme on voit ici cette solution est une combinaison de solutions qu'on les a détaillées dans la section précédente.

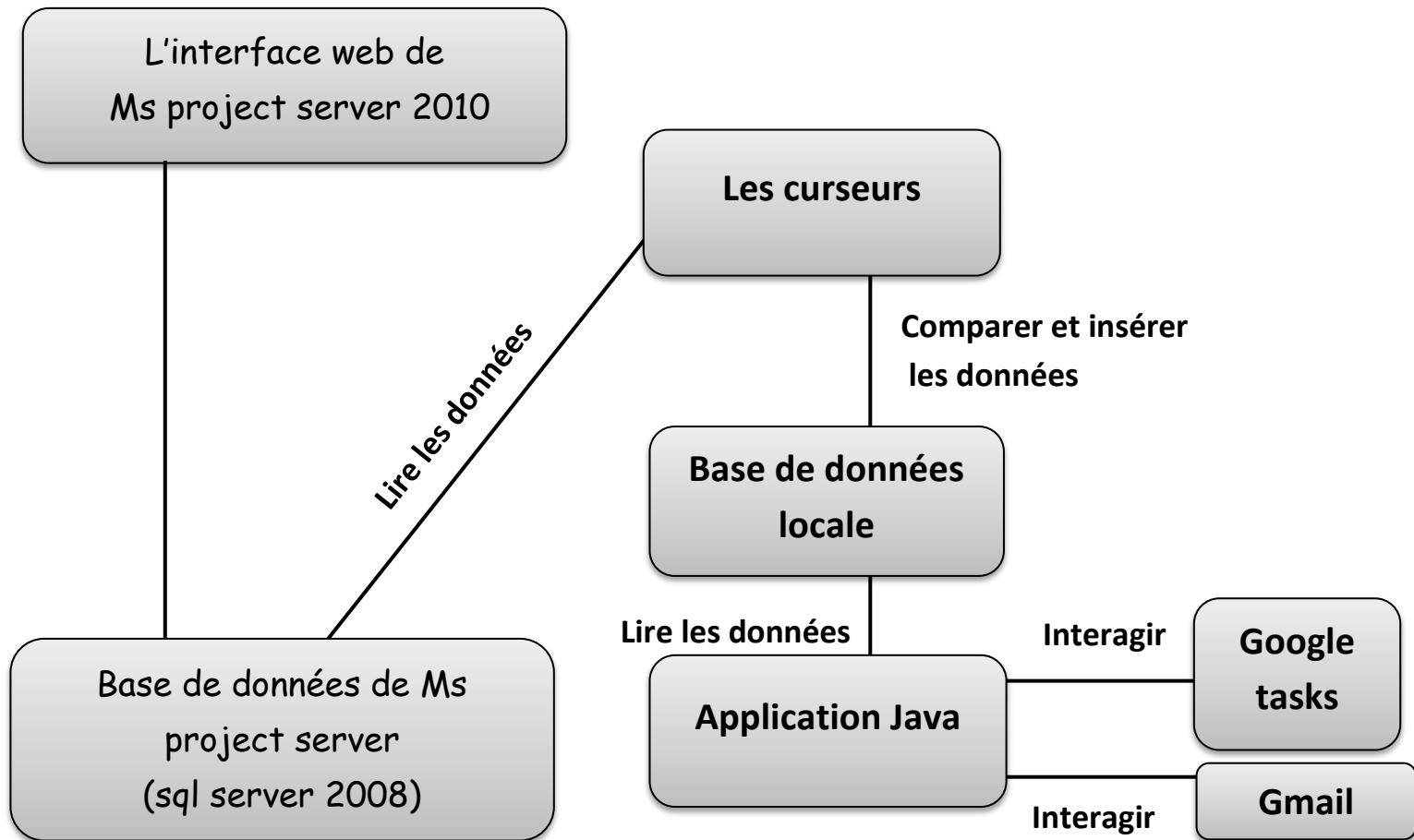


Figure 24: La forme générale de la solution

## B)Les avantages

Les avantages de cette solution sont :

- Elle ne cause pas des problèmes à l'importation du projet à partir d'application du bureau
- Elle est rapide car il ne demande pas des traitements lourds pour le serveur et on n'est pas obligé de lire tous la base de donnée de ms project server
- On évite la redondance des notifications.
- Elle n'est pas compliquée à réaliser au niveau technique en le comparant avec la première solution
- Facile à maintenir : L'utilisation des curseurs donne à cette solution plus de flexibilité car on prend une partie de programmation de Java vers PL/SQL, et le changement dans un code PL/SQL est assez facile que Java .
- L'utilisation des curseurs donne aussi une rapidité à l'exécution de cette solution car ils sont assez rapide que l'utilisation des déclencheurs.

## C)Les technologies utilisées

Le tableau suivant présente les technologies utilisées dans notre projet

La technologie	L'intérêt d'utilisation
Java	On a utilisé Java comme le langage de programmation métier principale, on l'a utilisé pour développer l'application qui interagit avec Google apps et avec base donnée locale
transact sql	Pour développer les curseurs
Google Api tasks V1	Pour interagir avec Google tasks
XML	Pour créer la fichier de configuration ( à voir chapitre 3)
SQL server 2008	Comme un système de gestion des bases de données
JavaMail – GMail via TLS	Pour communiquer avec SMTP de gmail pour envoyer des alertes( à voir chapitre 3)
Microsoft project server 2010	

Tableau 9 : Les technologies utilisées dans le projet

# 5- La planification du projet

La figure suivante présente la planification **réelle** de projet (réalisé avec Microsoft project) les mots sont en anglais car j'utilise la version anglais de Microsoft project.

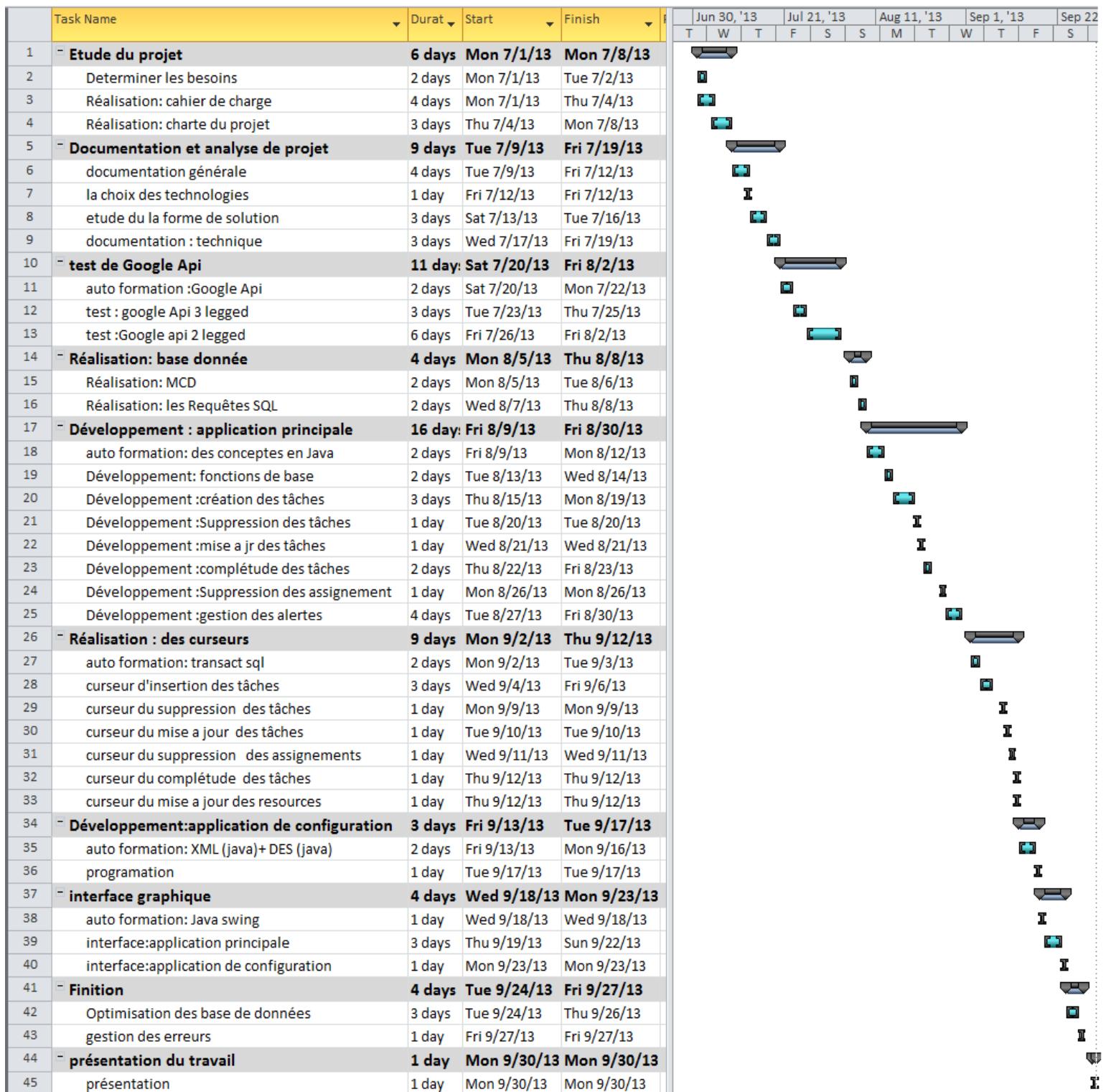


Figure 25 : la planification réelle de projet



# CHAPITRE 3

## Mise en oeuvre du projet

Une fois les besoins sont définis et analysés, la phase de conception détaillée et la réalisation est l'étape suivante dans le processus de développement en cascade. Cette phase a pour rôle de réaliser ce qui a été analysé en le concevant tout d'abord, le mettant en œuvre par la suite et le déployant vers la fin. Ainsi, la première partie est réservée à la structure du système en appliquant quelques notions d'architecture logiciel. Puis du structure de system on va sauter vers la structure de base de données .

En suit on va parler des fonctionnalités des applications Java , puis une description de côté sécurité de la solution , encore on va présenter les interfaces graphiques qui existent finalement on va parler des techniques d'optimisations de notre base de données



# 1- L'architecture de la solution

Dans cette partie on va parler de l'architecture de notre solution, en appliquant les diagrammes d'architecture logicielle.

L'architecture logicielle décrit d'une manière symbolique et schématique les différents éléments d'un ou de plusieurs systèmes informatiques, leurs interrelations et leurs interactions, l'architecture décrit le « comment le faire » c'est pourquoi on a laissé l'architecture pour le chapitre 3 , on va la présenter selon trois vues : vue contexte ,vue cas d'utilisation , vue logique et finalement vue physique.

## A)Vue contexte :

La vue contexte est représentée par un diagramme de contexte.

Le **diagramme de contexte** d'un système organisationnel est l'ensemble des objets échangés (produits, services et/ou informations) entre un système organisationnel (la boîte noire) et des acteurs de l'environnement externe du système organisationnel à l'étude.

Le but du modèle de contexte est d'identifier les acteurs et les objets échangés avec le système à l'étude.

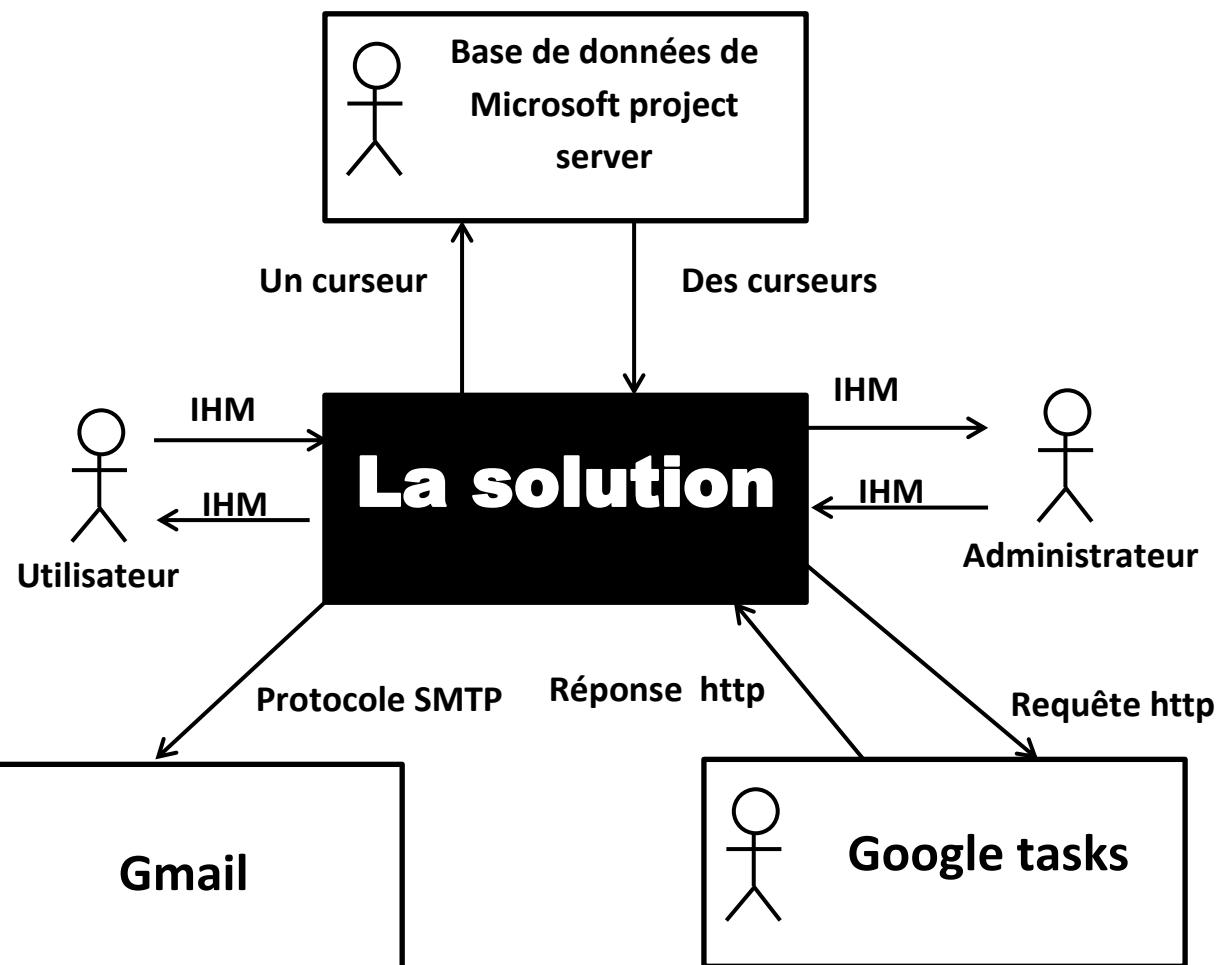


Figure 26 : Le diagramme de contexte de la solution

Notre solution interagit avec la base de données de Microsoft projet server à partir des curseurs en 2 sens, on va détailler tous les fonctionnalités des curseurs plus tard dans ce chapitre.

Interaction avec Gmail se fait avec protocole SMTP , pour Google Tasks on utilise des Requête de type http pour envoyer des requête et recevoir des ressources.

Pour voir la forme générale de la solution vous veuillez revenir au page 41

L'administrateur d'application et l'utilisateur interagissent avec la solution à partir d'une interface graphique

## B)Vue cas d'utilisation :

Un cas d'utilisation est défini comme un ensemble de scénarios d'utilisation, au contraire de cas d'utilisation cité dans la page 36, on va parler dans cette partie de cas d'utilisation de vue solution .

Les acteurs dans cette vue sont les acteurs cité dans la vue contexte

On va présenter cette vue en utilisant un diagramme de cas d'utilisation (UML) avec une modification, on ne va pas seulement les interactions les acteurs avec la solution, mais aussi les interactions de la solution avec les acteurs en indiquant le sens d'interaction avec une flèche.

Le diagramme suivant présent les cas d'utilisation de vue solution :

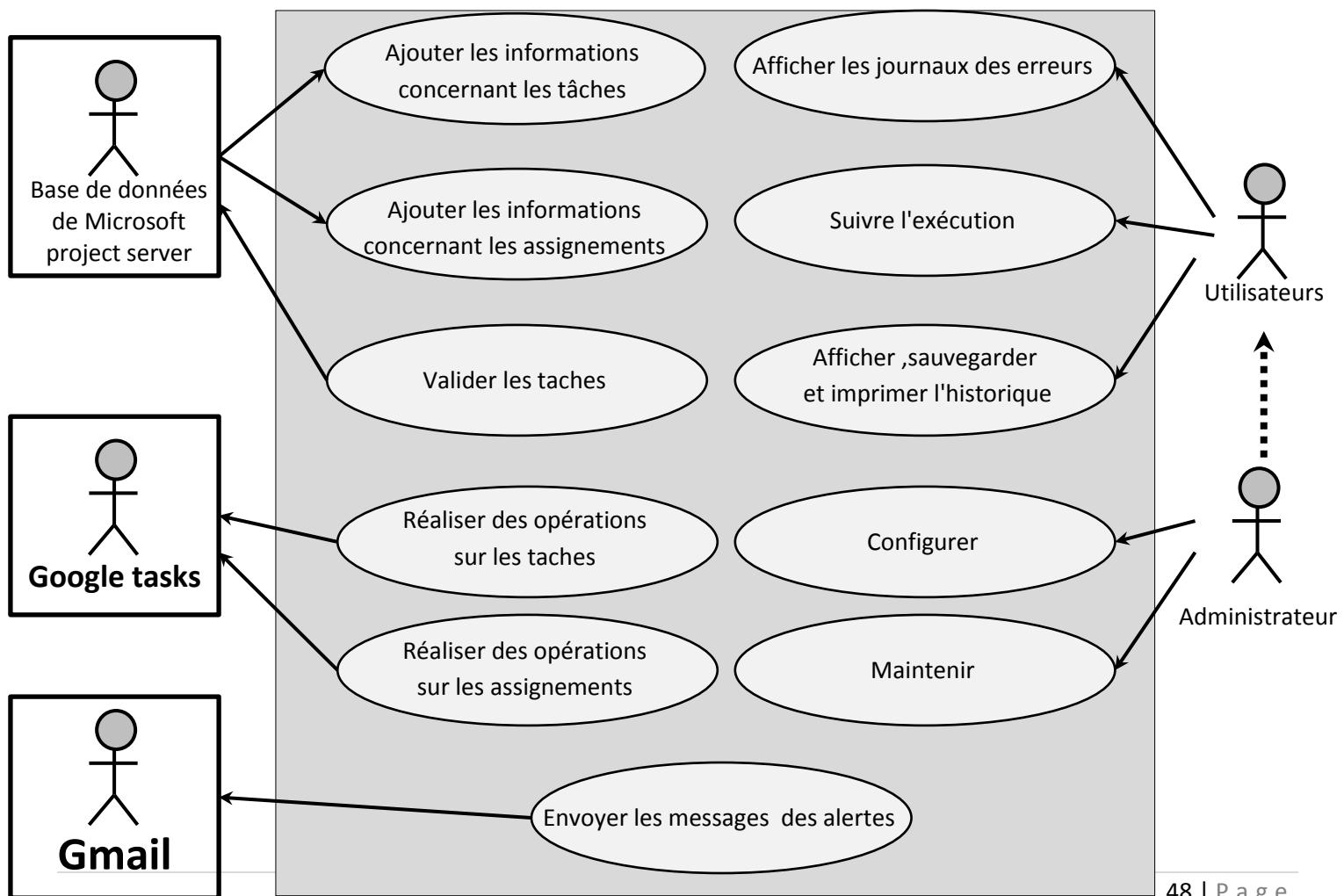


Figure 27 : le diagramme des cas d'utilisation de projet(vue solution)

## Explication :

On va expliquer certains cas d'utilisation et leur relation avec les acteurs :

- **Ajouter les informations concernant les tâches:** en utilisant des curseurs , on va copier des informations concernant des tâches à partir des bases de données de microsoft project server vers la base de données de la solution lorsque un utilisateur de Ms project réalise l'une des opérations sur les tâches citées dans le cahier de charges : Page 28 .
- même chose pour le cas d'utilisation : Ajouter les informations concernant les assignements.
- **Valider les tâches :** c'est le seul cas où la solution interagit avec La base de données de ms project server , lorsque un utilisateur valide une tâche en « Google tasks » , la solution vérifie la validation et modifie des données dans La base de données de Ms project server (C'est le cas dont nous avons parlé dans la page 41 et présenté dans le diagramme de contexte )
- **Réaliser des opérations sur les tâches et les assignements :** en lisant les données dans la base donnée de la solution on va réaliser des opérations de gestion des tâches et des assignements cité dans le cahier de charges (page 28) en « Google tasks » de chaque utilisateur.
- **Envoyer les messages d'alerte :** la solution va envoyer des messages d'alerte en utilisant un protocole SMTP.

## C)Vue logique :

La vue logique constitue la principale description architecturale d'un système. Cette vue décrit le système en termes d'objets et de classes. La vue logique permet d'identifier les différents éléments et mécanismes du système à réaliser. Elle permet de décomposer le système en abstractions et constitue le cœur de la réutilisation.

Dans notre cas on va utiliser le diagramme des composantes pour exprimer les principaux composants dans notre projet en donnant une petite définition pour chaque composant avant d'entrer dans les détails dans les prochaines sections.

Le diagramme suivant présente les composantes de notre solution et leurs types de communication avec les acteurs présentés dans les sections précédentes.

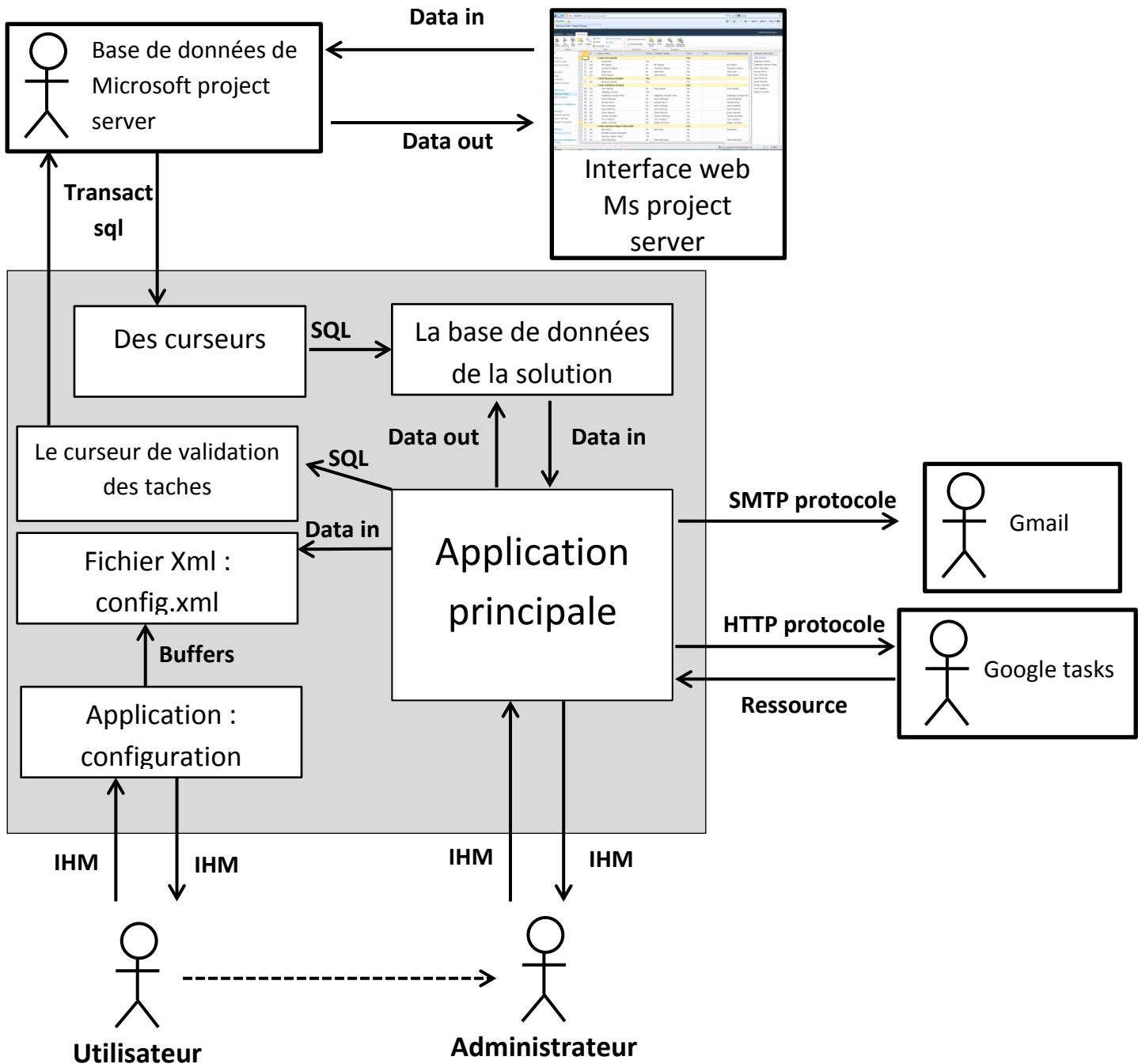


Figure 28 : Le diagramme de composant de la solution

## **Description des composants :**

On va donner une description générale de chaque composants de la solution avant d'entrer dans les détails plus tard.

### **➤ Application : configuration .**

C'est une application java contient une interface graphique pour configurer la solution, il contient des données d'accès et d'autorisation, seulement l'administrateur utiliser cette application, cette application génère comme résultat le fichier **config.xml**

### **➤ Fichier Xml : config.xml**

Ce fichier contient des données d'accès et d'autorisation afin d'être utilisé par l'application principale.

### **➤ La base de données de la solution**

C'est une base de donnée contient des informations sur les taches, les opération à réaliser, historiques ... etc.

cette base de donnée utilise SQL server 2008 comme un SGBD .

### **➤ Des curseurs**

La fonctionnalité principale des curseurs c'est de détecter les informations stratégiques concernant les taches et les assignements dans la base de données de ms projet server et de copier ces informations au base de données de la solution

### **➤ Le curseur de validation des taches**

On a séparé ce curseur des autres, car c'est le seul curseur qui modifie des données dans la base de données de ms project sever , ce curseur lit des données dans la base de données de la solution , détecte des taches qui sont complets et modifie leur pourcentage dans Ms projet server .

### **➤ L'application principale**

C'est une application java , c'est le cœur de la solution , elle gère tous les opération principaux , et elle est le plus importante composant

## D)Vue physique:

Dans cette vue on va présenter la distribution des composantes de la solution sur les composants physiques, ce distribution est la distribution réel de la solution.

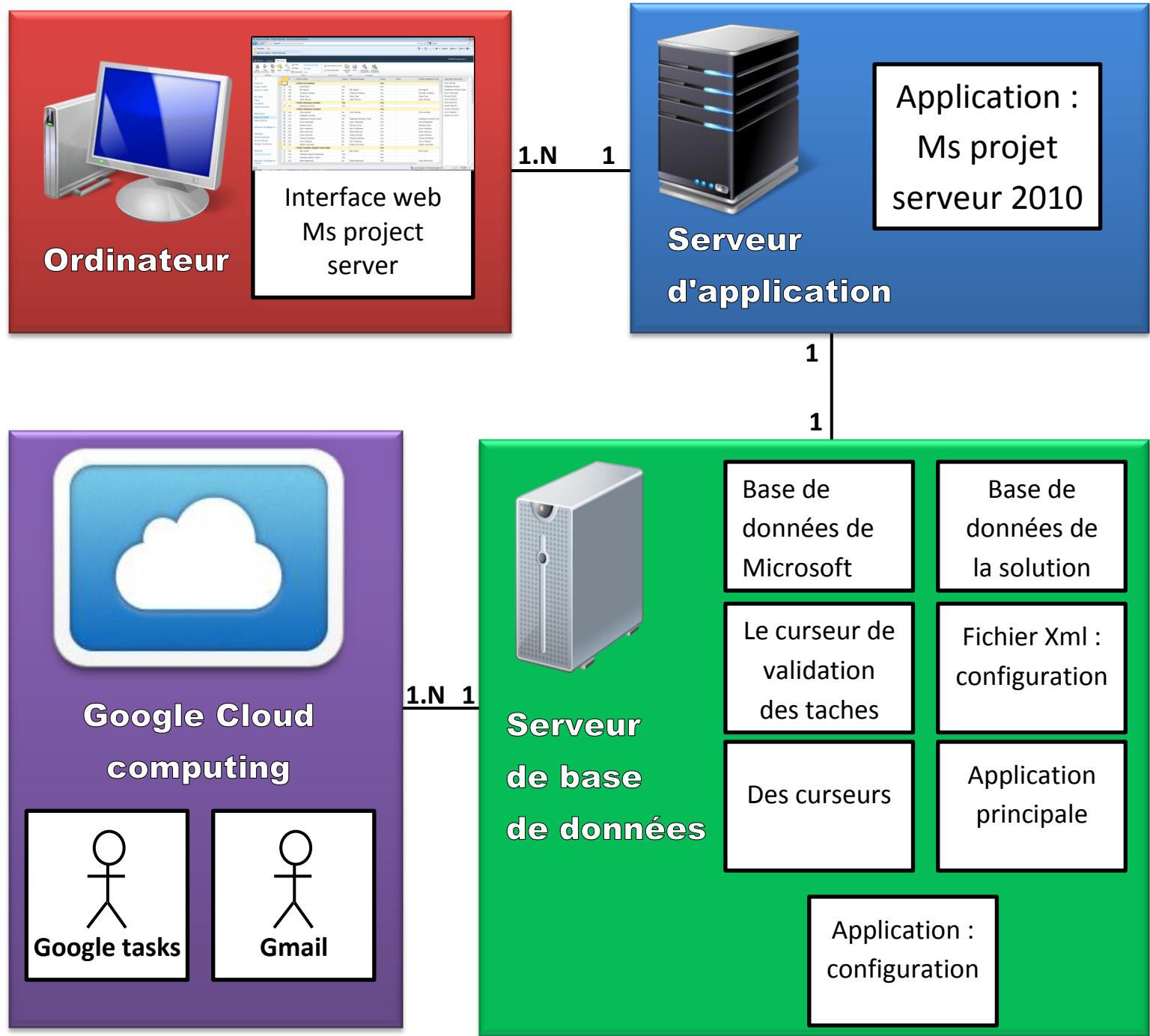


Figure 29 : La vue physique de la solution

## 2- Structure des bases de données

Dans cette section on va parler premièrement de la structure de la base de données de Microsoft Project server 2010, et quelles sont les principales tables dans cette base de données, puis on va construire le MCD et déduire MLD.

Puis on va parler de la base de données de notre solution, on va présenter premièrement le dictionnaire des données, puis MCD, ensuit MLD

Finalement on va parler des curseurs en donnant une définition de chaque curseur existé.

### A) La base de données de Microsoft project server :

La base de données de Microsoft project server 2010 consiste de plusieurs bases de données physiques (plus de 24 base de données physiques), mais il existe trois bases de données qu'ils contiennent des informations sur les tâches, les assignements, les projets ..etc :

- **ProjectServer\_Archive** : cette base de données contient des informations sur les projets archivés.
- **ProjectServer\_Draft**: cette base de données contient des informations sur les projets en brouillant ou bien en cours de la réalisation.
- **ProjectServer\_Published**: cette base de données est la plus importante pour notre projet , elle contient des informations sur des projets publiés et enregistrées , et a partir de cette base de données les curseurs extraient tous les informations des tâches qu'on les besoin dans la solution .

Chacune de ces bases de données contiennent plus de 80 tables, mais dans l'extraction des informations on a besoin seulement de 4 tables Et ils sont tous situés au même base de données (**ProjectServer\_Published**)

Le tableau suivant présente le dictionnaire des données de Ms project server qu'on a utilisé dans la réalisation de notre solution :

Nom de la table	Définition	Les noms des attributs	Les types des attributs	Définitions des attributs
MSP_RESOURCES	Contient des informations sur l'utilisateur de Ms project server	RES_UID	varchar(200)	L'UID d'utilisateur ou bien de ressource
		WRES_EMAIL	varchar(200)	L'email d'utilisateur
		RES_NAME	varchar(200)	Le nom d'utilisateur
MSP_PROJECTS	Contient des informations sur les projets	PROJ_UID	varchar(200)	L'UID de projet
		PROJ_NAME	varchar(200)	Le nom du projet
		WPROJ_DESCRIPTION	varchar(200)	La description du projet
		WRES_UID	varchar(200)	L'UID de chef de projet de

Nom de la table	Définition	Les noms des attributs	Les types des attributs	Définitions des attributs
				la table « MSP_RESOURCES »
MSP_TASKS_SAVED	Contient des informations sur les taches créées et enregistrées	TASK_UID PROJ_UID TASK_NAME TASK_START_DATE TASK_FINISH_DATE TASK_DUR TASK_PCT_COMP	varchar(200) varchar(200) varchar(200) Datetime Datetime Int Int	L'UID de la tache L'UID de projet dans la table « MSP_PROJECTS » Le nom de la tache La date de début de la tache La date de la fin de la tache Le durée de la tache ( unité c'est milliseconde ) Le pourcentage de complétude de la tache
MSP_ASSIGNMENTS_SAVED	Contient des information sur les assignements	TASK_UID RES_UID	varchar(200) varchar(200)	L'UID de la tache dans la table « MSP_TASKS_SAVED » L'UID d'utilisateur « MSP_RESOURCES »

Tableau 10 : Le dictionnaire de données de Ms projet server

## ➤ Le modèle conceptuel de données et Le modèle logique de données

A partir de de dictionnaire des données on peut construire le MCD suivante :

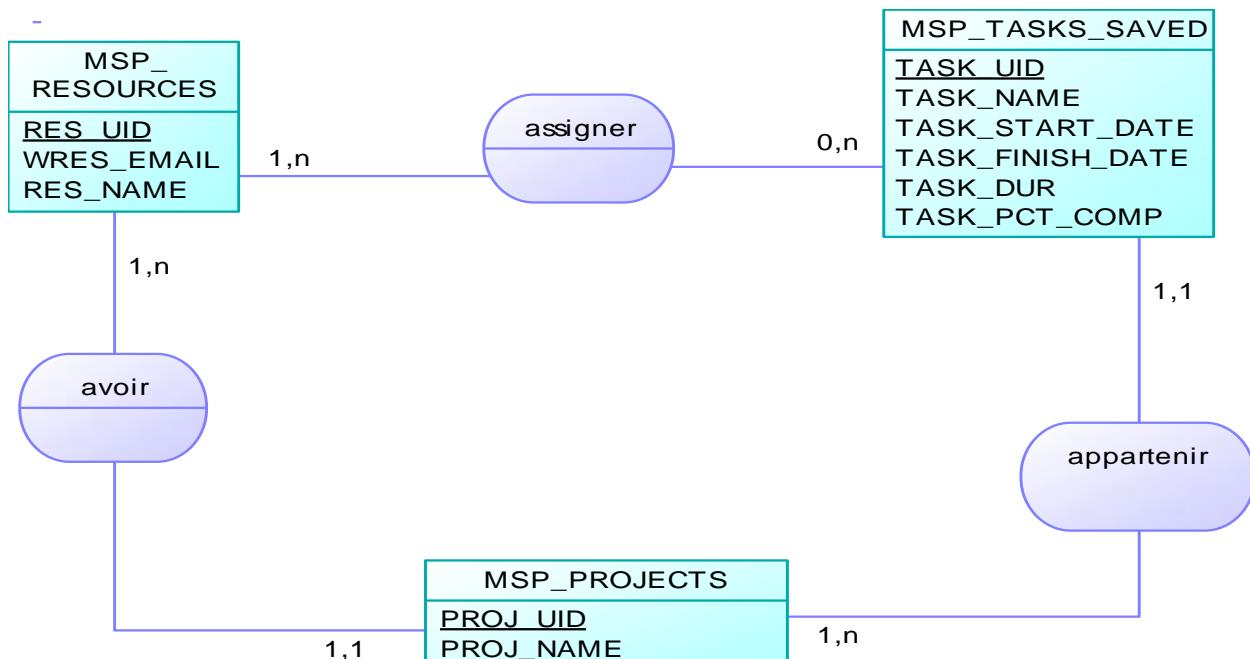


Figure 30 : Le MCD de la BD de MS project server

Et à partir de MCD on va déduire le MLD suivant :

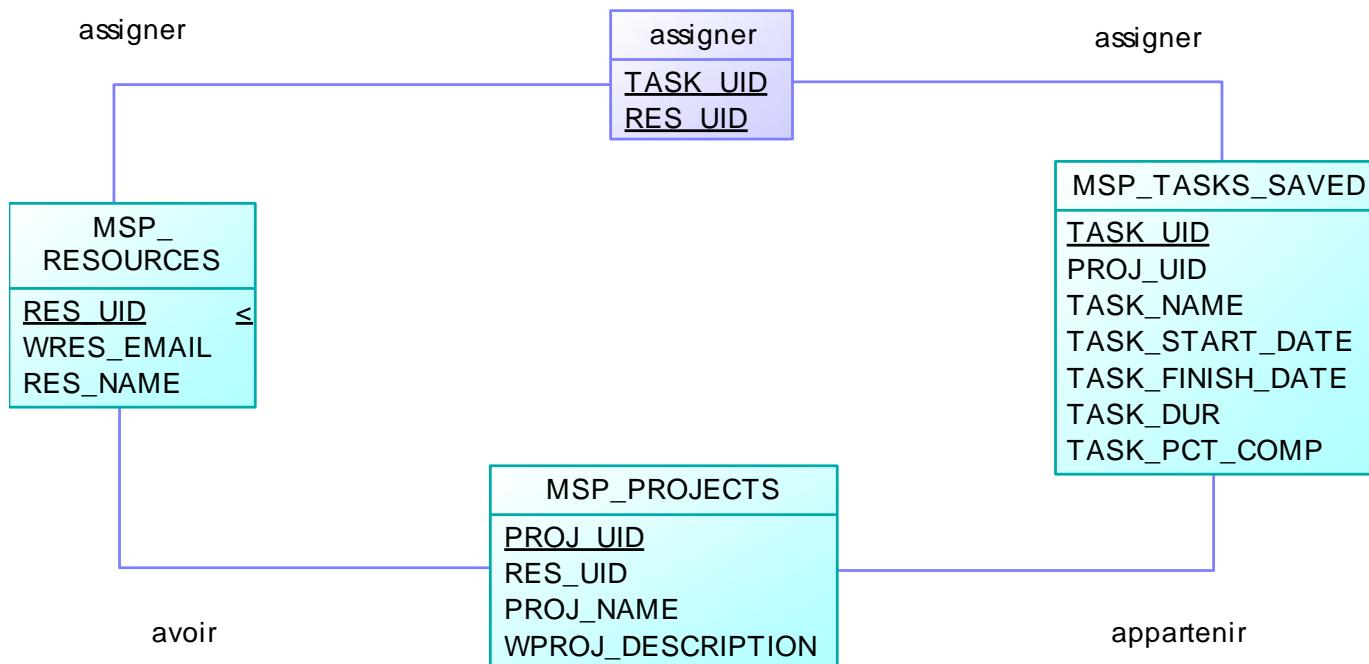


Figure 31 : Le MLD de la BD de MS project server

## B) La base de données de la solution :

### ➤ Le dictionnaire de données

Notre base de données consiste de 8 tables, les tableaux suivants représentent les dictionnaires de données de notre solution, certain des attributs signifient la même chose que les attributs de BD de ms projets server :

Nom de la table	person	
Description	Cette table contient des informations sur les utilisateurs (pas des chefs de projet)	
Le nom d'attribut	Type d'attribut	Définitions d'attribut
id_person	INT IDENTITY(1,1)	La clé primaire d'utilisateur
person_email	varchar(200)	L'email d'utilisateur
person_nom	varchar(200)	Le nom et le prénom d'utilisateur
id_person_ms_projet	varchar(200)	L'UID d'utilisateur dans la base de donnée de MS projet server , on utilise ce attribut pour comparer les utilisateur a partir de leurs UID .

Tableau 11: Le dictionnaire des données de la table Person

<b>Nom de la table</b>	<b>chef_projet</b>	
Description	Elle contient des informations sur les chefs de projet	
Le nom d'attribut	Type d'attribut	Définitions d'attribut
id_chef_projet	INT IDENTITY(1,1)	La clé primaire de chef de projet
chef_projet_email	varchar(200)	L'email de chef de projet
chef_projet_nom	varchar(200)	Le nom et le prénom de chef de projet
id_chef_projet_ms_projet	varchar(200)	L'UID de chef de projet dans la base de donnée de MS projet server .

Tableau 12 : Le dictionnaire des données de la table chef\_projet

<b>Nom de la table</b>	<b>projet</b>	
Description	Elle contient des informations sur les projets	
Le nom d'attribut	Type d'attribut	Définitions d'attribut
id_projet	INT IDENTITY(1,1)	La clé primaire de projet
id_chef_projet	INT	La clé de chef de projet (clé étrange)
projet_nom	varchar(200)	Le nom de projet
projet_description	varchar(MAX)	La description de projet

Tableau 13 : Le dictionnaire des données de la table projet

<b>Nom de la table</b>	<b>tache_person</b>	
Description	Cette table contient des information sur les assignments déjà traités par l'application principale .	
Le nom d'attribut	Type d'attribut	Définitions d'attribut
id_tache	INT	La clé de la tache
id_person	INT	La clé d'utilisateur
id_tache_google	varchar(200)	La clé de la tache dans les serveurs de Google
est_complet	bit	<ul style="list-style-type: none"> <li>« true » si cette tâche est marquée comme complet dans Google tasks .</li> <li>« false » si non</li> </ul>

Tableau 14 : Le dictionnaire des données de la table tache\_person

<b>Nom de la table</b>	<b>taches</b>	
Description	Elle contient des informations sur les taches	
Le nom d'attribut	Type d'attribut	Définitions d'attribut
id_tache	INT IDENTITY(1,1)	La clé primaire de la tache
id_projet	INT	La clé de projet (clé étrange)
titre_tache	varchar(MAX)	Le nom de la tache
date_debut_tache	datetime	La date de début de la tache
date_fin_tache	Datetime	La date de fin de la tache
duree_tache	int	La durée de la tache
est_alter	bit	<ul style="list-style-type: none"> <li>« true » si les messages d'alertes de cette tache ont déjà envoyé</li> <li>« false » si non</li> </ul>
est_complet_ms	bit	<ul style="list-style-type: none"> <li>« true » si cette tâche est marquée comme complet dans Ms projet server .</li> <li>« false » si non</li> </ul>
id_tache_ms_project	varchar(200)	L'UID de la tache dans la base de donnée de Ms projet server .

*Tableau 15: Le dictionnaire des données de la table taches*

<b>Nom de la table</b>	<b>operation_tache_person</b>	
Description	Cette table contient des informations sur les assignements qu'ils ont insérés par les curseurs mais pas encore traités par l'application principale. L'application principale lit les données de cette table et applique les opérations nécessaires puis elle supprime tous les données qu'elle a traités de cette table.	
id_tache	INT	La clé de la tache (clé étrangère)
id_person	INT	La clé d'utilisateur (clé étrangère)
operations	varchar(10)	<ul style="list-style-type: none"> <li>« c » : s'il s'agit d'une opération d'un assignement en Ms projets server .</li> <li>« d » : s'il s'agit d'une opération de suppression d'un assignement en Ms projets server .</li> </ul>

*Tableau 16: Le dictionnaire des données de la table operation\_tache\_person*

<b>Nom de la table</b>			<b>operation_tache</b>
Description			Cette table contient des informations sur les taches qu'ils ont insérées par les curseurs mais pas encore traitées par l'application principale. L'application principale lit les données de cette table et applique les opérations nécessaires puis elle supprime tous les données qu'elle a traitées de cette table.
id_operation_tache	INT IDENTITY(1,1)	Clé primaire de operation_tache	
id_tache	INT	La clé de la tache (clé étrangère)	
operation_tache	varchar(10)	Le type d'opération :	<ul style="list-style-type: none"> <li>« c » : s'il s'agit d'une opération de création d'une tache en Ms projets server .</li> <li>« d » : s'il s'agit d'une opération de suppression d'une tache en Ms projets server .</li> <li>« u » : s'il s'agit d'une opération de mise à jour d'une tache en Ms projets server .</li> <li>« com » : s'il s'agit d'une opération de validation d'une tache en Ms projets server.</li> </ul>

*Tableau 17 : Le dictionnaire des données de la table operation\_tache*

<b>Nom de la table</b>			<b>historique_taches</b>
Description			Cette table contient des informations d'historisation, à chaque fois l'application principale réalise une opération elle insère des données dans cette table
Le nom d'attribut	Type d'attribut	Définitions d'attribut	
id_historique	INT IDENTITY(1,1)	La clé primaire d'historique	
historique_date	datetime	La date d'historique	
operations	varchar(20)	Le type d'opération	
projet_nom	varchar(200)	Le nom de projet	
titre_tache	varchar(300)	Le nom de la tache	
date_debut_tache	Datetime	La date de début de la tache	
date_fin_tache	Datetime	La date de la fin de la tache	
duree_tache	varchar(50)	La durée de la tache	
person_email	varchar(200)	L'email d'utilisateur	
person_nom	varchar(200)	Le nom d'utilisateur	
chef_projet_email	varchar(200)	L'email de chef de projet	
chef_projet_nom	varchar(100)	Le nom de chef de projet	

*Tableau 18 : Le dictionnaire des données de la table historique\_taches*

## ➤ Le modèle conceptuel de données

on a construit ce MCD , qui respecte exactement les caractères et contraintes citées dans le cahier de charges page 28:

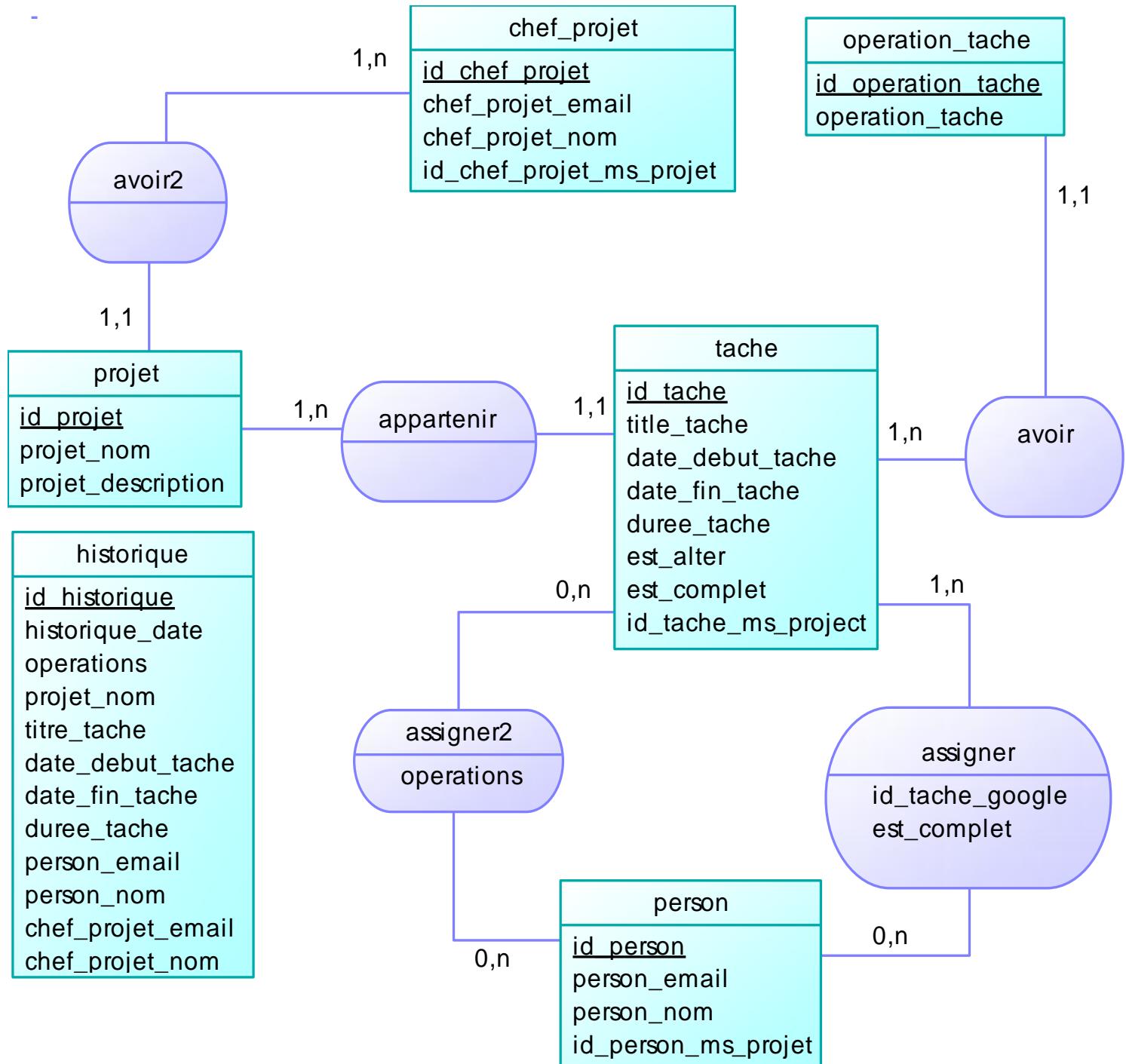


Figure 32 : Le MCD de la solution

## ➤ Le modèle logique de données

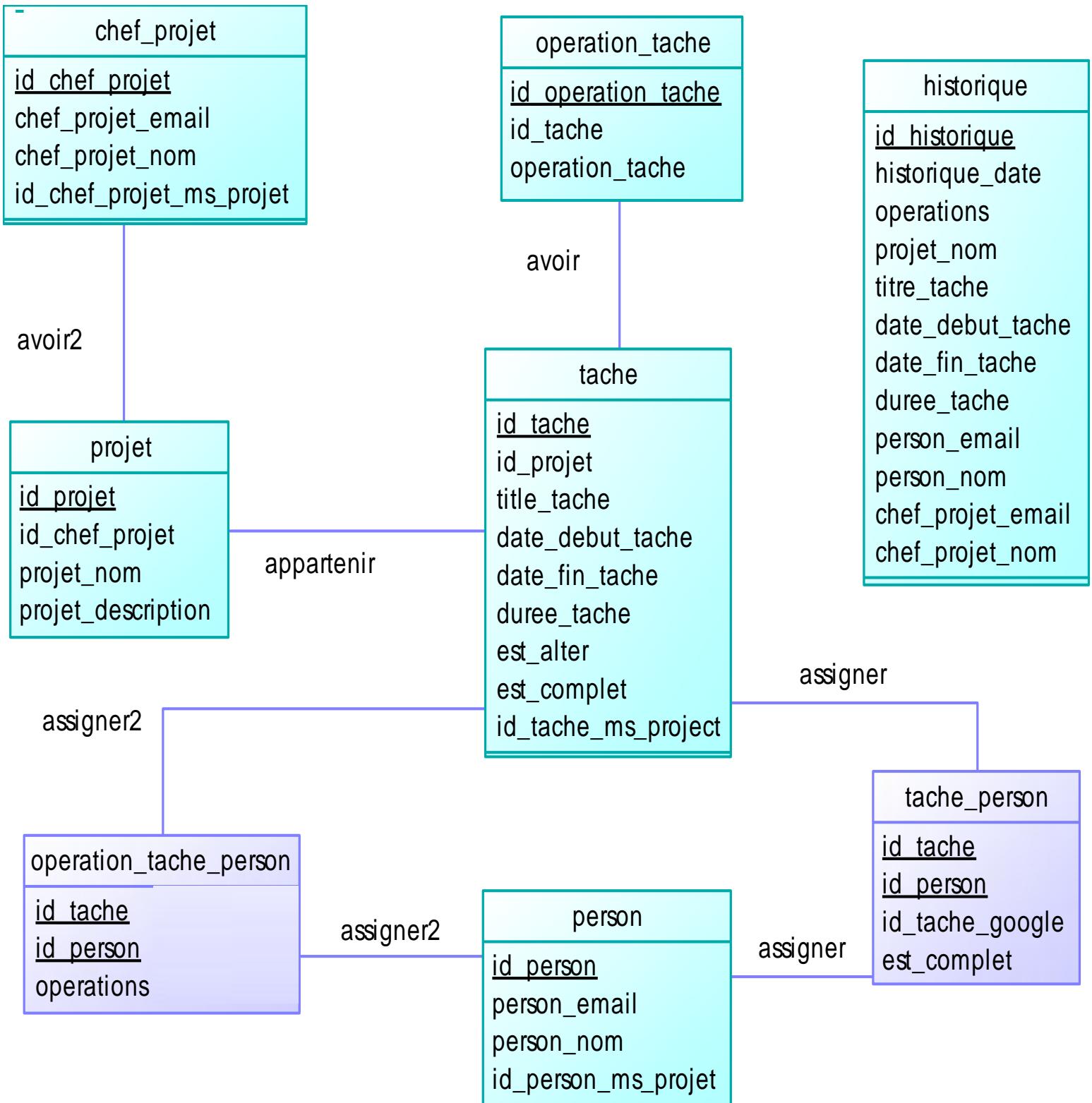


Figure 33: Le MLD de la solution

## C)Les curseurs :

Les curseurs sont un composant très important dans notre solution , leur objectif principales est de comparer les données dans la base de données de la solution et la BD de MS project server , il existe plusieurs curseurs dans ce projet , dans cette partie on va décrire chaque curseur ainsi donner leurs fonctionnalités principales , et on va finir avec un exemple d'un simple curseur de notre solution.

### ➤ **Le curseur de création des tâches et des assignments**

L'objectif principale de ce curseur est d'insérer dans la base de données de la solution les taches et assignments qu'ils ont été créés dans ms projetc et non dans Google tasks .

Premièrement on lance une requête qui sélectionne tous les informations concernant : les taches, les projets, les utilisateurs et les assignement reliant a des taches qui n'ont pas UID dans la table « Taches » .

Après ça on commence l'insertion de ces informations :

1. insertion des données des utilisateurs dans la table « person »
2. insertion des données des chefs de projet dans la table « chef\_projet »
3. insertion des données des projets dans la table « projet »
4. insertion des données des taches dans la table « taches »

et à chaque fois on vérifie si les données sont déjà existés dans les tables ou non , si oui on ne les insère pas , sinon on les insère .

Finalement on insère des données dans la table « operation\_tache\_person » avec l'attribut operations = 'c'

### ➤ **Le curseur de la suppression des tâches**

L'objectif principale de ce curseur est de notifier la base de données de la solution S'il trouvait tâches supprimés de Ms projet server et toujours existent dans la base de donnée de la solution.

Au contraire du curseur précédent on cherche maintenant des taches qu'elles ont UID dans la table « taches » et non dans la table « MSP\_TASKS\_SAVED »

Après le lancement de la requête de comparaissant on insérer dans la table « operation\_tache » la clé primaire de la tache supprimée et 'd' dans l'attribut « operation\_tache », et on laisse le reste de travail pour l'application principale

### ➤ **Le curseur de la mise à jour des tâches**

L'objectif principale de ce curseur est de notifier la base de données de la solution S'il trouvait tâches mis à jour de Ms projet server.

Ce curseur compare les taches qui ont la même UID dans les deux bases, en comparant les attributs de la table « tache » avec les attributs de la table « MSP\_TASKS\_SAVED » .

1. TASK\_NAME avec titre\_tache
2. TASK\_START\_DATE avec date\_debut\_tache
3. TASK\_FINISH\_DATE avec date\_fin\_tache
4. TASK\_DUR avec duree\_tache

Après le lancement de la requête de comparaissant on insérer dans la table « operation\_tache » la clé primaire de la tache supprimée et ‘u’ dans l’attribut « operation\_tache », et on laisse le reste de travail pour l’application principale

### ➤ **Le curseur de la suppression des assignments**

L’objectif principale de ce curseur est de notifier la base de données de la solution S'il trouvait des assignments supprimés de Ms projet server et toujours existent dans la base de donnée de la solution.

Ce curseur compare les attributs « id\_tache\_ms\_project » de table « taches » , « id\_person\_ms\_projet » de table « person » en joitirant ces deux table ensemble avec la table « tache\_person » en les comparant avec les attributs « TASK\_UID » et « RES\_UID » de la table « MSP\_ASSIGNMENTS\_SAVED »

Après le lancement de la requête de comparaissant on insérer dans la table « operation\_tache\_peron» la clé primaire de la tâche et d’utilisateur reliant a l’assignment supprimé et ‘d’ dans l’attribut « operations », et on laisse le reste de travail pour l’application principale.

### ➤ **Le curseur de la complétude des tâches en Google apps**

L’objectif principale de ce curseur est de notifier la base de données de la solution S'il trouvait des taches complets dans Ms projet server et pas dans Google tasks .

Ce curseur sélectionne les UID des taches qu’ils ont l’attribut « TASK\_PCT\_COMP » de table « MSP\_TASKS\_SAVED » égale à 100 avec l’attribut « est\_complet\_ms » de table « taches » égale ‘false’

Après le lancement de la requête on insérer dans la table dans la table « operation\_tache » la clé primaire de la tache supprimée et ‘com’ dans l’attribut « operation\_tache », et on laisse le reste de travail pour l’application principale

### ➤ **Le curseur de la mise à jour des ressources**

L’objectif principale de ce curseur est de mettre à jour des ressources ( des emails et des noms soit des utilisateurs ou bien des chefs de projet) chaque fois on les change dans la base de données de Ms project server .

La mécanisme de ce curseur est la même que le curseur de la mise à jour des taches, il compare l’email et le nom des utilisateurs et des chefs de projet qu’ils ont les même UID en B de MS projet server .

Après le lancement de la requête principale, le curseur mit à jour directement les ressources dans la base de données locale.

## ➤ **Le curseur de validation des taches en ms Project server**

C'est le seul curseur qui modifie des valeurs en BD de Ms Project server , l'objectif de ce curseur et de valider les taches en MS project lorsque les taches ont été valider en Google tasks , ce curseur agit seulement lorsque tous utilisateur assignées à une tache ont la marqué comme complet en Google tasks .

Ce curseur sélectionne toutes les tâches qui ont l'attribut «est\_complet »='true' dans la table « tache\_person » dans tous les lignes ou « id\_tache » égale a la clé primaire de la tâche.

Après le sectionnement des données on arrive à la phase modifiée les données en MS project server.

Ce curseur modifie plus de 35 champ en Ms project , ces champs distribués sur les trois base de données ProjectServer\_Archive , ProjectServer\_Draft et ProjectServer\_Published.

Car on veut que les taches apparaitre comme complets dans tous les types de projet.

Par exemple dans la base de données « ProjectServer\_Published » ce curseur mit à jour des données :

- De la table « MSP\_ASSIGNMENTS\_SAVED » :
  - ASSN\_ACT\_WORK= ASSN\_WORK pour dire que la durée de travail sur cet assignement égale à la dure de la tache lui-même.
  - ASSN\_Rem\_WORK=0.000000 : la durée de travail restant sure l'assignement égal à 0.
  - ASSN\_PCT\_WORK\_COMPLETE=100 le pourcentage d'assignement égal à 100%.
- Même chose pour la table « MSP\_TASKS\_SAVED » :
  - TASK\_ACT\_DUR= TASK\_SCHED\_DUR
  - TASK\_ACT\_WORK= TASK\_WORK
  - TASK\_Rem\_DUR=0
  - TASK\_PCT\_COMP=100
  - TASK\_PCT\_WORK\_COMP=100s
  - TASK\_Rem\_WORK= 0.000000

Et on répète la même chose pour les bases de données restantes.

## ➤ **Exemple d'un curseur :**

Voici un exemple de le plus simple curseur dans la solution , c'est le curseur Le curseur de la complétude des tâches en Google apps :

```
CREATE PROCEDURE curseur_complet_tache
AS
DECLARE
    @number_tache int,
```

```

@id_ms_tache varchar(200),
@id_tache INT

DECLARE curseur_complet_tache CURSOR FOR
    select TASK_UID

        from ms_taches taches

        WHERE taches.TASK_PCT_COMP='100'
        AND EXISTS
        (
            select t.id_tache
            from taches t
            WHERE taches.TASK_UID = t.id_tache_ms_project
            AND t.est_complet_ms='false'
        );
    OPEN curseur_complet_tache
    FETCH curseur_complet_tache INTO      @id_ms_tache

    WHILE @@FETCH_STATUS = 0
    BEGIN
        set @id_tache =(select id_tache from taches where
        id_tache_ms_project=@id_ms_tache);

        set @number_tache = ( select count(*)  from
        operation_tache WHERE id_tache=@id_tache  AND operation_tache='com' );

        IF (@number_tache = 0)
        BEGIN
            Insert into operation_tache(id_tache,operation_tache)
            values(@id_tache,'com');
        END
    FETCH curseur_complet_tache INTO      @id_ms_tache

    END
    CLOSE curseur_complet_tache
    DEALLOCATE curseur_complet_tache
    GO

```

### 3- Les applications Java de la solution

Il existe deux applications de java dans la solution, la première application pour le but de générer le fichier « config.xml » a fin de configurer l'application, et la deuxième c'est l'application principale, elle considère comme le cœur de la solution.

Dans cette partie on va parler de ces deux applications en donnant une description générale de leurs fonctionnalités.

#### A) L'application de La génération du fichier config.xml:

Le rôle de cette application est de générer un fichier « config.xml » pour configurer l'application principale, il contient une interface graphique pour utiliser par l'administration.

Elle contient quatre catégories d'accès et de configuration :

- **La base de données** : sont des variable d'accès a la base de donnée cette catégorie contient 4 variable : le nom de server et numéro de port , le nom d'utilisateur de BD , le mot de passe et finalement le nom de la base de données.
- **Le serveur SMTP** : ici on trouve les variables de server SMTP qu'on va utiliser pour l'envoie des messages d'alerte , elle contient 2 variables : le nom de server SMTP et le mot de passe .
- **Google Apps** : sont des variables de configuration des comptes de Google apps : le compte de service , chemin de clé secrète et finalement le nom d'application .
- **Autre** : cette catégorie contient des variables de configuration d'application principale, ils sont : Nombre de jours avant l'envoie des messages d'alerte. Nombre d'heures pour réexécution d'application Nombre maximum de lignes d'historique.

Utilisation de cette application donne plus de flexibilité et de contrôle au solution car on pour changer des paramètres sans modifier le code source.

#### B) L'application principale :

Cette application est considérée comme le cœur de la solution, elle réalise tous les opérations importantes.

Piètremment on va donner une description de rôle de l'application puis on va parler de leur mécanisme :

## ➤ **Le rôle d'application :**

On peut résumer les rôles d'application en :

- L'application lire des données insérées par les curseurs dans la base de données de la solution et réalise des opérations dans Google tasks .
- Elle est aussi responsable de l'envoie des messages d'alerte
- Elle exécute les curseurs car les curseurs sont enregistrés comme des procédures stockées dans SQL server et l'application utilise les données d'accès pour exécuter ces curseurs.
- L'application est aussi responsable l'historisation des opérations.
- Génération des journaux d'erreurs
- On peut imprimer et sauvegarder l'historique.
- L'application affiche l'état d'exécution et la date de prochaine réexécution.

## ➤ **Les composants d'application :**

On deviser cette application selon des composants :

- **Gestion de base de données** : ce composant est responsable de toutes les interactions avec la BD ce qui inclut le lancement des requêtes de sélections et d'insertion...Etc.
- **Gestion d'IHM** : ce composant est responsable d'afficher l'état d'exécution et d'interaction avec l'utilisateur.
- **Gestion d'historique** : enregistre tous les opérations réalisées par l'application.
- **Gestion de SMTP** : gère la communication avec Gmail pour envoyer les messages d'alerte.
- **Gestion de communication avec Google tasks** : il réalise des opérations sur Google tasks , et gère les réponses des serveurs de Google .
- **Gestion de fichier de configuration** : pour lire des variables dans ce fichier et les utiliser pour paramétriser l'application .
- **Gestion des curseurs** : pour exécuter les curseurs à partir des procédures stockées

## ➤ **Le mécanisme d'application :**

Dans cette partie on va déclarer les phases d'exécution d'application principale on présentant les sous phases successivement.

L'application s'exécute automatiquement après un période de temps déterminé par l'administrateur ( dans la phase de configuration) , et donne aussi la possibilité à l'utilisateur d'exécuter application même avant la prochaine tour ( en cliquant sur le bouton « Re-exécuter » ) .

Elle s'exécute en 9 phases :

### **1. L'initialisation :**

- Construit les éléments d'interface graphique.
- Lire des variables de « config.xml »
- connecter à la base de données de la solution

## **2. L'exécution des curseurs :**

- Exécution du curseur d'insertion des tâches et des assignments
- Exécution du curseur de la suppression des tâches
- Exécution du curseur de la mise à jour des tâches
- Exécution du curseur de la suppression des tâches et des assignments
- Exécution du curseur de la complétude des tâches en Google apps
- Exécution du curseur de la complétude des tâches en Ms projet
- Exécution du curseur de la mise à jour des ressources

## **3. La création des tâches:**

- Lancement de la requête principale
- le stockage de résultats de la requête
- insertion des tâches en Google tasks
- insertion des tâches en base de données [tache\_person]
- Historisation [historique\_taches]
- vidage du cache [operation\_tache\_person]

## **4. La suppression des tâches:**

- Lancement de la requête principale
- le stockage de résultats de la requête
- Suppression des tâches en Google Agenda
- Historisation[historique\_taches]
- Suppression des tâches de la base de données [taches]
- Suppression des tâches de la base de données [operation\_tache]
- Suppression des tâches de la base de données [operation\_tache\_person]
- Suppression des tâches de la base de données [tache\_person]

## **5. La suppression des assignments:**

- Lancement de la requête principale
- le stockage de résultats de la requête
- sectionnement [id\_tache\_google] de la base de données
- Suppression des tâches en Google Agenda
- Historisation [historique\_taches]
- vidage du cache [operation\_tache\_person]
- Suppression des assignments du base de données [operation\_tache\_person]

## **6. La mise à jour des tâches:**

- Lancement de la requête principale
- le stockage de résultats de la requête
- La mise à jour des tâches en Google Agenda
- Historisation[historique\_taches]
- vidage du cache[operation\_tache\_person]

## **7. La complétude des tâches:**

- Lancement de la requête complétude des Taches en Ms-project
- le stockage de résultats de la requête
- Vérification de la complétude du tache en Google Agenda
- Mise à jour les taches en base de données
- Historisation[historique\_taches]
- Lancement de la requête complétude des Taches en Google Agenda
- le stockage de résultats du requête
- Historisation[historique\_taches]
- vidage du cache[operation\_tache\_person]
- Enregistrement des Modification Dans la base de données[tache\_person]

## **8. L'envoie des alertes:**

- Préparation des variables de connexion SMTP
- le stockage de résultats de la requête
- L'envoie des messages des alertes
- Enregistrement des Modification Dans la base de données [taches]
- Historisation[historique\_taches]

## **9. Rapport de l'exécution :**

- Afficher Le temps d'exécution
- Afficher l'état d'exécution.
- Afficher la date de prochaine exécution

## 4- La sécurité de la solution

La sécurité informatique est l'ensemble des moyens techniques, organisationnels, juridiques et humains nécessaires et mis en place pour conserver, rétablir, et garantir la sécurité des systèmes informatiques. Elle est intrinsèquement liée à la sécurité de l'information et des systèmes d'information.

La sécurité est un facteur très important dans n'importe qu'il projet informatique, dans cette section on va parler de côté sécurité de notre solution , on va premièrement parler de l'authentification de Google Apps en donnant une description général de type d' l'authentification qu'on a utilisé , puis on va parler de le fichier « config.xml » , ensuit la sécurité pour utilisation de SMTP , puis en idée générale de la sécurité de notre base de données , et finalement on va parler un peu de la gestion des erreurs dans notre solution .

### A) l'authentification de Google Apps:

En générale, L'authentification est la procédure qui consiste, pour un système informatique, à vérifier l'identité d'une personne ou d'un ordinateur afin d'autoriser l'accès de cette entité à des ressources.

Dans notre projet notre application principale jeu le rôle d'un client pour les servers de Google apps (spécialement Google tasks) , par conséquence notre application doit être authentifié pour avoir l'autorisation de réaliser des opérations sur les taches en Google tasks des utilisateurs de la solution .

La phase d'authentification c'était la phase le plus difficile dans la réalisation de projet, car on a besoin d'une authentification quelle n'est pas classique .

Dans l'utilisation de Google Api il existe 2 type d'authentification, 3-legged OAuth et 2-legged OAuth

On a utilisé 2-legged OAuth car c'est la plus adaptée au notre projet , dans cette partie on va parler première d'authentification 3-legged OAuth et pourquoi on ne peut pas l'utiliser dans notre cas , en suit on va présenter les fonctionnalités de 2-legged OAuth en la comparant avec 3-legged OAuth.

Remarque : tous les sources d'information professionnel sur « Google API » en général sont en anglais , c'est pourquoi on va présenter certain terme et diagramme en anglais.

#### ➤ 3-legged OAuth :

Ce type d'authentification pour Google apps et le plus classique et le plus utilisé pour les applications « Google Api » :

Le diagramme de séquence suivant présente le scénario d'utilisation de notre application avec l'authentification 3-legged OAuth pour obtenir avec succès l'accès au Google tasks d'utilisateur.

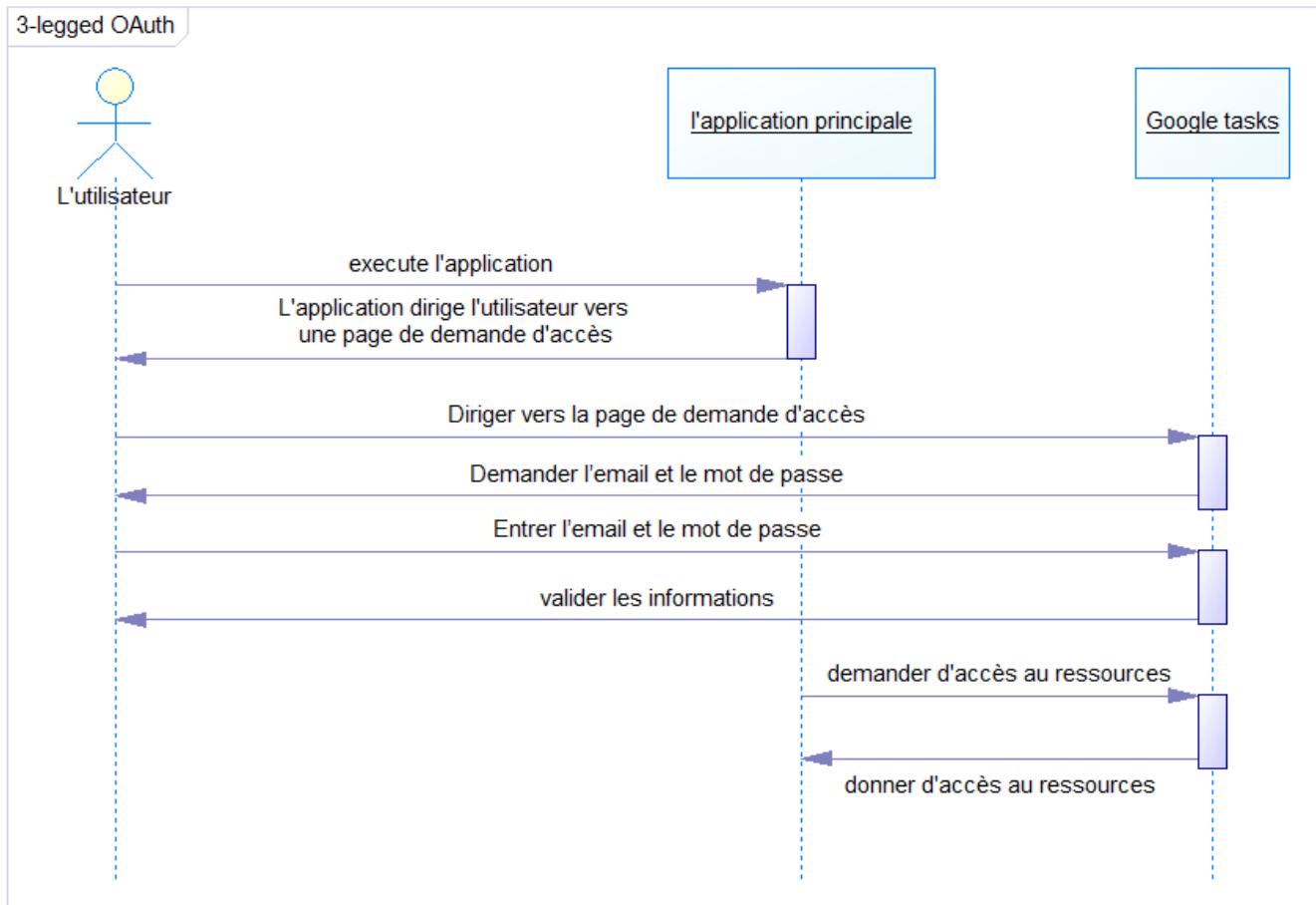
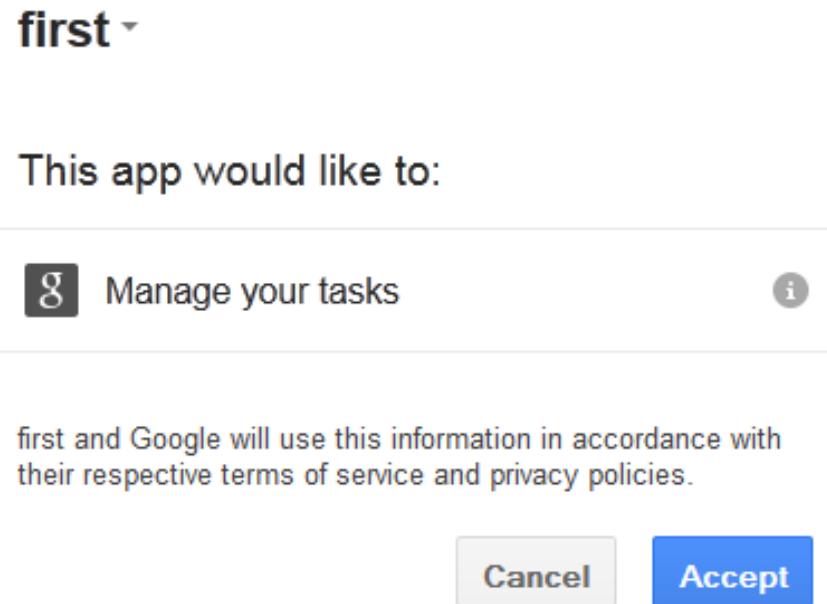


Figure 34 : Diagramme de séquence de la solution avec l'authentification 3-legged OAuth

Donc à partir de ce diagramme on peut remarquer 3 choses :

18. L'utilisateur faut qu'il exécute l'application lui-même, c.à.d. à chaque fois on veut créer une tache dans un agenda de Google il faut qu'on envoie l'application au utilisateur ciblé pour l'exécuter et nous donne l'accès et cela n'est pas pratique et impossible pour notre projet (car on a plusieurs utilisateur et on traite plusieurs taches)
19. L'utilisateur doit entrer son email et mot de passe (de compte de Google apps) chaque fois on veut réaliser une opération.
20. Finalement on remarque que les procédures sont très long et ça va affecter le temps d'exécution de solution et la rendre impossible d'appliquer (car on va perdre beaucoup de temps par conséquence notre solution sera inutile )

Voici un exemple de page de demande d'accès:



*Figure 35: un exemple de page de demande d'accès*

**Conclusion :** à cause de ces trois raisons j'ai essayé de trouver une nouvelle méthode d'authentification et après une recherche long (voir la planification page 43) j'ai trouvé la méthode 2-legged OAuth qui va nous donner une solution pour cette problématique.

#### ➤ 2-legged OAuth :

Ce type d'authentification fonctionne avec Google Apps for business ( Page 19) .

Pour utiliser 2-legged-Oauth il faut premièrement identifier l'application en Google Apps, pour ça on va utiliser Google Api console (page 21) Pour obtenir ce qu'on appelle « Service account » , ce type de compte contient 3 choses :

- **Client ID:** c'est la clé de « Service account » , on l'utilise pour identifier et autoriser l'application dans un demain de « Google apps »
- **Private key :** c'est un fichier d'extension .p12 (Personal Information Exchange File) , ce fichier représente un clé privée d'application , il est téléchargeable juste une fois lorsque vous créer « Service account »
- **Email address:** c'est un email de demain @developer.gserviceaccount.com on l'utilise avec « private key » pour identifier l'application.

Après la création de « Service account », on utilise **Client ID et le lien**

« <https://www.googleapis.com/auth/tasks> » dans page d'administration de notre demain de « Google apps » pour autoriser « Service account ».

Et on utiliser **Private key et Email address** pour identifier notre application pour obtenir les accès.

Le diagramme de séquence suivant présente le scénario d'utilisation de notre application avec l'authentification 2-legged OAuth pour obtenir avec succès l'accès au Google tasks d'utilisateur.

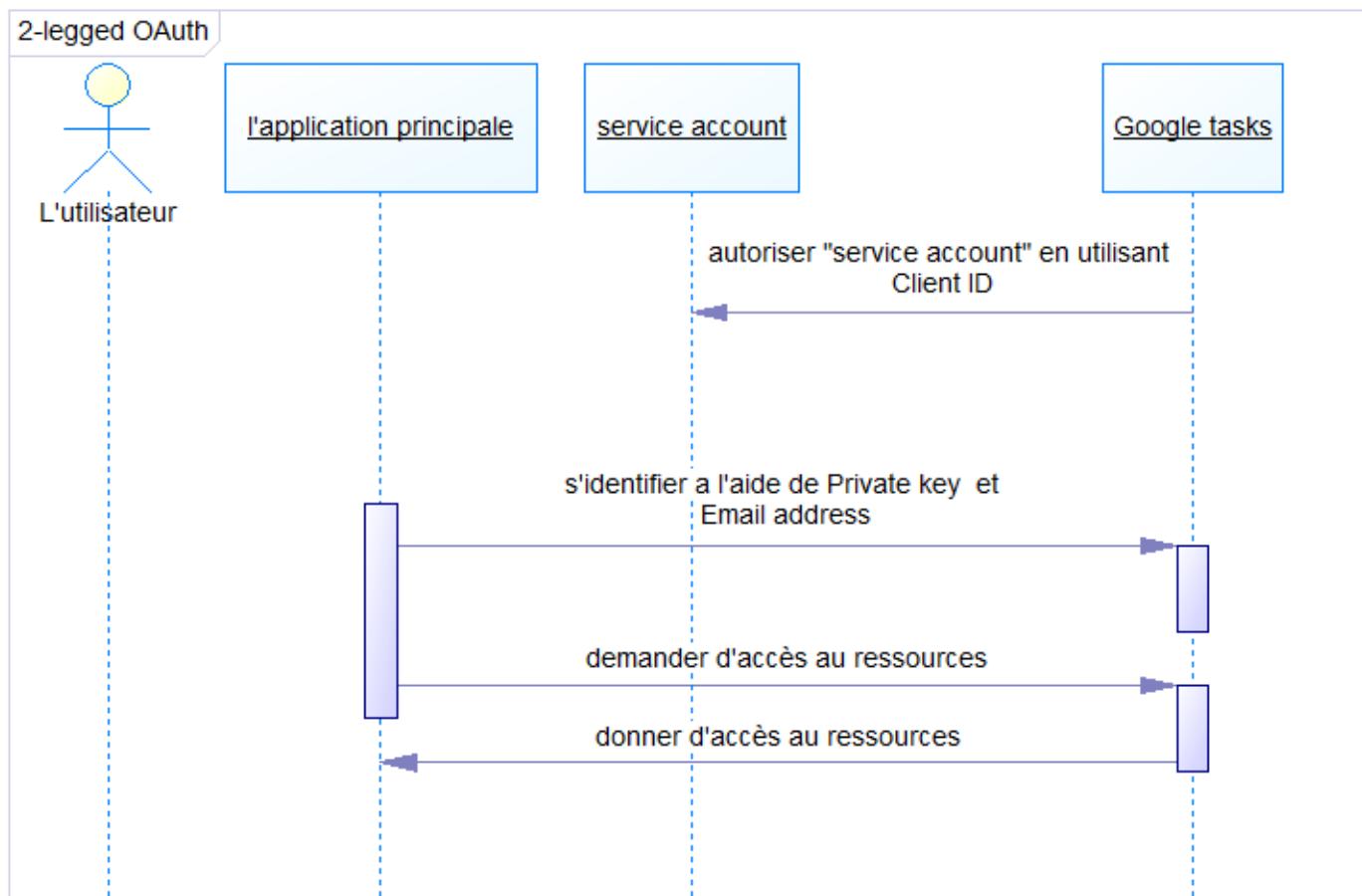


Figure 36 : Diagramme de séquence de la solution avec l'authentification 3-legged OAuth

Comme le diagramme montre l'utilisateur n'interagit pas avec les autres acteur, et on a pas besoin d'email et de mot de passe d'utilisateur, **par conséquence cette méthode est la plus adaptée à notre solution.**

## B) La communication SMTP :

SMTP est un protocole de communication utilisé pour transférer le courrier électronique (courriel) vers les serveurs de messagerie électronique.

Dans notre projet on a utilisé les serveurs Gmail comme notre serveur de messagerie électronique et on a utilisé la connexion TSL .

Transport Layer Security (TLS) est un protocole qui garantit la confidentialité entre les applications de communication et leurs utilisateurs sur Internet. Lorsqu'un serveur et le client communiquent, TLS garantit qu'aucun tiers ne peut être espionné ou altérer les messages.

TLS fonctionne suivant un mode client-serveur. Il fournit les objectifs de sécurité suivants :

- l'authentification du serveur ;
- la confidentialité des données échangées (ou session chiffrée) ;
- l'intégrité des données échangées ;
- de manière optionnelle, l'authentification ou l'authentification forte du client avec l'utilisation d'un certificat numérique ;
- la spontanéité, c'est-à-dire qu'un client peut se connecter de façon transparente à un serveur auquel il se connecte pour la première fois ;
- la transparence, qui a contribué certainement à sa popularité : les protocoles de la couche d'application n'ont pas à être modifiés pour utiliser une connexion sécurisée par TLS. Par exemple, le protocole HTTP est identique, que l'on se connecte à un schème http ou https.

## C)La base de données:

Pour notre base de données on a utilisé des contraintes pour sécuriser les entrées.

Il existe 3 types des contraintes qu'on a utilisées :

- **Les Contraintes d'unicités** : c'est le fait d'avoir une valeur unique d'une donnée pour une attribue d'une table , on a utilisé pour éviter la répétition des certains données : par exemple l'UID d'une tache ou l'email d'une utilisateur , car par exemple si un email se trouve deux fois dans notre base de données on peut tromper dans la création des taches et comme résultat on peut créer les mauvaises tâches pour faux email, Cela peut conduire à des fuites d'informations sur les projets d'entreprise .
- **Les Contraintes de format** : pour contrôler les formats d'entrées car un mauvais format d'une variable peut bloquer des procédures d'exécution et générer des erreurs.
- **Les Contraintes des clés étrangères** : pour assurer les liens entre les tables , ce type de contraintes assure l'insertion et la suppression des données , c.a.d on ne peut pas insérer une donnée qui se référence à une autre donnée dans une autre table qui se n'existe pas .
- **Les Contraintes « NOT NULL »** : pour assurer l'existence des données stratégiques par exemple : email d'utilisateur , le nom de la tâche , la date de début de la tâche , les UIDs ...etc.

## D) La sécurité de fichier « config.xml » :

Le fichier « config.xml » contient des données d'accès, et ces données sont très sensibles et peuvent causer des risques de sécurité sur l'ensemble du système, donc il est très important de sécuriser ce fichier.

Pour le sécuriser, on a appliquée deux techniques :

1. **Masquage des noms de variable** : dans la génération de ce fichier on change l'ordre des variables et leurs nom (on les nomme variable1 , variable2 ...etc)
2. **Cryptage des données** : de deuxièmes on applique des techniques de cryptographie, à l'aide d'un algorithme AES avec une clé de 16 bit.

La figure suivante représente un exemple d'un fichier « config.xml » généré par la solution.

```
1  <?xml version="1.0" encoding="UTF-8" standalone="no"?><information><variable1>CVzanaJV221KmXRpbwnxcA==#13;
2  </variable1><variable2>4AgzvLKrDYWqKPfpMqB8JGcbDWMWiGZwfmTq31pD7KA==#13;
3  </variable2><variable3>59+mfrYRYOShOMQjHSUb1w==#13;
4  </variable3><variable4>AsajLswGTTGvh03wK6pEVQ==#13;
5  </variable4><variable5>um0Nnz6gXL9bNSHMijVbC7GUkGnaHyyD1QXOGISQGtAb1879UGxicA1MJB2gh/F8#13;
6  </variable5><variable6>Pwy64DFQmVQuK0pudAaCNw==#13;
7  </variable6><variable7>myMmfOSyL1J6CrN2ugfrcw==#13;
8  </variable7><variable8>Bp8LRCJ00/WCVq1n8A0lhQ==#13;
9  </variable8><variable9>48044hjgAC1fU7N7K0s5ig==#13;
10 </variable9><variable10>HIOUN0JxB7EG6mJWb0JIJkfZ7tciI+eHwBZeq8QpDdQ==#13;
11 </variable10><variable11>FY30EBk6SyHVkfZHcvxD1WcbDWMWiGZwfmTq31pD7KA==#13;
12 </variable11></information>
```

Figure 37 : un exemple d'un fichier « config.xml » généré par la solution

## E) La gestion des erreurs :

Comme tous les grands projets, on a implémenté un système de gestion des erreurs, qui contient 50 erreurs de différents impacts sur la solution, on distingue 3 catégories des erreurs :

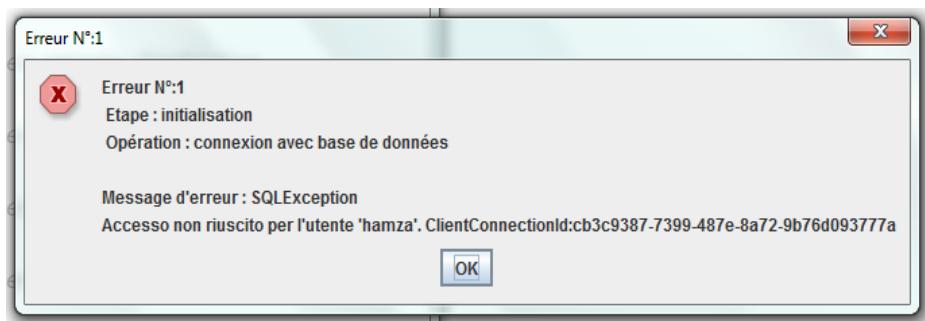
- **Erreurs bloquantes d'exécution** : sont des erreurs qui causent l'arrêt d'exécution d'application et elles sont trois :
  - **Erreur N °1 décrypte les données de fichier config.xml** : cette erreur peut être causée par : non existence de fichier « Config.xml » ou la modification manuelle du fichier.
  - **Erreur N °2 connexion avec la base de données** : cette erreur peut être causée par : les informations d'accès à la base de données ne sont pas justes.

- **Erreur N ° :49 Le Chemin de clé secrète n'est pas valide.**
- **Erreurs bloquantes d'une phase:** sont des erreurs qui causent l'arrêt l'une des phases d'exécution citées dans page 66 , ce type des erreurs n'arrête pas l'exécution de l'application, mais on saute a la phase suivante en affichant un message d'erreur, cette erreur peut être causé par : mauvaise lancement de la requête principale par exemple.
- **Erreurs bloquantes d'une sous phase :** sont des erreurs qui causent l'arrêt l'une des sous phases d'exécution citées dans page 66 , ce type des erreurs n'arrête pas l'exécution de l'application, mais on saute a la sous phase suivante en affichant les erreurs dans un journal d'erreurs. cette erreur peut être causé par exemple par : un email invalide ou bien erreur d'historisation.

Pour les erreurs de premier et deuxième catégories on affiche un message d'erreur qui contient les informations suivantes :

- Numéro d'erreur
- Nom d'erreur
- La phase
- L'exception

Exemple d'un message d'erreur.



*Figure 38 : Exemple d'un message d'erreur généré par la solution*

Et à la fin d'exécution pour toutes les catégories des erreurs on prépare un journal des erreurs qui contient pour chaque erreur :

- Date d'erreur
- Numéro d'erreur
- Nom d'erreur
- La phase
- L'exception

Exemple :

```

1 2013/11/20 07:34:59
2 Erreur N°:1
3 Etape : initialisation
4 Opération : connexion avec base de données
5
6
7 Message d'erreur : SQLException
8 Accesso non riuscito per l'utente 'hamza'. ClientConnectionId:cb3c9387-7399-487e-8a72-9b76d093777a

```

*Figure 39 : exemple du journal des erreurs*

# 5- L'interface graphique

Dans cette section on va donner une description détaillée d'interface graphique de la solution, en présentant leur caractéristique, en suit on va donner la liste des Critères ergonomiques qu'on a respecté

## A) La fenêtre d'application de configuration :

Cette interface graphique1 à le but de générer le fichier « config.xml » crypté, la figure suivant présente L'interface de l'application de configuration.

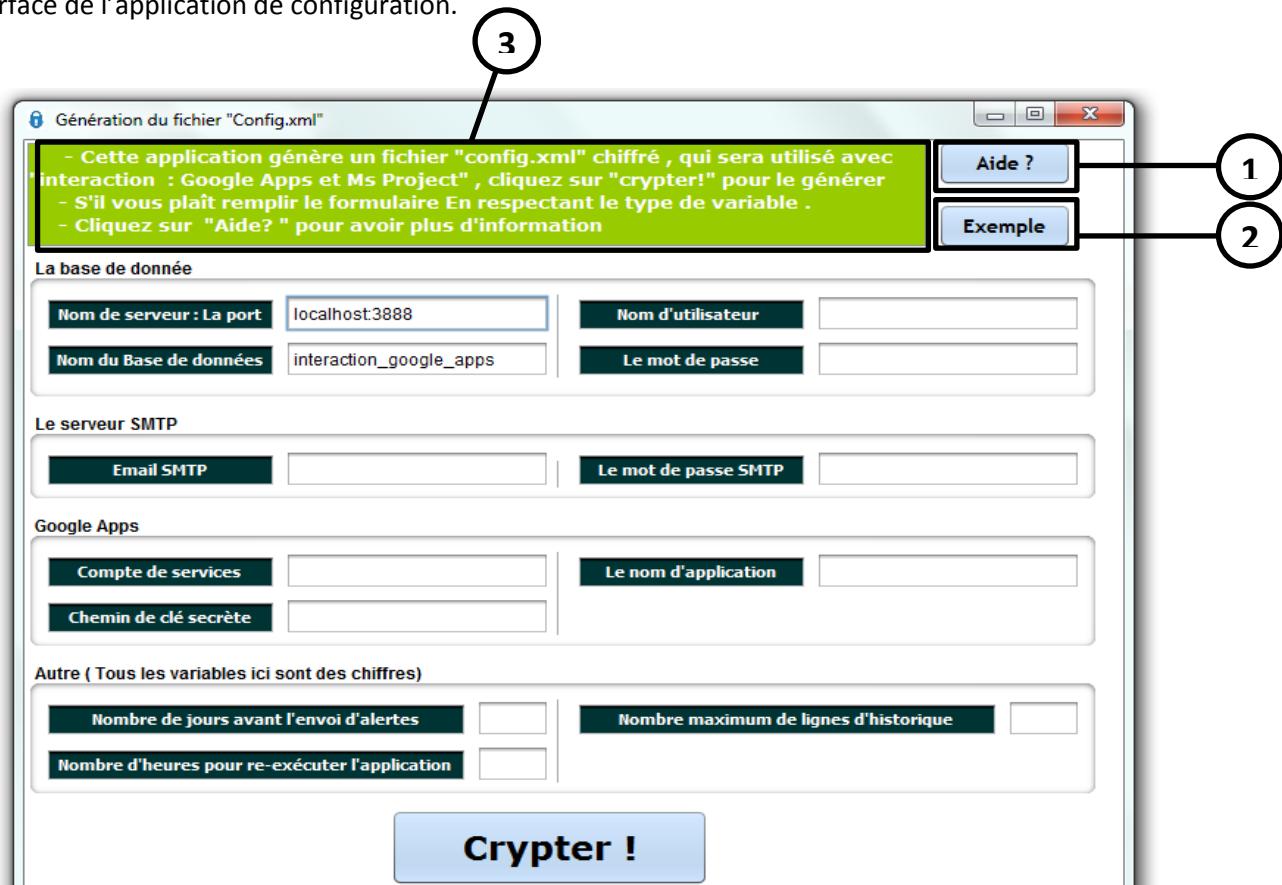


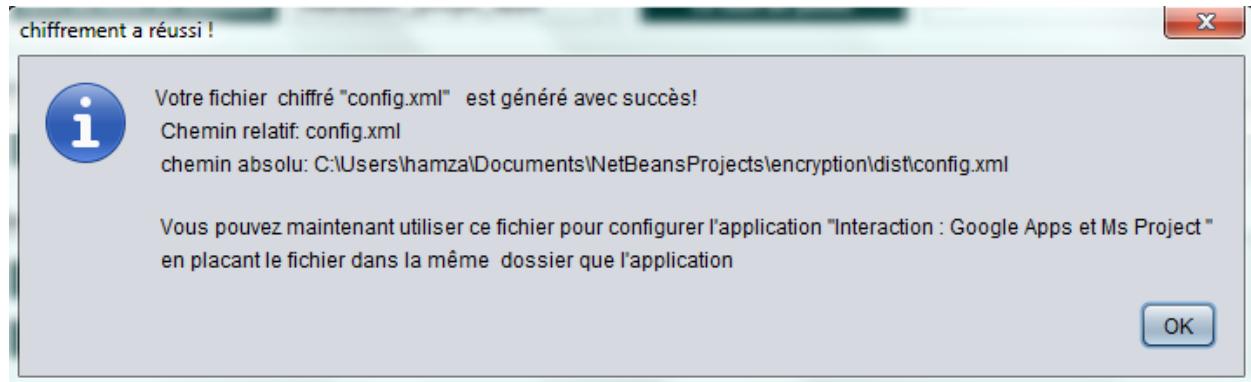
Figure 40 : La fenêtre d'application de configuration

Comme on remarque les variables de même nature sont regroupées dans des catégories, en donnant à chaque catégorie un nom.

Les éléments de cette interface sont :

- 1) Ce bouton affiche une page web d'aire qui contient des informations sur la configuration de l'application et des informations sur l'interface graphique et comment utilisé la solution.
- 2) Ce bouton affiche une image qui contient un exemple pré-rempli pour donner une idée sur le format des variables.
- 3) En haut de l'interface on trouve une introduction sur l'utilisation de l'application

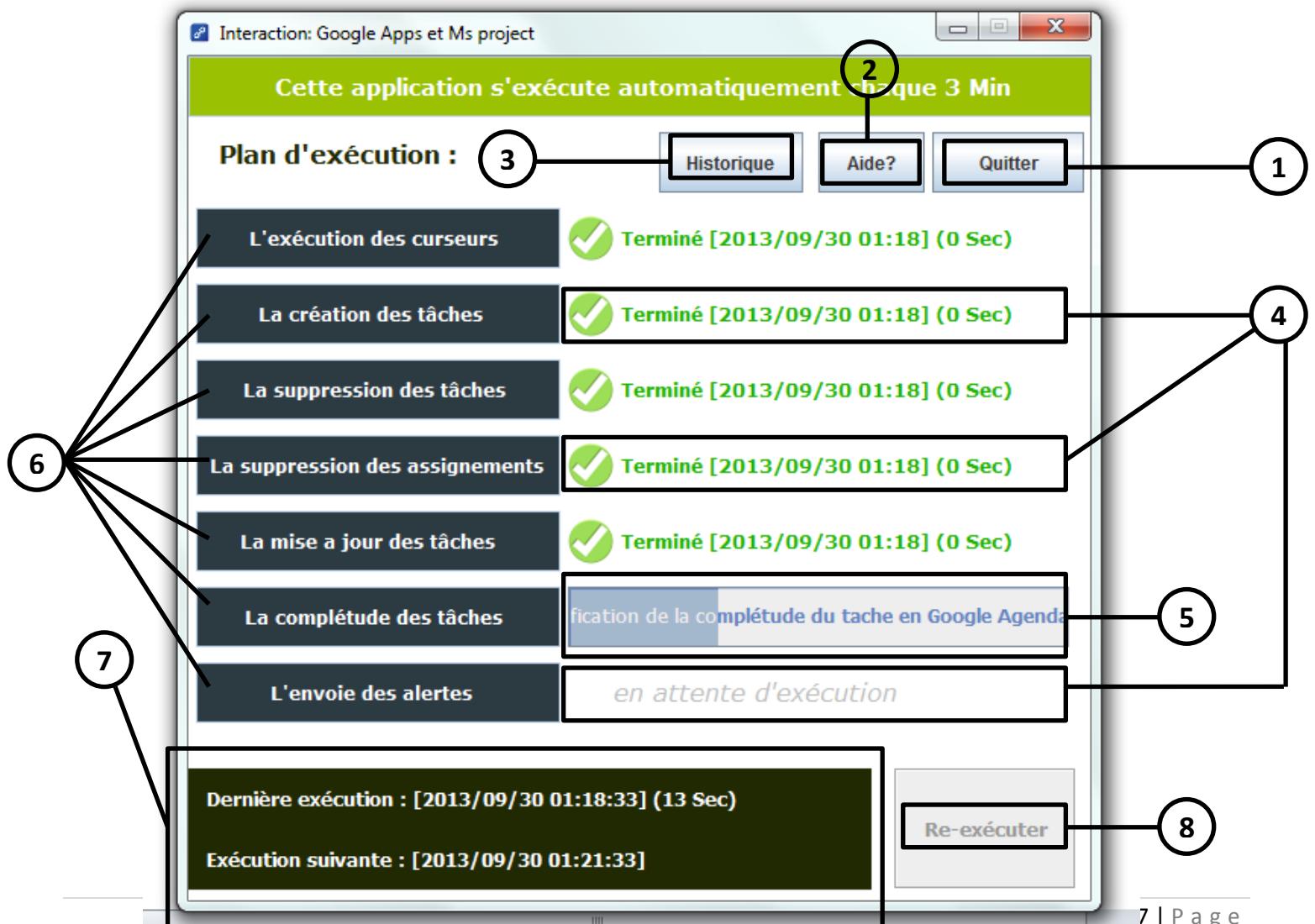
Après le remplissage des champs on clique sur « crypter ! » pour générer le fichier « Config.xml », en suit un message s'affiche les détails de la génération :



*Figure 41: le message du réussite de la génération de "config.xml"*

## B) La fenêtre d'application principale :

La figure suivante représente la fenêtre principale de l'application.



*Figure 42: la fenêtre principale d'application*

- 1) Cliquez sur ce bouton pour quitter l'application, on a mis ce bouton car si on clique sur « Fermer » l'application va être minimisé dans la barre des tâches.
- 2) Ce bouton s'affiche une page web d'aide.
- 3) Cliquez sur ce bouton pour afficher une autre fenêtre qui contient le journal d'historique
- 4) L'interface afficher l'état d'exécution de chaque phase, pour les phases terminées on affiche « la date d'exécution » et « la durée d'exécution »
- 5) Pour les phases en cours d'exécution, on affiche le pourcentage d'exécution et l'opération en cours d'exécution
- 6) On coupe l'exécution en des phases ( voir la page 66)
- 7) A chaque tour d'exécution on affiche les informations sur la dernière exécution et la date de la prochaine exécution.
- 8) Ce bouton sert à Re-exécuter l'application sans attendre la prochaine exécution, on le grise lorsque l'application en cours d'exécution.

Si on clique sur « Fermer », l'application se minimise dans la barre de tâche et continue sa exécution .

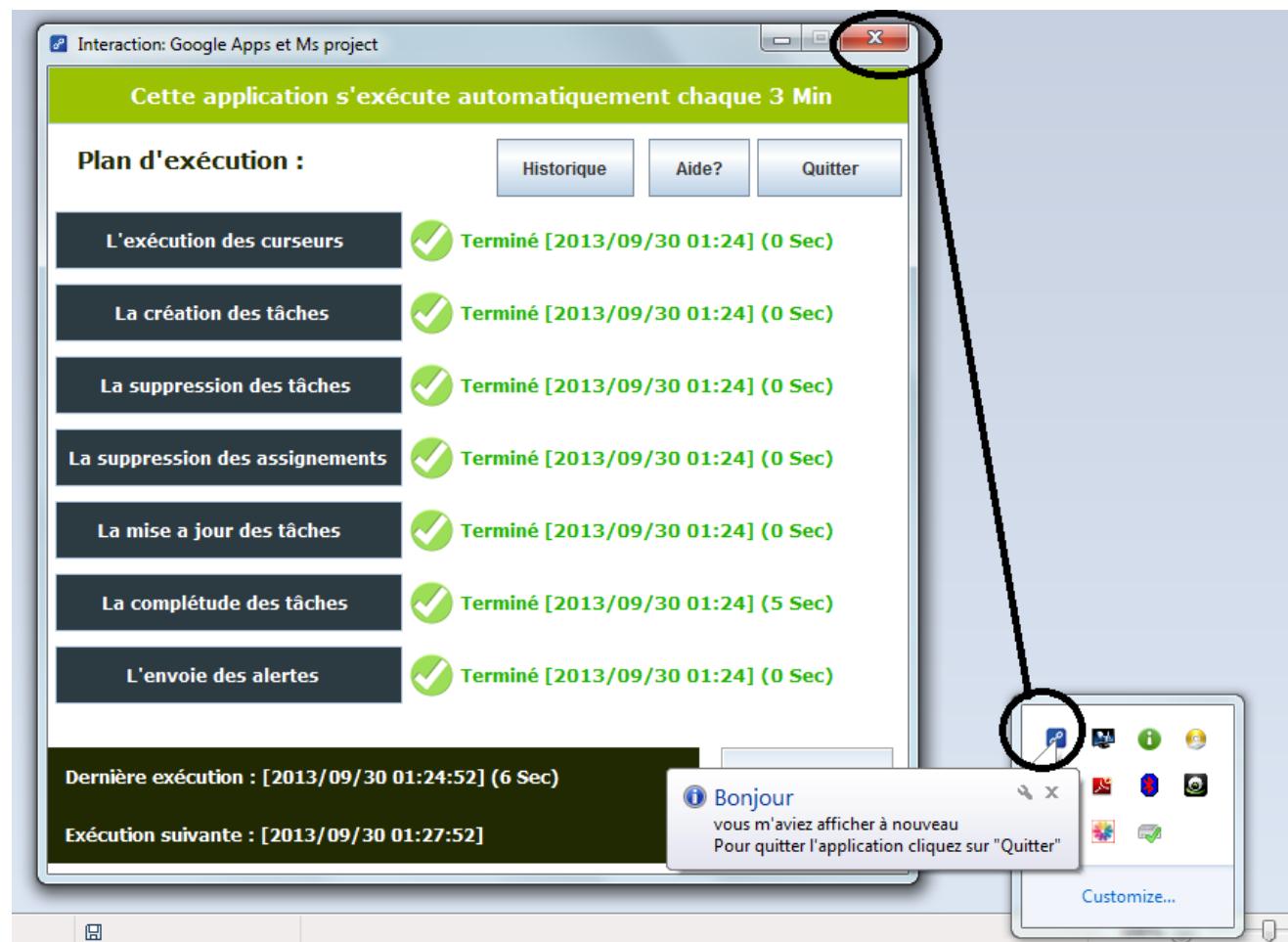


Figure 43 : minimisation d'application dans la base des tâches

## C) La fenêtre d'historique :

Si on clique sur le bouton « Historique », la fenêtre suivante va apparaître :

The screenshot shows the 'Historique' window with the following interface elements:

- Header:** 'Historique' (1) and a green banner with instructions: 'Cette page sert à afficher l'historique de tous les opérations effectuées par l'application.' and 'Vous pouvez filtrer les résultats en utilisant le panneau de recherche'.
- Search Bar:** 'Recherche : Par nom , par email ...etc' (1) with input 'elmetni hamza' and dropdown 'Opération : Tous'.
- Buttons:** 'Rechercher' (1), 'Sauvgarder' (2), and 'Aide ?'.
- Table:** A grid of historical operations with columns: id\_historique, historique\_date, operations, projet\_nom, titre\_tache, person\_email, and person\_nom. A row for operation 100 is highlighted.
- Details View:** A modal window titled 'Détails' (3) showing the details for operation 100:
 

<b>Clé historique :</b> 100	<b>Assigné à (email) :</b>
<b>Date :</b> 2013-09-29 21:15:07.78	<b>Le début du tâche :</b> 2013-08-13 15:00:00.0
<b>L'opération :</b> creation	<b>La fin du tâche :</b> 2013-08-30 17:04:00.0
<b>Le nom du projet :</b> projet de traitement d image	<b>La durée du tâche :</b> 2 Heures
<b>Le nom du tache :</b> tache 3	<b>Chef du projet :</b> yassine abdellah
<b>Tâche assigné a :</b> elmetni hamza	<b>Chef du projet (email) :</b> projet2@ensa-tetouan.mygbiz.com

Figure 44: La fenêtre d'historique

- 1) Utiliser ce bar pour effectuer des recherches et filtrer les résultats.
- 2) On clique sur ce bouton pour sauvegarder l'historique sous format PDF (ou autre format) ou bien pour l'imprimer
- 3) Clique sur une ligne et puis sur une flèche droite ou gauche pour afficher les détails d'historique.

La fenêtre suivante s'affiche en cliquant sur le bouton « sauvegarder » :

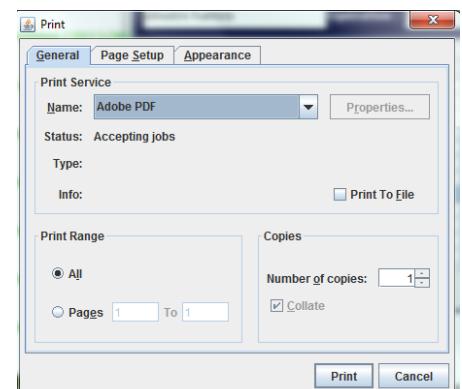


Figure 45 : la fenêtre de sauvegarde d'historique

## D) Exemple d'un message d'alerte :

Voici un exemple d'un message d'alerte envoyé automatiquement par l'application principale.



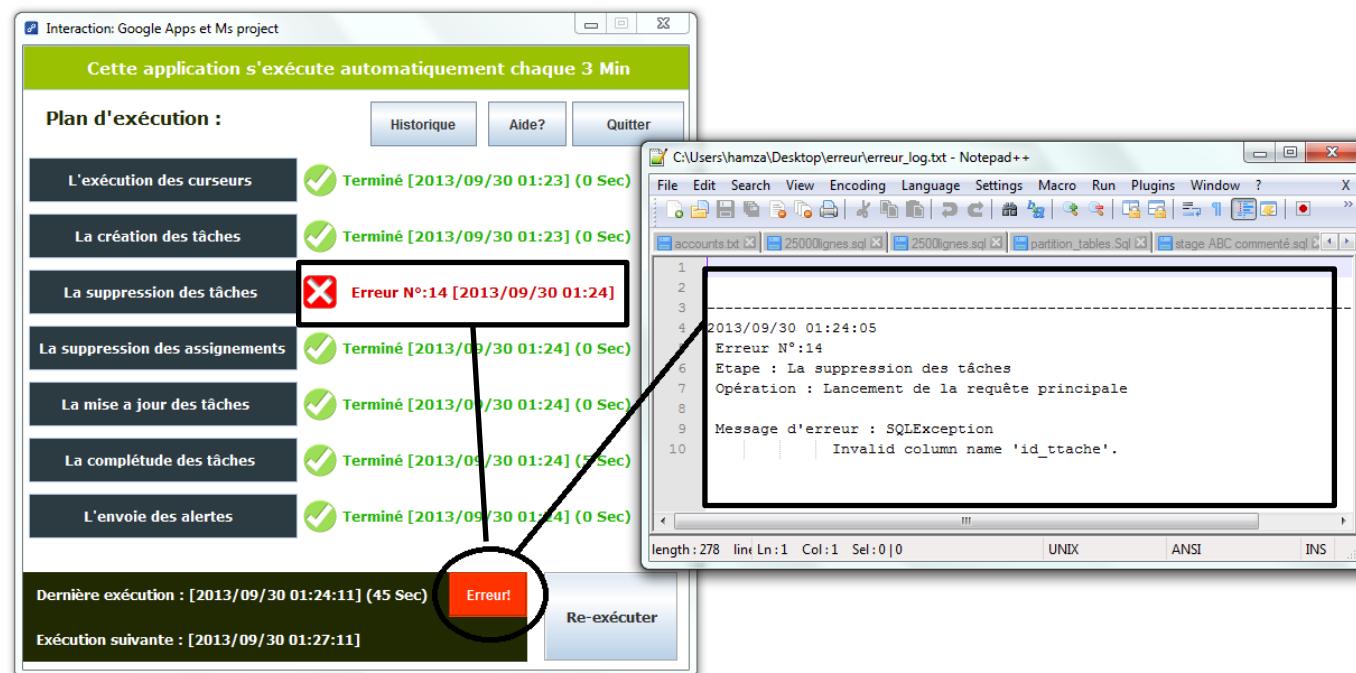
elmetni.hamza@gmail.com  
à moi 00:54 (Il y a :  
  
**Bonjour admin server**  
  
On vous informe qu'il ya un retard d'une tache dans le projet "interaction google apps ms projet"  
les informations du retard :  
  
• Nom du projet : interaction google apps ms projet  
• Titre de la tache : manitenance  
• Assigné à : hamza elmetni ([elmetni.hamza@ensa-tetouan.mygbiz.com](mailto:elmetni.hamza@ensa-tetouan.mygbiz.com))  
• Date de debut du tache : 2013-09-14 08:00:00.0  
• Date de fin du tache : 2013-09-18 17:00:00.0  
• La durée du tache : 32 Heures  
• La description du projet : c'est projet de stage chez ABC tanger

Vous avez reçu ce mail car vous êtes le chef de projet de "interaction google apps ms projet"  
Cette alerte a envoyé le : 2013/09/30 00:54:32

Figure 46 : Exemple d'un message d'alerte

## E) Les interfaces de gestion des erreurs :

Après la fin de chaque exécution, l'application principale s'affiche l'état d'exécution de chaque phase, s'il s'agit d'une erreur bloquant d'une phase il va apparaître en Rouge et en cliquant sur « erreur » elle va afficher le journal des erreurs, par exemple :



Interaction: Google Apps et Ms project  
Cette application s'exécute automatiquement chaque 3 Min

Plan d'exécution : Historique Aide? Quitter

L'exécution des curseurs	<span style="color: green;">Terminé [2013/09/30 01:23] (0 Sec)</span>
La création des tâches	<span style="color: green;">Terminé [2013/09/30 01:23] (0 Sec)</span>
La suppression des tâches	<span style="color: red;">X Erreur N°:14 [2013/09/30 01:24]</span>
La suppression des assignments	<span style="color: green;">Terminé [2013/09/30 01:24] (0 Sec)</span>
La mise a jour des tâches	<span style="color: green;">Terminé [2013/09/30 01:24] (0 Sec)</span>
La complétude des tâches	<span style="color: green;">Terminé [2013/09/30 01:24] (0 Sec)</span>
L'envoie des alertes	<span style="color: green;">Terminé [2013/09/30 01:24] (0 Sec)</span>

Dernière exécution : [2013/09/30 01:24:11] (45 Sec) Erreur! Exécution suivante : [2013/09/30 01:27:11] Re-exécuter

**C:\Users\hamza\Desktop\erreur\erreur\_log.txt - Notepad++**

```
1
2
3
4 2013/09/30 01:24:05
5 Erreur N°:14
6 Etape : La suppression des tâches
7 Opération : Lancement de la requête principale
8
9 Message d'erreur : SQLException
10 | Invalid column name 'id_ttache'.
```

Figure 47: l'interface de gestion des erreurs

S'il s'agit par exemple d'un email invalide , elle va s'afficher en orange :

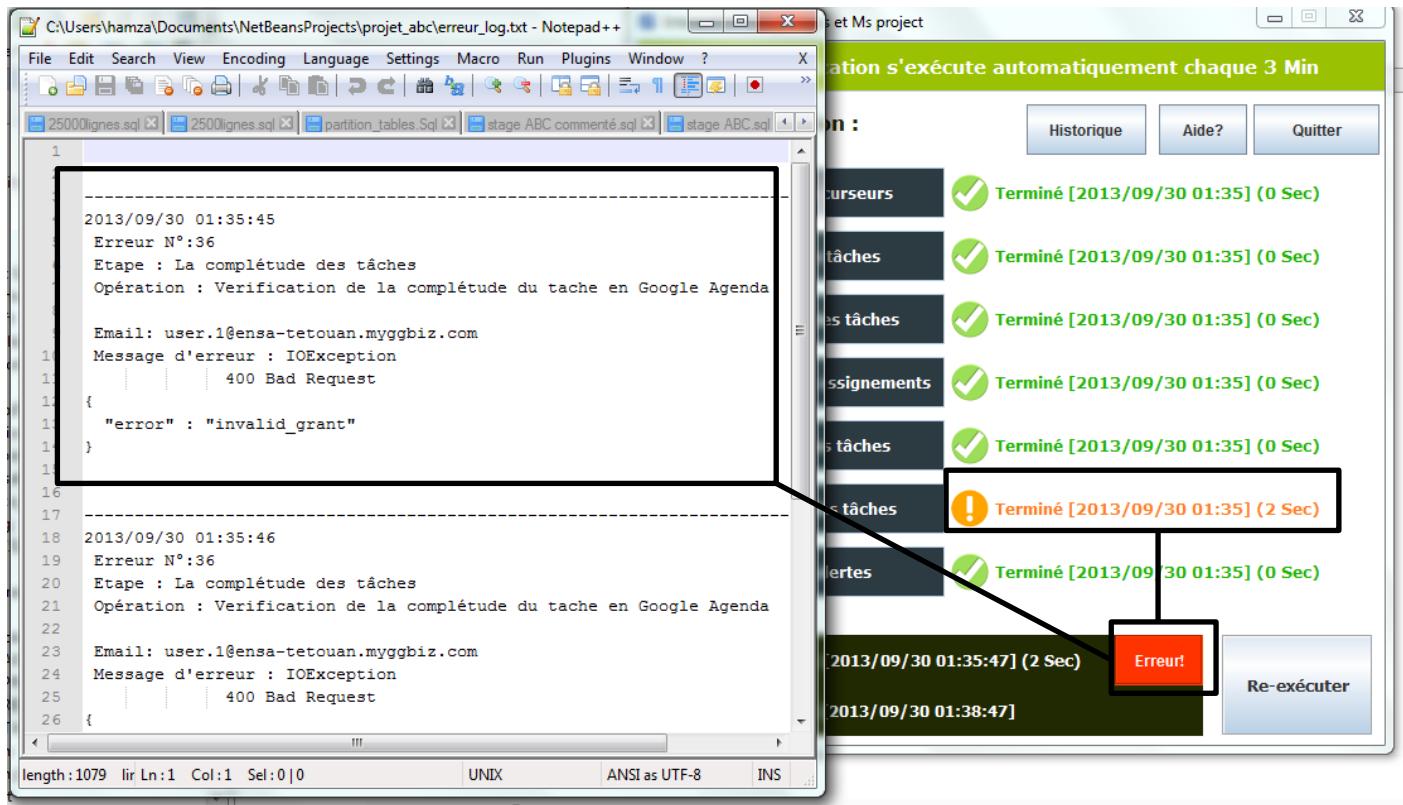


Figure 48 :l'interface de gestion des erreurs 2

## F) L'application des critères ergonomiques :

Les critères ergonomiques constituent une classification (typologie) des propriétés d'une interface homme-machine qui conditionnent son utilisabilité.

L'établissement de critères ergonomiques vise principalement deux objectifs complémentaires :

- Ils permettent d'évaluer l'utilisabilité d'un logiciel (évaluation analytique) en servant de base pour l'établissement de grilles d'évaluation (check-list).
- D'autre part, et c'est encore plus important, ils servent de guide lors de la conception d'une interface utilisateur en aidant les concepteurs à prendre en compte - dès la phase initiale du développement - les aspects ergonomiques du logiciel.

Les critères ergonomiques qu'on a appliqués dans notre projet sont :

- **Compatibilité :**
  - Parler le langage de l'utilisateur (éviter le jargon informatique)
  - Utiliser des métaphores familières
  - L'accès aux fonctions doit être compatible avec la tâche de l'utilisateur
- **Guidage :**
  - Griser les fonctions non disponibles (options de menu, boutons, ...)
  - Fournir la liste des saisies attendues (listes déroulantes, codes à utiliser, ...)
  - indications sur l'opération à effectuer
  - Montrer clairement comment aller en avant et en arrière
  - Regrouper les informations de même type par le format ou par la position (regroupement par menus, bordure, ...)
  - Distinguer par une présentation différente ou un emplacement différent, les informations distinctes pour éviter les risques de confusions
  - Indiquer les modes de fonctionnement du système (état)
  - Rendre visible les traitements réalisés par le logiciel et indiquer clairement lorsqu'ils ont échoué
- **Cohérence:**
  - Utiliser le même schéma d'agencement pour toutes les fenêtres
  - La sémantique des boutons de la souris doit être constante
  - Le même vocabulaire doit être utilisé pour désigner les mêmes fonctions
  - Utiliser une organisation et une syntaxe cohérente pour les menus
  - Utiliser de manière cohérente les symboles graphiques (icônes, couleurs, ...)
- **Gestion des erreurs :**
  - Placer les messages d'erreur là où l'utilisateur est censée regarder
  - Afficher des messages d'erreur explicites (utiliser le langage de l'utilisateur)
  - Eviter les textes trop longs (rester bref, utiliser des liens, des références, ...)
  - Eviter les textes réprobateurs

On conclusion on peut dire qu'on a réussi à appliquer les plus parts des critères ergonomiques



## 6- Optimisation de base de données

Afin d'augmenter la performance de notre solution on a utilisé des techniques d'optimisation de base de données, dans cette partie on va parler de ces techniques puis leurs impacts sur la performance.

## A)Les techniques d'optimisation de base de données :

La choix de technique d'optimisation est dépend forcement de la nature des champs et des attributs dans une table, on a appliqué 3 technique .

### ➤ **Les index Bitmap :**

Un index de base de données de type bitmap fonctionne selon le principe de la cardinalité. En effet, chaque valeur ou plage en binaire d'un champ donné permet de se repérer dans l'index, afin d'identifier tous les n-uplets correspondant à ces valeurs.

On a utilisé ces index pour tous les champs de « UID » dans notre base de données pour accélérer la recherche.

Exemple : CREATE NONCLUSTERED INDEX index\_taches ON taches(id\_tache\_ms\_project);

### ➤ **Les partitions :**

Dans une base de données, une partition est une division logique d'une table stockée en plusieurs parties indépendantes. Le partitionnement de tables est généralement effectué pour améliorer la gestion, la performance ou la disponibilité.

On a appliqué des partitions horizontale sure les champs :

operation\_tache[operation\_tache] , operation\_tache\_person[operations]

### ➤ **L'optimisation des requêtes :**

On a essayé aussi d'appliquer tous les forme normale de base de données afin d'optimiser notre MCD et éviter les redondances.

Et aussi au niveau des jointures on a essayé de minimiser ses utilisations.

Finalement on a essayé d'optimiser nos requêtes SQL par exemple : remplacer « NOT IN » dans les requêtes par « NOT EXIST »

## B)Le résultat d'optimisation :

Le figure suivante présente une comparaison entre 2 cas d'exécution pour le curseur « insertion des taches » en appliquant les techniques d'optimisation, et sans l'application de ces techniques.

Ils sont les résultats de création de 20 000 taches dans la base de données de la solution qui contient 5000 taches.

On a générée ces statiques avec un outil de SQL server 2008

	<b>Sans optimisation</b>	<b>Avec optimisation</b>
Query Profile Statistics		
Number of INSERT, DELETE and UPDATE statements	42687	17662
Rows affected by INSERT, DELETE, or UPDATE statem...	46892	17662
Number of SELECT statements	120342	100340
Rows returned by SELECT statements	108338	90338
Number of transactions	42687	17662
Network Statistics		
Number of server roundtrips	3	1
TDS packets sent from client	998	4
TDS packets received from server	750	568
Bytes sent from client	4082568	14246
Bytes received from server	3065875	2323016
Time Statistics		
Client processing time	26162	12850
Total execution time	29104	13072
Wait time on server replies	2942	222

Figure 49 : statiques sur l'optimisation de la base de données

On remarque au niveau des statiques sur les requêtes (Query profile statistics ) en appliquant des techniques d'optimisation des base de données on a pu diminuer le nombre de déclarations et aussi le nombre des lignes traitées , et le nombre des transaction .

et cela va nous faire gagner du temps et des ressources.

Au niveau de statistique sur le réseau on remarque que l'optimisation fasse un travail incroyable, pour TDS paquets envoyées par le clients ( TDS est un protocole de couche d'application, qui sert à transférer des données entre un serveur de base de données et un client) a diminué de presque 1000 jusqu'à seulement 4 .

Et le plus important c'est les bytes envoyé par le client de plus de 4 millions jusqu'à seulement 14000.

Pour le temps d'exécution de presque 30 seconds jusqu'à seulement 13 second ( plus de 60% ) .

En conclusion on peut dire que les techniques d'optimisation de la base de données qu'on a appliquée ont parfaitement fonctionné dans notre projet

# Conclusion

Tout d'abord je dois dire que ce stage fut une réussite. Les enseignements que j'en ai tiré me serviront inéluctablement dans mon futur emploi, où la face sociale est aussi (si ce n'est plus) importante que le côté technique. De nos jours le rôle d'un ingénieur est de gérer son équipe, de faire du management. Or pour pourvoir diriger, guider les ouvriers vers les bons objectifs, il est nécessaire d'être passé de l'autre côté de la barrière. Ce stage m'a montré que la gestion d'une équipe d'ouvriers est possible dans une bonne entente, ou tout au moins par un intéressement aux résultats de l'entreprise.

Je suis conscient que l'entreprise dans laquelle j'ai effectué mon stage avait une santé financière excellente, ce qui lui permet de rendre la vie de ses ouvriers moins pénible.

Les ingénieurs s'accordent sur un point au niveau de la supervision des ouvriers ; pour se faire prendre au sérieux, il est nécessaire de parler, d'expliquer, de montrer les tâches à accomplir. Or cela ne se fait que par le biais de l'apprentissage de la base, puis des notions plus abstraites et immatérielles.





# Listes des figures

Figure 1 : Logo de Diana Holding .....	11
Figure 2 : Logo d'Atlas Bottling Company .....	11
Figure 3 : Une photo de l'extérieur de l'usine d'ABC tanger (2013).....	12
Figure 4 : Répartition géographique du marché des boissons au maroc .....	13
Figure 5 : la structure du marché national du boissons .....	14
Figure 6 : l'Evolution du chiffre d'affaires d'ABC .....	14
Figure 7 : l'organigramme d'ABC.....	15
Figure 8 : logo de Microsoft projet 2010 (application de bureau).....	16
Figure 9 : logo de Microsoft projet server 2010 .....	17
Figure 10 : Le fonctionnement de Microsoft project server 2010.....	17
Figure 11 : une capture d'écran du Microsoft project server 2010 .....	19
Figure 12 : Logo de Google Apps .....	19
Figure 13 : une capture d'écran du Google Calendar .....	20
Figure 14 : page d'accueil de Google APis Console .....	21
Figure 15 : Logo du Google Apis Console.....	21
Figure 16 : Exemple d'un rapport de Google Apis console .....	22
Figure 17 : Moéle en cascade .....	23
Figure 18: un schéma represente L'utilité de projet .....	33
Figure 19:Structure de l'équipe de projet .....	34
Figure 20 : le diagramme des cas d'utilisation de projet(vue global).....	36
Figure 21:La forme de première solution possible .....	37
Figure 22: La forme de la deuxième solution possible.....	38
Figure 23:La forme de la troisième solution possible .....	39
Figure 24: La forme générale de la solution .....	41
Figure 25 : la planification réelle de projet.....	43
Figure 26 : Le diagramme de contexte de la solution .....	47
Figure 27 : le diagramme des cas d'utilisation de projet(vue solution).....	48
Figure 28 : Le diagramme de composant de la solution.....	50
Figure 29 : La vue physique de la solution.....	52
Figure 30 : Le MCD de la BD de MS project server .....	54
Figure 31 : Le MLD de la BD de MS project server.....	55
Figure 32 : Le MCD de la solution .....	59
Figure 33: Le MLD de la solution .....	60
Figure 34 : Diagramme de séquence de la solution avec l'authentification 3-legged OAuth .....	70
Figure 35: un exemple de page de demande d'accès.....	71
Figure 36 : Diagramme de séquence de la solution avec l'authentification 3-legged OAuth .....	72
Figure 37 : un exemple d'un fichier « config.xml » généré par la solution .....	74
Figure 38 : Exemple d'un message d'erreur généré par la solution .....	75

<i>Figure 39 : exemple du journal des erreurs .....</i>	75
<i>Figure 40 : La fenêtre d'application de configuration .....</i>	76
<i>Figure 41: le message du réussite de la génération de "config.xml" .....</i>	77
<i>Figure 42: la fenêtre principale d'application.....</i>	77
<i>Figure 43 : minimisation d'application dans la base des taches .....</i>	78
<i>Figure 44:La fenêtre d'historique .....</i>	79
<i>Figure 45 : la fenêtre de sauvegarde d'historique .....</i>	79
<i>Figure 46 : Exemple d'un message d'alerte.....</i>	80
<i>Figure 47: l'interface de gestion des erreurs 1 .....</i>	80
<i>Figure 48 :l'interface de gestion des erreurs 2 .....</i>	81
<i>Figure 49 : statiques sur l'optimisation de la base de données.....</i>	82



# Listes des tableaux

<i>Tableau 1 : Fiche Signalétique d'ABC.....</i>	15
<i>Tableau 2: Equipe de projet.....</i>	27
<i>Tableau 3: Milestone de projet.....</i>	30
<i>Tableau 4:les livrables de projet .....</i>	31
<i>Tableau 5:les risques de projet .....</i>	32
<i>Tableau 6:les contraintes de projet .....</i>	33
<i>Tableau 7:Rôles et responsabilités d'équipe de projet .....</i>	34
<i>Tableau 8:Les installations et les ressources de projet.....</i>	35
<i>Tableau 9 : Les technologies utilisées dans le projet .....</i>	42
<i>Tableau 10 : Le dictionnaire de données de Ms projet server .....</i>	54
<i>Tableau 11: Le dictionnaire des données de la table Person.....</i>	55
<i>Tableau 12 : Le dictionnaire des données de la table chef_projet .....</i>	56
<i>Tableau 13 : Le dictionnaire des données de la table projet .....</i>	56
<i>Tableau 14 : Le dictionnaire des données de la table tache_person .....</i>	56
<i>Tableau 15: Le dictionnaire des données de la table taches .....</i>	57
<i>Tableau 16: Le dictionnaire des données de la table operation_tache_person .....</i>	57
<i>Tableau 17 : : Le dictionnaire des données de la table operation_tache .....</i>	58
<i>Tableau 18 : Le dictionnaire des données de la table historique_taches .....</i>	58