Table des métiers:

[Introduction générale: 1](#_Toc325125721)

[I. Cadre du projet 4](#_Toc325125721)

[Introduction 4](#_Toc325125721)

1. L'organisme d'accueil Pro-id : [4](#_Toc325125721)
2. Les acteurs principaux : [5](#_Toc325125721)

1. [Les petites et moyennes entreprises(PME) 5](#_Toc325125721)

2. [Les très petites entreprises(TPE) 5](#_Toc325125721)

1. [Les collectivités locales 5](#_Toc325125721)

[Conclusion 6](#_Toc325125721)

II[. Phase d'incubation 7](#_Toc325125721)

III[. Phase d’élaboration 29](#_Toc325125721)

IV[. Phase de consctruction 44](#_Toc325125721)

V[. Phase de transition 56](#_Toc325125721)

**INTRODUCTION GENERALE**

L’outil informatique, dans le cadre de la mondialisation économique, est devenu essentiel dans tous les secteurs, en vue d’accroitre le rendement du travail et d'améliorer la qualité des différentes tâches réalisées. Ainsi, l’allégement du travail administratif répétitif, le gain de temps considérable et réduction des coûts ne sont plus un luxe mais une nécessité permettant la bonne gestion des entreprises d’une part, et une meilleure réactivité vis-à-vis du monde extérieur d’autre part, surtout dans une ère où la mondialisation favorise la concurrence.

Avec l’essor d’Internet, maintes entreprises font évoluer leur système d’information. On assiste à une émergence d’applications Internet dynamiques, qui répondent spécifiquement aux besoins différents des entreprises et qui rendent la consultation et l'élaboration de certaines tâches plus attrayantes pour les utilisateurs, notamment la gestion des appels d’offres (AO).

Une plateforme de gestion des appels d’offre en tant qu'outil informatique doit faire l'objet d'une attention particulière pour pouvoir garantir un bon traitement et une bonne transmission des informations relatives aux différents appels d’offre, d'où le besoin d’en assurer l'intégrité, celle des processus qui y sont associés, et d’en maintenir la disponibilité.

De ce fait, une analyse conceptuelle a été réalisée, et focalisée sur l’approche à suivre, qui doit être homogène avec les nécessités primordiales pour aboutir aux fonctionnalités accessibles dans l’application. C’est pour cela, qu’on avait opté pour l’élaboration d’une plateforme Internet, qui repose sur l’approche orienté objet ; le ‘processus unifié’ et un SGBD de développement MySQL, dans l’intention de matérialiser nos acquis théoriques et enrichir notre formation.

Ce rapport est composé de 5 chapitres au sein desquels nous exposeront progressivement les fonctionnalités de l’application à réaliser. Dans le 1er chapitre intitulé ‘Cadre du rapport’, nous décrivons le cadre du projet ainsi que les spécificités de l’entreprise **PRO-ID**.

Au niveau du 2ème chapitre intitulé ‘Phase d’incubation’, nous décèlerons le contexte de fonctionnement de l’application et les cas d’utilisation tout en ordonnant leurs priorités dans notre travail et en traitant les principaux.

Dans le 3ème chapitre nommé ‘Phase d’élaboration’, nous explorerons l’application, en vue d’aboutir à une analyse, une conception détaillée et une implémentation des différents cas d’utilisation en dernier lieu.

Dans le 4ème qui portera le nom de ‘Phase de construction’, nous terminerons la conception et l’implémentation des derniers cas d’utilisations restants et une version de l’application sera prête pour le test.

Dans le 5ème et dernier chapitre intitulé ‘Phase de transition’, nous procéderons à l’exécution de notre application.

Nous terminerons enfin notre rapport, par une conclusion qui résume le contenu de notre rapport et des perspectives futures de notre application.

CHAPITRE I :

Cadre du projet

**Introduction:**

Afin de s'adapter aux évolutions ainsi que les nouvelles technologies, la modernisation et l'automatisation du secteur économique notamment le domaine des appels d'offre deviennent une nécessité.

Dans ce contexte, la société Pro-id propose comme amélioration une plateforme qui répond aux besoins de communication de tous les acteurs du secteur économique.

1. **L'organisme d'accueil Pro-id :**

Pro-id est une Société Anonyme à Responsabilité Limitée(SuARL) spécialisée dans la réalisation des projets de web vitrine, de site de commerce ou encore les sites à caractère événementiel.

Sa vocation première étant de générer de la valeur ajoutée à ses clients, "Pro-id" a mis en place une stratégie qui repose sur 5 activités principales à savoir :

* **Création des sites Internet :** Ce service est destiné aux clients souhaitant la création ou la refonte de leur image électronique.
* **Hébergement des sites web et achat de noms de domaines :** Ce service offert par "PRO-ID" à titre gracieux, afin de consolider l’identité électronique et mettre par la suite, le nom de marque à l’abri de ses concurrents.
* **Référencement et Optimisation dans les différents outils de recherche (moteurs de recherche, annuaires,…) :** "PRO-ID" offre une panoplie de moyens techniques de référencement et d’optimisation, permettant ainsi de voir le site concerné site bénéficier des meilleures chances d’être repéré par les prospects lors de l’utilisation des moteurs et/ou des annuaires de recherche.
* **Mise en place des stratégies de E-commerce :** "Pro-id" met à disposition un large éventail de solutions de e-commerce, afin d’appuyer la stratégie marketing en ligne et générer par la suite du trafic sur un site Internet ex : opérations d‘e-mailing, bannières publicitaires, catalogues en ligne, payement électronique,…
* **Développement de différentes applications web :** Ce service est destiné aux clients désireux d’intégrer dans leurs sites Internet, des applications web spécifiques selon leurs besoins. Intranet, logiciels de programmation...

1. **les acteurs principaux:**
2. **Les petites et moyennes entreprises(PME)**:

Les PME comportent les entreprises ayant de 20 à 249 salariés travailleurs. Elles ont généralement des difficultés de passation ou d'acquisition des appels d'AO. Le manque de communication entre les différentes parties du secteur économique.

1. **Les très petites entreprises(TPE):**

Les TPE Rencontrent les mêmes problèmes que les PME sauf que ceux-ci sont plus importants d'autant plus que les TPE ont un nombre de salariés moindre allant de 1 à 19.

1. **les collectivités locales:**

Ce sont les municipalités, les gouvernorats, les délégations et les Imadas, les administrations et les entreprises publiques (comme La Poste Tunisienne ainsi que le ministère de l'industrie)...

Les collectivités les plus visées sont celles isolées et ayant une gestion des appels d'offre faible.

**Conclusion:**

Au cours de ce chapitre nous avons présenté le cadre général de notre projet. Dans le chapitre qui suit nous allons approfondir les connaissances et délimiter les contours.

CHAPITRE II :   
Phase d’incubation

**Introduction :**

Cette étape du rapport, constituant la 1ère étape du cycle de vie du processus unifié, est très importante au niveau de l’élaboration de notre projet puisqu’elle va nous permettre de détecter le fonctionnement du système actuel existant et d’en déterminer les limites. En outre l’identification des acteurs concernés, ainsi que tous leurs besoins et actions accomplies qui devraient être plus facile à traiter à travers l’application via la spécification des fonctionnalités pertinentes du projet, les utilisateurs, l’architecture adoptée du système ainsi que les contraintes majeures.

I. Objectifs :

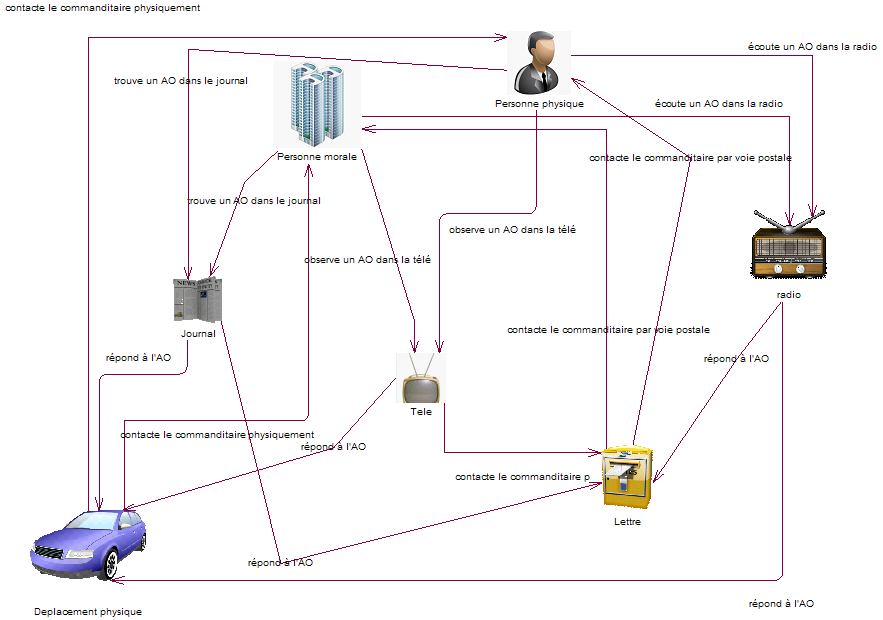
Le projet consiste à concevoir et à développer une plateforme web de gestion des appels d’offres qui répond aux exigences suivantes :

* Informatiser le processus de gestion des AO.
* Faciliter le traitement des AO.
* Gagner du temps, notamment en simplifiant les procédures pour le soumissionnaire de répondre aux différents appels d’offres disponibles sur le portail.
* Assurer l'intégrité des AO traités et la transparence des procédures.
* Consulter instantanée de chaque AO, à tout moment et en tout lieu, dès lors que l’on dispose d’une connexion internet.
* Ce qui nous amène à faire une étude de l’existant, tant au niveau de l’état actuel et de sa critique, que celui de la solution proposée

L’étude de l’existant permet de détecter les insuffisances d’un domaine et de proposer des solutions à retenir. Nous commencerons cette étude par une description de l’état actuel.

**II.1 Etat actuel :**

Le système actuel de gestion des AO, comporte plusieurs obstacles pour que leur traitement soit efficient, puisque les opérations sont effectuées manuellement et de façon non normée, le schéma suivant donne une vue clair de l’ancien système:



**Figure II.1 :** *Ancien Système*

* Ainsi, nous remarquons la complexité des procédures de recherche et de réponse aux AO. Nous pouvons constater que cette complexité cause une perte d’efficience dont l’amélioration serait adéquate.

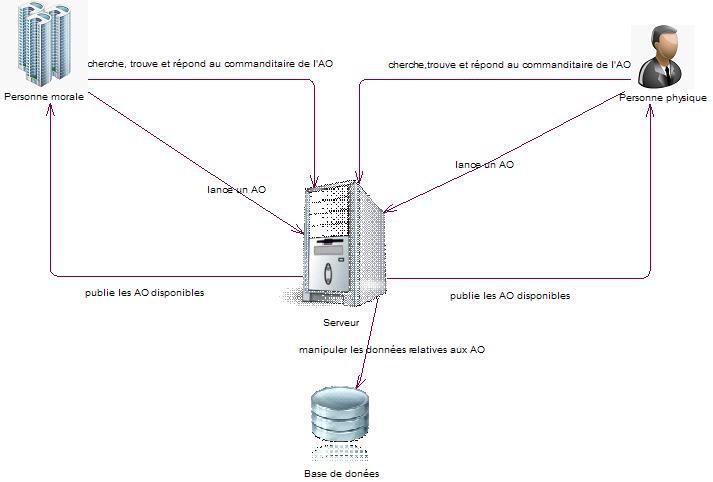
**II.2 Critique de l’existant :**

Au cours de cette étude, nous avons constaté un certain nombre de défaillances qui sont à l’origine de la gestion manuelle des différentes activités. En effet, nous avons remarqué :

* La perte de temps lors de la recherche des données datant parfois de plusieurs mois voire plus.
* Travail manuel et bureaucratique répétitif.
* La difficulté de la mise à jour des informations circulant au niveau du centre de gestion.
* La dispersion des AO et l’absence de contacts entre les soumissionnaires et les commanditaires.
* Le risque de perte des documents dont la trace a été perdue ce qui entraîne l’incohérence des données et le gâchis d’opportunité commerciales.
* La redondance des informations au niveau des fiches qui sont refaites plusieurs fois pour en garder la trace. D’où le gaspillage de papier qui peut avoir lieu.
* L’incapacité de l’archive pour garder tous les dossiers par manque de matériel de personnel et de local.
* Difficulté de stockage des données qui nécessitent un espace important et des frais de maintenance et de classement
* Perte de temps à comparer vue la lenteur et l’inadéquation des procédures suivies.
* Compte tenu des problèmes cités ci-dessus, ceci nous amène à avoir les exigences fonctionnelles suivantes.

**II.3 Solution proposée**

Afin de remédier aux insuffisances de l’ancien système, l’informatisation du bureau d’ordre s’impose :



**Figure II.2 :** *Solution à mettre en place*

Ainsi la solution proposée permettra de :

* Informatiser et centraliser des données.
* Mettre à jour des différentes tables des utilisateurs ou celle des AO au fur et à mesure des nécessités.
* Assurer la procédure qui permet la réalisation de la consultation et l’émission d’AO sur la plateforme.
* L’identification des utilisateurs, Pour bien assurer la sécurité des données personnelles (un login et un mot de passe).

1. **Exigences fonctionnelles :**

Les principales exigences fonctionnelles sont des règles de gestion à suivre sont :

* L’inscription des différents utilisateurs moyennant un abonnement renouvelable.
* L’identification parfaite des acteurs inscrits via un identifiant unique, et éviter surtout les doubles codifications.
* La mise à jour régulière et continue pour un échange des informations avec le Serveur Informatique de la plateforme après une éventuelle rectification de la validité de chaque opération établie.
* Sécurité d’accès **:** Le système devra assurer la sécurité et la confidentialité des données et des traitements, tout en respectant les privilèges accordés à chaque utilisateur du système.
* Sécurité des données : Le transfert des données entre les acteurs et au sein du système informatique doit être sécurisé, et les données échangées ne doivent dans aucun cas être modifiables par un acteur qui n’aurait pas eu les droits d’effectuer de telles opérations. Par ailleurs, les acteurs qui ont le plus faible privilège d’accès n’a la possibilité d’accès qu’en consultation et émissions aux données relatives à leurs profils.

1. **Exigences non fonctionnelles**

Dans ce cadre nous nous intéressons à l’interface et l’ergonomie de l’application.

En fait, il faut assurer :

* La convivialité de l’interface graphique et son ergonomie.
* La possibilité de revenir au menu principal de l’application à partir de n’importe quelle fenêtre de celle-ci.
* Les composants graphiques et les boutons devront être munis d’icônes qui indiquent la fonction afin qu’elle soit facilement identifiée.
* La rapidité de traitement : En effet, vu le nombre important des sessions des acteurs, il est impérativement nécessaire que la durée de chargement de l’interface soit dans les délais les plus courts.

1. **Identification des acteurs et des cas d’utilisations**

**V.1. Les acteurs**

Nous pourrons discerner parmi les acteurs deux types :

* Les utilisateurs identifiés comme administrateurs.
* Les utilisateurs identifiés comme clients (soit soumissionnaire ou commanditaire).

Les utilisateurs classés en tant qu’Administrateur et présentant un statut d’informaticien au sein de la PRO-ID, sont responsables de la structure de gestion de l’application, le suivi, la mise à jour des données, la gestion de droit d’accès et la gestion des utilisateurs.

Les agents répertoriés en tant que Client et présentant un statut administratif (entreprise \ organisme public \ personne physique) au sein d’PRO-ID sont invités à gérer leurs AO de façon informatisée, et bénéficient d’un espace de stockage personnel d’une diversité de données.

**V.2 Les fonctionnalités**

Les fonctionnalités attendues par le système et saisissant un besoin fonctionnel sont décrites à leurs utilisateurs comme suit, concernant les clients, ils auront à:

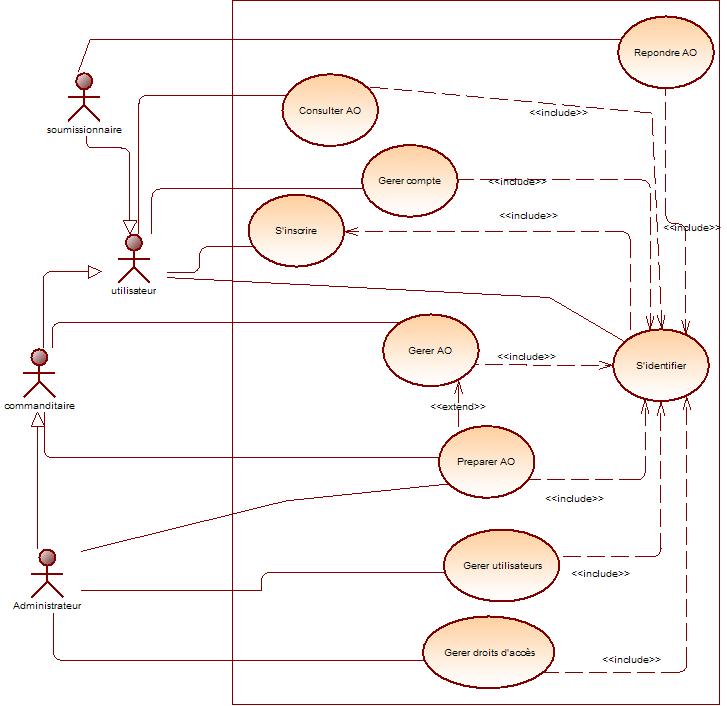
* **S’inscrire**: chaque utilisateur devra s’inscrire au préalable en fournissant ses caractéristiques professionnelles et en payant son abonnement s’il s’agit d’un compte client (commanditaire/soumissionnaire).
* **S’identifier** : chaque client accédant à l’application doit être impérativement identifié, et cela s’effectue à travers le nom de l’utilisateur (login) et le mot de passe ce qui permettra d’avoir en quelque sorte un système sécurisé contre les intrusions d’autrui et d’avoir un cadre de sérénité vis-à-vis de l’application.
* **Gérer AO :** chaque client (commanditaire) ayant l’opportunité de gérer son AO que ce soit en préparant, en envoyant, en retirant ou en modifiant des AO.
* **Gérer Compte :** Permet aux clients ayant accès à l’application de modifier des informations, de payer éventuellement l’abonnement, d’altérer les données stockées au niveau de la session
* **Répondre AO :** Autoriser l’ajout, la modification et la suppression des contacts de l’annuaire de l’agent
* **Consulter AO**: Chaque utilisateur devra pouvoir accéder au journal des offres et/ ou aux archives.
* **Préparer AO**: En amont de la gestion des AO émis, le commanditaire aura la possibilité de saisir et d’enregistrer des formulaires qui seront prés à être lancés sous forme d’appels d’offre. Le commanditaire pourra éventuellement, supprimer ou modifier les formulaires cités ci-dessus.

En ce qui concerne, en plus les administrateurs :

* **Gérer utilisateurs**: L’administrateur pourra éventuellement ajouter, supprimer et/ou modifier des paramètres propres au client concerné.
* **Gérer droits d’accès :** De même que sus-dit, l’administrateur a la possibilité d’octroyer, de révoquer et de visualiser les droits d’accès à un quelconque client.

**V.3 Modèle initial du cas d’utilisation :**

Il s’agit du modèle initial du diagramme des cas d’utilisation général non raffiné, dont les principales fonctionnalités ont été citées ci-dessus.



**Figure II.3 :** Diagrammes des cas d’utilisation initial

* **Affectation des priorités aux cas d’utilisations :**

Nous affectons la priorité la plus forte aux cas d’utilisations qui représentent les principales fonctionnalités du système donnant une vision globale de l’ensemble, ainsi qu’une priorité faible à ceux qui vont être détaillés ultérieurement.

* **L’acteur « utilisateur » :**

L’acteur « utilisateur », dans notre travail, peut être soit client, soit administrateur et nullement autre chose.

* **Le « Client » :**

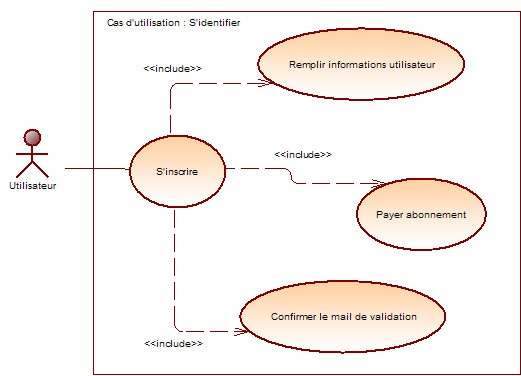
Le client est à la fois Soumissionnaire et commanditaire. Il a ces deux rôles qu’il inter-change au grès des besoins auxquels il fait face. Ainsi un même client peut répondre à un AO et en préparer un autre.

**Tableau II.1 :** Affectation des priorités aux cas d’utilisations

|  |  |  |  |
| --- | --- | --- | --- |
| N° | Cas d’utilisation | Acteurs | Priorité |
| 1 | S’inscrire | Commanditaire/Soumissionnaire/Administrateur | 1 |
| 2 | S’identifier | Commanditaire/Soumissionnaire/Administrateur | 1 |
| 3 | Préparer AO | Commanditaire/Administrateur | 1 |
| 4 | Répondre AO | Soumissionnaire/Administrateur | 1 |
| 5 | Gérer AO | Commanditaire/Administrateur | 1 |
| 6 | Gérer Compte | Commanditaire/Soumissionnaire/Administrateur | 2 |
| 7 | Gérer droits d’accès | Administrateur | 2 |
| 8 | Gérer utilisateurs | Administrateur | 2 |
| 9 | Consulter AO | Administrateur | 2 |

**VI Raffinement des cas d’utilisation :**

**VI.1 Raffinement du cas d’utilisation « S’inscrire »**

****

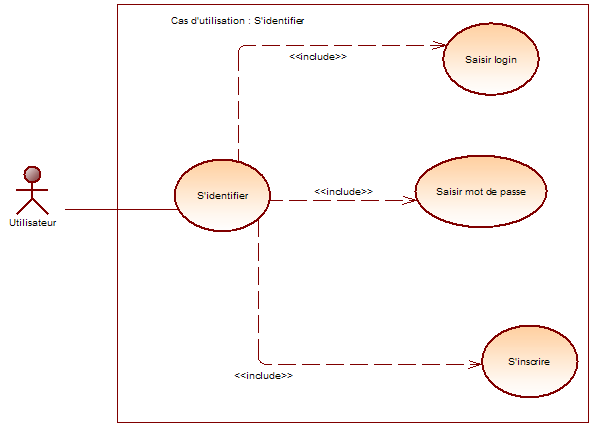
**Figure II.4 :** Structuration du cas d’utilisation raffiné « S’inscrire »

L’utilisateur doit être inscrit sur la plateforme pour accéder aux autres fonctionnalités du système. Tout en remplissant les informations concernant l’utilisateur, et en payant l’abonnement et en confirmant le mail de validation.

**Tableau II.2 :** Description du cas d’utilisation « S’inscrire »

|  |  |
| --- | --- |
| Cas d’utilisation | S’inscrire |
| Acteurs | Commanditaire/soumissionnaire/Administrateur |
| Pré-condition | L’utilisateur n’est pas inscrit sur le site |
| Post-condition | L’utilisateur est inscrit et peut accéder aux tâches qui lui sont permises selon son type |
| Scénario | 1. L’utilisateur remplit des formulaires relatifs à divers types d’information le concernant. 2. Il passe à l’étape du payement des frais d’inscription pour son abonnement. 3. Il confirme le code de validation |
| Exception | L’utilisateur fait des erreurs de saisie se répétant plusieurs fois. il choisit d’interrompre l’inscription. |

**VI.2 Raffinement du cas d’utilisation « S’identifier »**

****

**Figure II.5 :** structuration du cas d’utilisation raffiné « s’identifier »

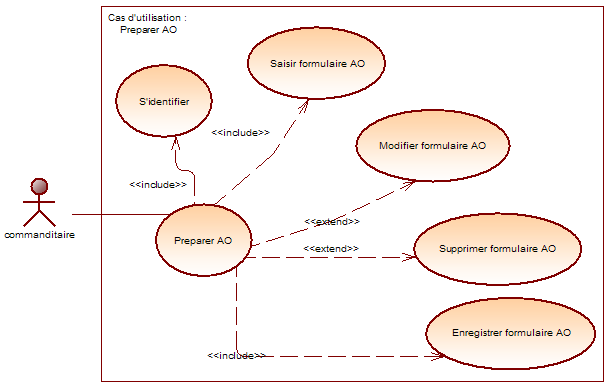
L’utilisateur doit être capable de s’identifier.

**TableauII.3 :** Cas d’utilisation : S’identifier

|  |  |
| --- | --- |
| Cas d’utilisation | S’identifier |
| Acteurs | Commanditaire/soumissionnaire/Administrateur |
| Pré-condition | L’utilisateur ne s’est pas identifier pour accéder à la plateforme l’utilisateur est d’ores et déjà inscrit |
| Post-condition | L’utilisateur est identifié et peut accéder aux tâches qui lui sont permises selon son type |
| Scénario | 1. L’utilisateur saisi son login et son mot de passe. 2. L’utilisateur valide son choix. |
| Exception | L’utilisateur fait des erreurs de saisie des identifiants se répétant plusieurs fois il choisit d’interrompre l’identification. |

**VI.3 Raffinement du cas d’utilisation «préparer AO »**

**Figure II.6 :** structuration du cas d’utilisation raffiné « préparer AO »

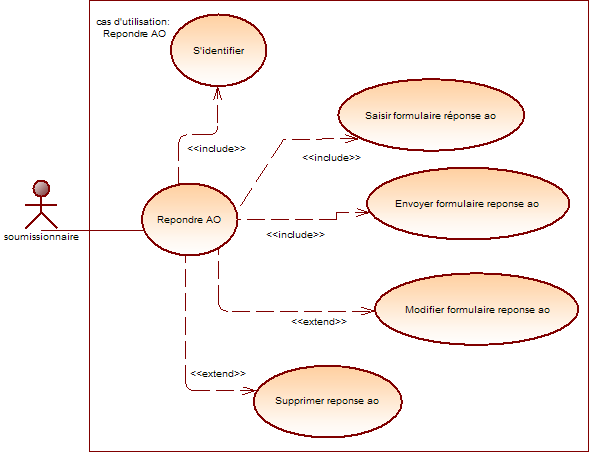


Le commanditaire et l’administrateur doivent saisir, enregistrer des formulaires d’AO pour préparer un AO, et doivent pouvoir en supprimer et en enregistrer afin de préparer un AO.

**Tableau II.4 :** Cas d’utilisation : Préparer AO

|  |  |
| --- | --- |
| Cas d’utilisation | Préparer AO |
| Acteurs | Client (commanditaire) |
| Pré-condition | Le client se doit de s’être identifié au préalable |
| Post-condition | L’AO est préparé |
| Scénario | 1. Le formulaire AO doit être saisi et enregistré 2. Par extension il pourrait éventuellement modifier le formulaire AO pour que la préparation soit valide. 3. Relevant du CRUD, nous traitons ici de la suppression :   Accédant au menu de préparation, l’utilisateur est en état de supprimer chacun des AO lancé sur la plateforme. |
| Exception | L’administrateur n’exécute pas de requête spécifique ou il décide d’interrompre ses opérations |

**VI.4 Raffinement du cas d’utilisation « répondre AO»**

****

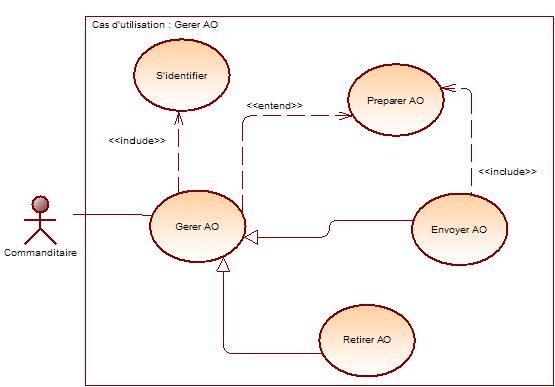
**Figure II.7 :** structuration du cas d’utilisation raffiné « Répondre AO »

Le soumissionnaire doit pouvoir répondre à un AO en saisissant et en envoyant un formulaire réponse AO, il pourra par extension, modifier et supprimer des réponses aux AO.

**Tableau II.5 :** Cas d’utilisation : Répondre AO

|  |  |
| --- | --- |
| Cas d’utilisation | Répondre AO |
| Acteurs | Soumissionnaire |
| Pré-condition | Soumissionnaire se doit de s’être identifié au préalable |
| Post-condition | La réponse à l’AO a été effectuée |
| Scénario | 1. Le soumissionnaire saisit un formulaire de réponse à l’AO. 2. Il l’envoie comme réponse. 3. *In extenso,* il pourra éventuellement le supprimer ou le modifier |
| Exception | Le soumissionnaire choisi de quitter le menu réponse aux AO ou il induit des erreurs répétitives dans la saisie ou la modification des formulaires |

**VI.5 Raffinement du cas d’utilisation «Gérer AO »**

****

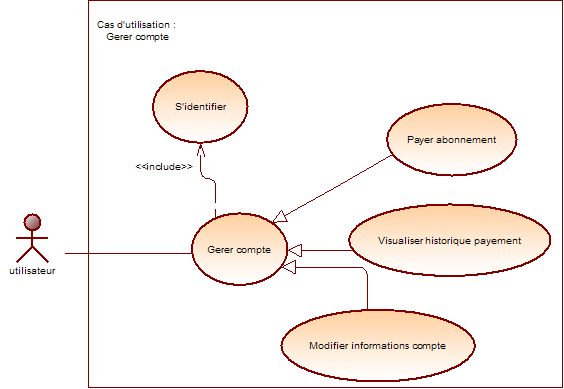
**Figure II.8 :** structuration du cas d’utilisation raffiné « gérer AO »

Le commanditaire et l’administrateur doivent être capables de gérer les AO, notamment en préparant, en renvoyant et en retirant des AO.

**Tableau II.6 :** Cas d’utilisation : Gérer AO

|  |  |
| --- | --- |
| Cas d’utilisation | Gérer AO |
| Acteurs | Commanditaire/Administrateur |
| Pré-condition | Le commanditaire s’est identifié |
| Post-condition | Les différents AO lancé sont géré sont gérés |
| Scénario | Dans ce cas, l’utilisateur |
| Exception | Le commanditaire faite une saisie erronée ou cesse ces opérations |

**VI.6 Raffinement du cas d’utilisation « gérer compte »**

****

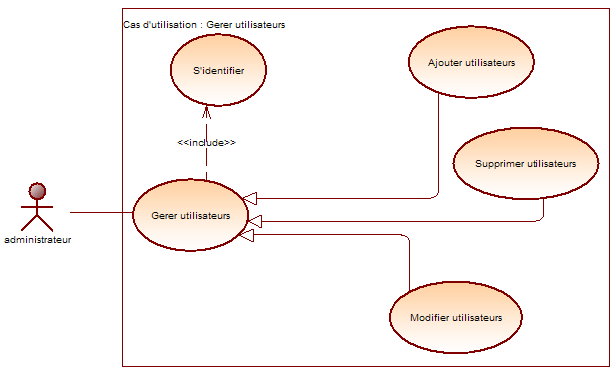
**Figure II.9 :** structuration du cas d’utilisation raffiné « gérer compte »

L’utilisateur doit être capable de gérer son compte. Notamment en payant son abonnement, en visualisant l’historique de ses payements et en modifiant les informations qui lui sont propres.

**Tableau II.7 :** Cas d’utilisation : Gérer compte

|  |  |
| --- | --- |
| Cas d’utilisation | Gérer compte |
| Acteurs | Administrateur/Soumissionnaire/Commanditaire |
| Pré-condition | L’utilisateur se doit d’être identifié au préalable |
| Post-condition | Le payement de l’abonnement est effectué |
| Scénario | La modification des informations relevant du CRUD, nous ne traitons ici que du payement des abonnements :  L’utilisateur paye les frais d’abonnement annuel ou semestriel. |
| Exception | L’utilisateur décide d’interrompre une tâche ou il répète une opération erronée plusieurs fois |

**VI.7 Raffinement du cas d’utilisation « gérer utilisateurs »**

****

**Figure II.10 :** structuration du cas d’utilisation raffiné « Gérer utilisateurs »

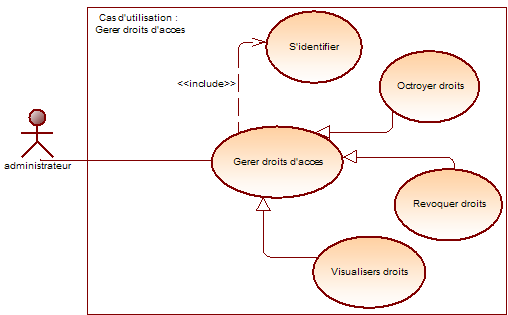
L’administrateur doit être capable d’ajouter, de supprimer des utilisateurs et de modifier des informations les concernant.

**Tableau II.8 :** Cas d’utilisation : Gérer utilisateurs

|  |  |
| --- | --- |
| Cas d’utilisation | Gérer utilisateurs |
| Acteurs | Administrateur |
| Pré-condition | L’administrateur est identifié |
| Post-condition | L’administrateur a géré utilisateurs |
| Scénario | Relevant du « CRUD \*», ce cas d’utilisation sera traité sur le volet Ajouter utilisateur : 1. L’administrateur saisie les données relatives à l’utilisateur ne sachant pas s’inscrire et dont les informations ont été fournies au préalable par courriel électronique. |
| Exception | L’administrateur n’a pas toutes les informations requises pour accomplir avec succès la tâche d’ajout.  L’administrateur décide de ne pas inscrire une demande |

\*CRUD : Create/Read/Update/Delete sont un ensemble d’opération semblable, en traiter un revient à presque traiter tous les autres.

**VI.8 Raffinement du cas d’utilisation « Gérer droits d’accès »**

****

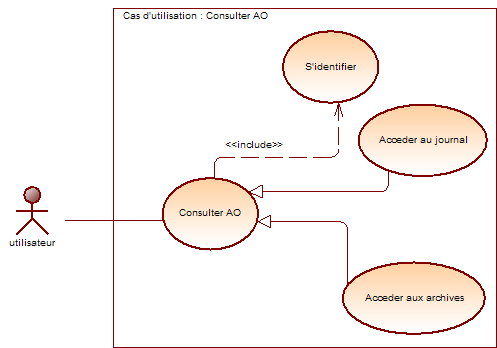
**Figure II.11 :** structuration du cas d’utilisation raffiné « Gérer droits d’accès »

L’administrateur doit pouvoir octroyer, révoquer et visualiser les droits d’accès des utilisateurs inscrits sur la plateforme. Il a ainsi, un droit de regard sur les opérations qu’ils sont autorisés à mener ou non. Ainsi ce cas d’utilisation est divisé selon les opérations citées ci-dessus.

**Tableau II.9 :** Cas d’utilisation : Gérer droits d’accès

|  |  |
| --- | --- |
| Cas d’utilisation | Gérer droits d’accès |
| Acteurs | Administrateur |
| Pré-condition | L’administrateur se doit de s’être identifié au préalable |
| Post-condition | Les droits d’accès des différents utilisateurs sont gérés |
| Scénario | Relevant du CRUD, nous traitons ici de la visualisation des droits :   1. L’administrateur accède au menu de visualisation des droits d’accès. 2. Il lance une requête spécifique pour un ou une liste d’utilisateurs donnés. 3. l’administrateur est en état de consulter les droits d’accès de chacun des utilisateurs inscrit sur la plateforme. |
| Exception | L’administrateur n’exécute pas de requête spécifique ou il décide d’interrompre la visualisation |

**VI.9 diagramme de cas d’utilisation « Consulter AO »**

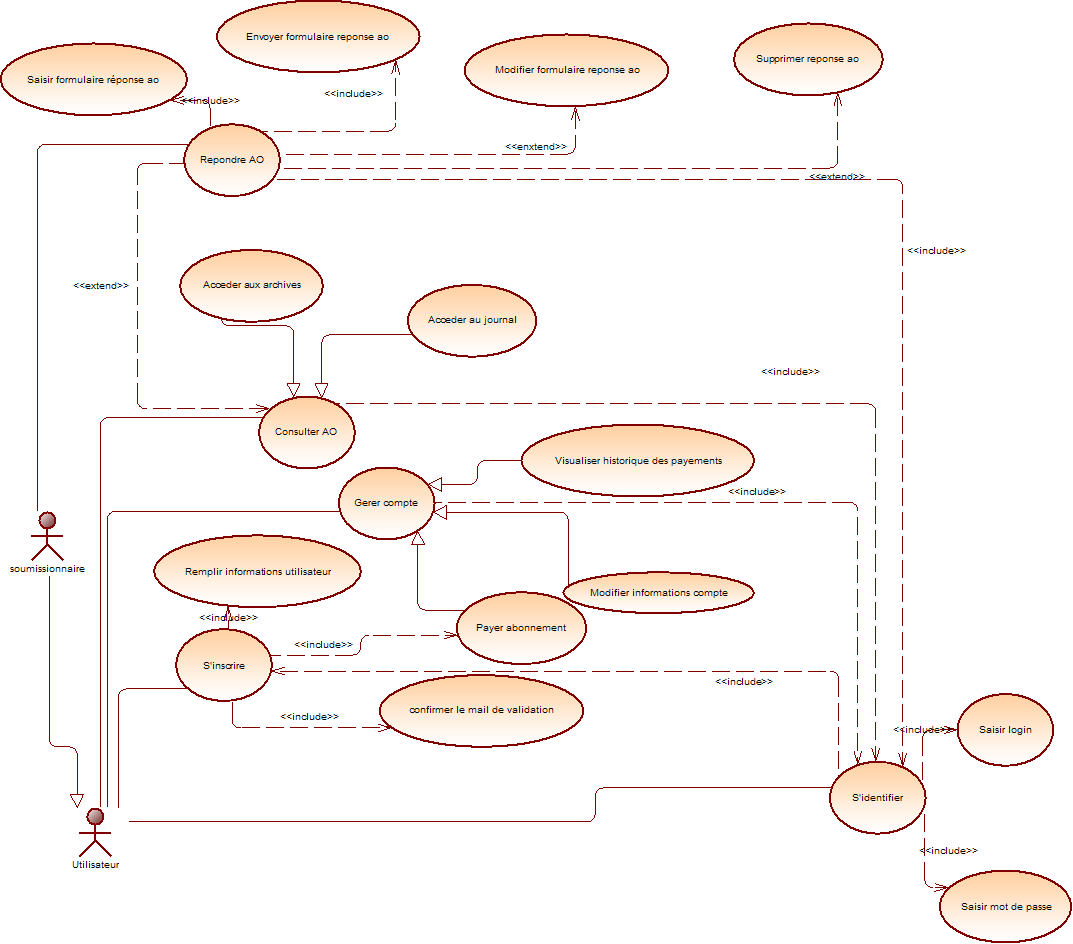


**Figure II.12 :** structuration du cas d’utilisation « consulter AO »

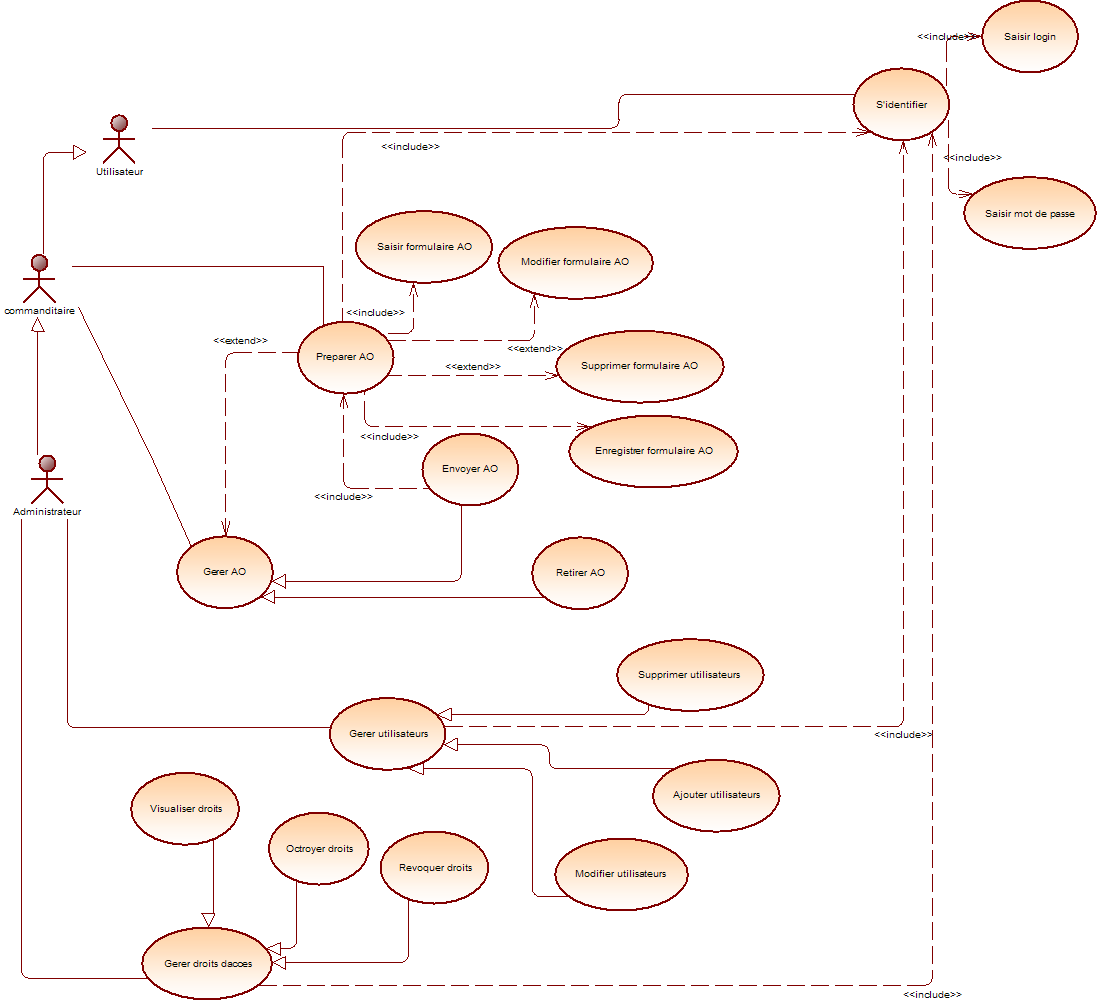
L’utilisateur pourra consulter les AO disponibles ou celles qui son archivées.

**Tableau II.10 :** Cas d’utilisation : Consulter AO

|  |  |
| --- | --- |
| Cas d’utilisation | Consulter AO |
| Acteurs | Soumissionnaire/Commanditaire/Administrateur |
| Pré-condition | L’utilisateur est identifié |
| Post-condition | Les AO ont été consultés |
| Scénario | Ce cas d’utilisation est subdivisé en deux parties :   1. L’utilisateur choisi de consulter les AO disponibles. 2. L’utilisateur choisi de consulter l’archive des AO qui ne sont plus disponibles. |
| Exception | L’utilisateur choisi de quitter le menu. |

****

**Figure II.13 :** Modèle final et raffiné du diagramme de cas d’utilisation (partie 1)



**Figure II.14 :** Modèle final et raffiné du diagramme de cas d’utilisation (partie 2)

* Ainsi, les cas d’utilisations bien définis et raffinés nous pouvons approfondir nos travaux pour une conception plus détaillé

1. **Conclusion**

Ayant réalisé cette phase, dans laquelle nous avons fait l’étude de l’existant et nous avons traité tous les cas d’utilisation, nous avons réussi à répondre aux questions suivantes :

* + Le projet vaut-il la peine d’être entrepris ?
  + Quels sont les principaux utilisateurs de notre futur système ?
  + Quelles fonctionnalités notre système doit-il offrir pour satisfaire les besoins des différents acteurs ?

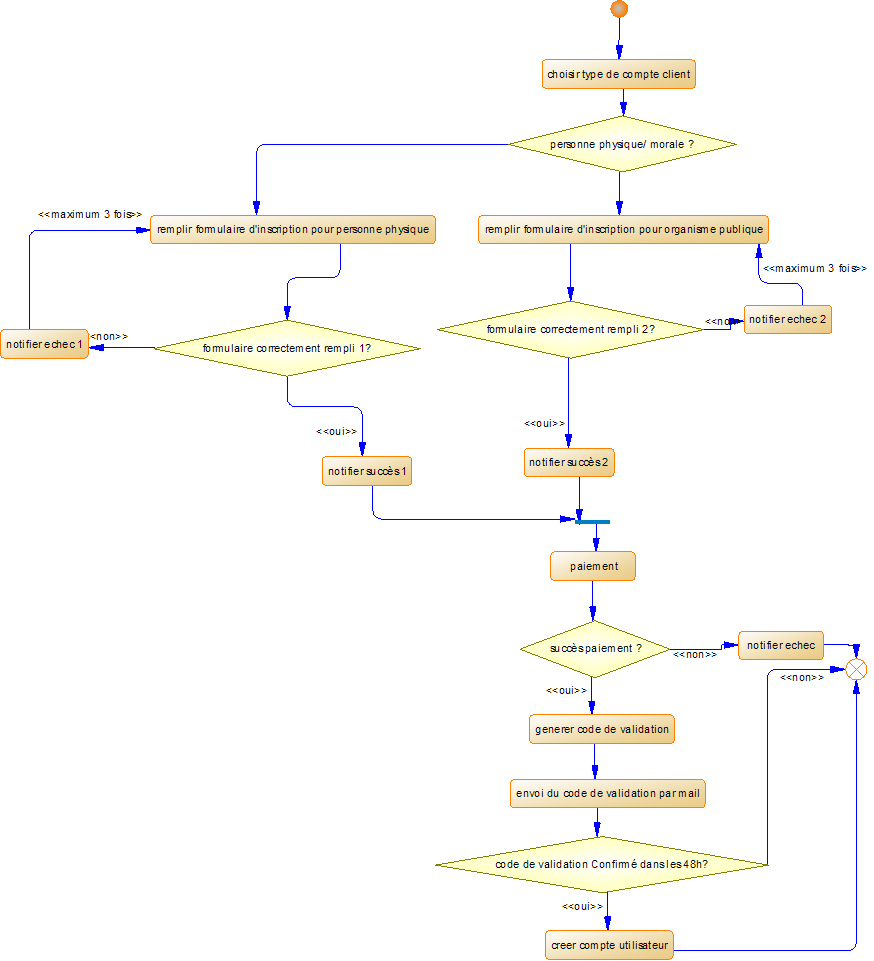
Ce qui nous a permis de passer à la phase d’élaboration, dans laquelle nous entamerons la capture de nouveaux besoins, l’analyse des cas d’utilisation secondaires et nouveaux, la conception des cas d’utilisation prioritaires et secondaires, l’implémentation des cas d’utilisation prioritaires.

*CHAPITRE III :***Phase d’élaboration**

**Introduction :**

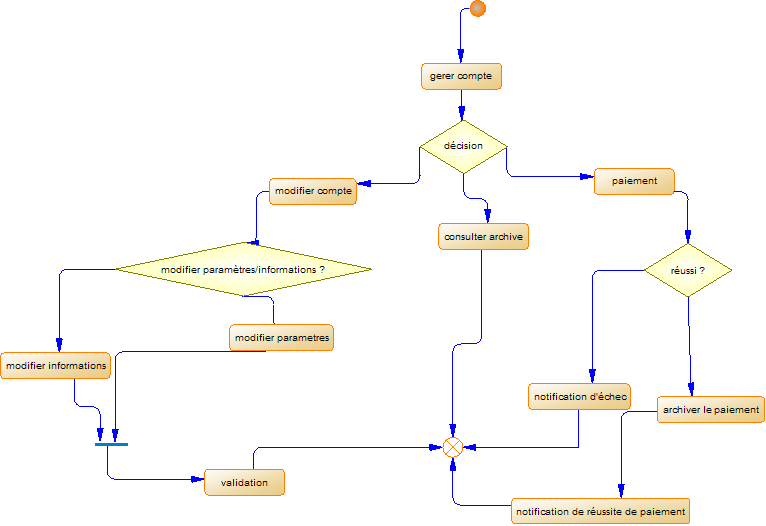
Ayant compris le contexte de notre système lors de la précédente phase, l’objectif maintenant est d’approfondir notre compréhension afin de mieux modéliser notre travail. En fait, pour la phase d’élaboration nous sommes appelés à présenter une conception des cas d’utilisation et à implémenter les plus importants.

1. **Diagrammes d’activité**
2. **Diagramme d’activité pour le cas d’utilisation « S’inscrire»**

****

**Figure III.1 :** diagramme d’activité « s’inscrire »

**Description :** l’utilisateur choisi le type de compte client avec lequel il va s’inscrire. S’agit-il de personne physique ou morale. Ensuite il remplit un formulaire d’inscription puis il passe pour faire la procédure de paiement qui, passée aboutira à la génération d’un code de validation qui devra être validé dans les 48 heures pour que le compte soit opérationnel.

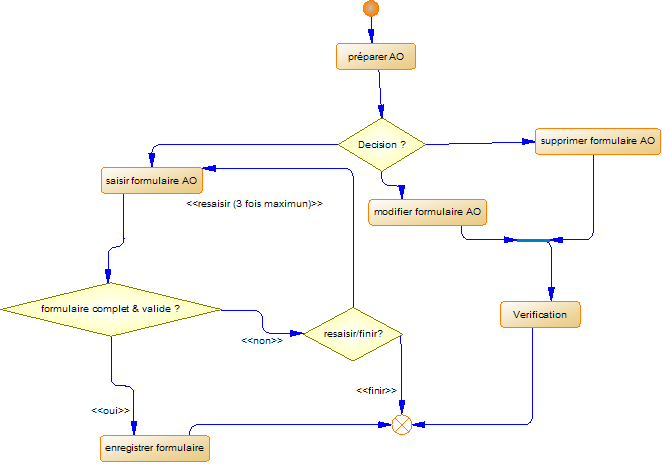
1. **Diagramme d’activité pour le cas d’utilisation « Gérer compte»**

**Figure III.2 :** diagramme d’activité « gérer compte »

**Description :** L’utilisateur, accédant au menu de gestion du compte choisi :

* + Modifier le compte soit en modifiant ses paramètres ou ses informations et puis il validera.
  + Consulter les archives de ces anciens paiements
  + De payer son abonnement. Et en cas de paiement réussi, le résultat sera archivé avec une notification de réussite de paiement sinon une notification d’échec sera envoyée.

1. **Diagramme d’activité pour le cas d’utilisation « Préparer AO»**

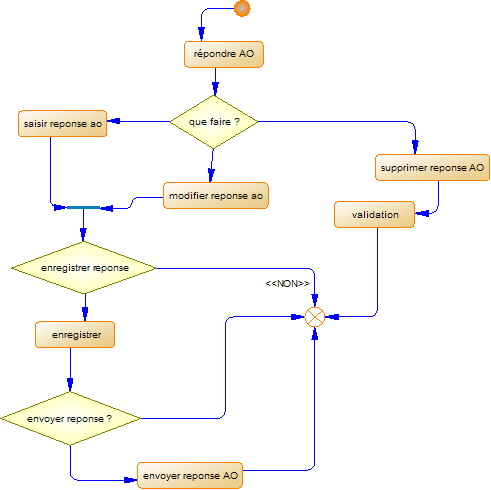


**Figure III.3 :** diagramme d’activité « préparer AO »

**Description :** le commanditaire, après avoir accédé au menu décision, il choisit soit de supprimer un formulaire AO ou d’en modifier un. S’il procède à une de ces deux actions, il devra les valider.

Il peut aussi faire la décision de saisir un formulaire AO qu’il complètera et validera puis il devra l’enregistrer pour le lancer.

1. **Diagramme d’activité pour le cas d’utilisation « Répondre AO»**

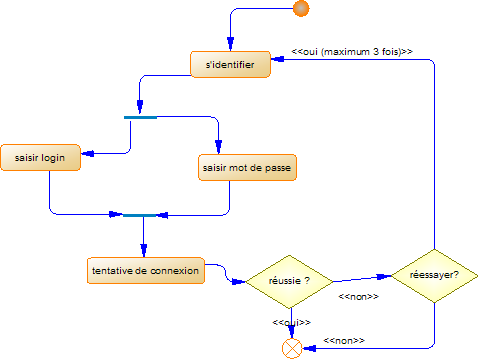


**Figure III.4 :** diagramme d’activité « répondre AO »

**Description :** Le commanditaire après son accès au menu répondre AO, choisi soit :

* + De saisir une réponse. Ensuite il aura la possibilité d’enregistrer cette réponse et de l’envoyer.
  + *In extenso,* il pourra modifier une ancienne réponse en suivant la même procédure sus-dite.
  + Il aura la possibilité de supprimer une réponse déjà faite.

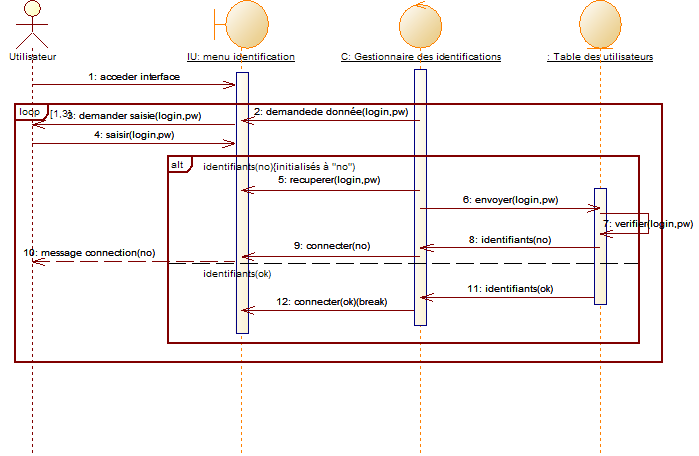
1. **Diagramme d’activité pour le cas d’utilisation « S’identifier»**



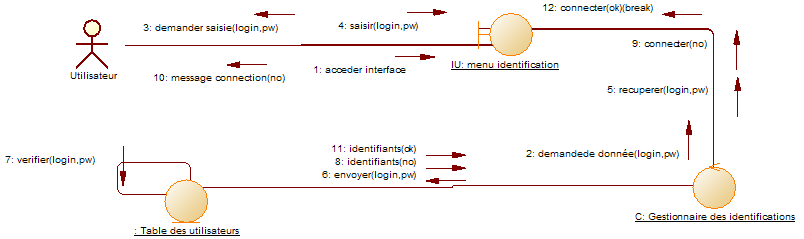
**Figure III.5 :** diagramme d’activité « s’identifier »

**Description :** l’utilisateur ayant accédé au menu « s’identifier » est amené à saisir son login et son mot de passe et si ces dernier sont valables, la connexion est établie

1. **Diagrammes de séquence et de collaboration**
2. **Diagramme de séquence et de collaboration pour le cas d’utilisation « S’identifier»**



**Figure III.6 :** diagrammes de séquence « s’identifier »



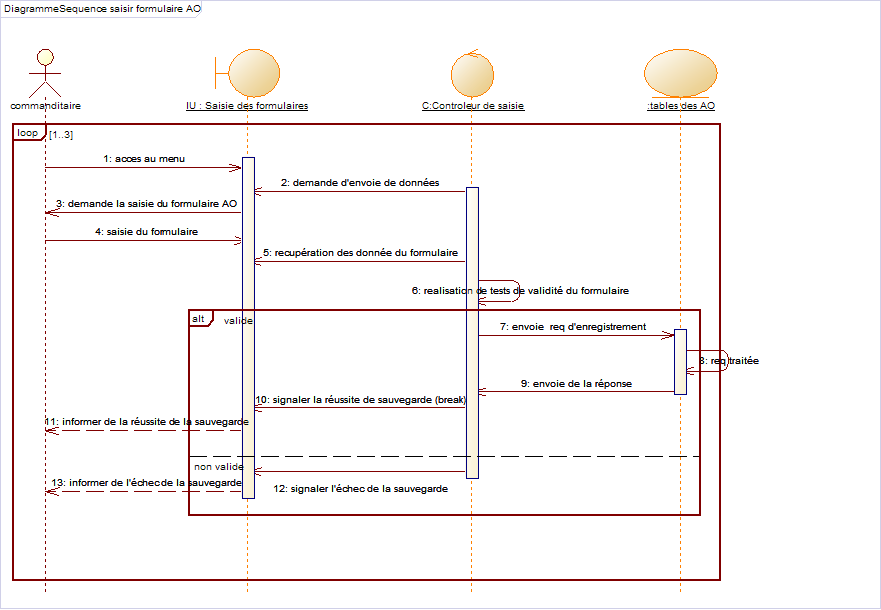
**Figure III.7 :** diagramme de collaboration « s’identifier »

**Description :** l’utilisateur accédant à interface(1) d’identification, le contrôleur demande à l’interface la saisie du login et du mot de passe, qui la demande à son tour à l’utilisateur qui les saisit pour qu’il soit récupéré par le gestionnaire des identification (2, 3,4,5 )

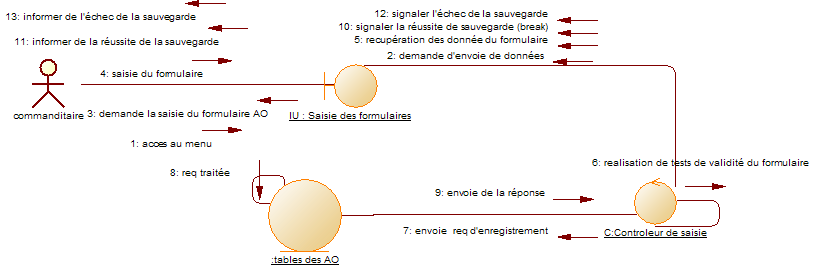
Ce dernier envoie une requête à la table des utilisateurs qui vérifie la justesse des login et mot de passe.

Selon le résultat (8,11) l’utilisateur est connecté ou non (9,12).   
 Si l’utilisateur n’arrive à se connecter un message (10) lui apparaitra et il lui restera 2 chances pour pouvoir se connecter.

1. **Diagramme de séquence et de collaboration pour le cas d’utilisation «Saisir formulaire AO »**



**Figure III.8 :** diagramme de séquence « saisir formulaire AO »

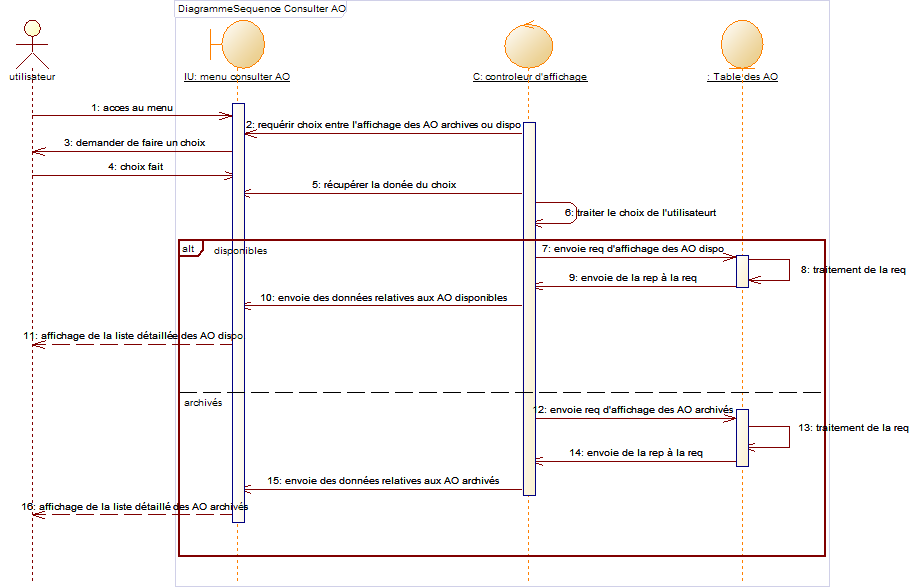


**Figure III.9 :** diagramme de collaboration « saisie formulaire AO »

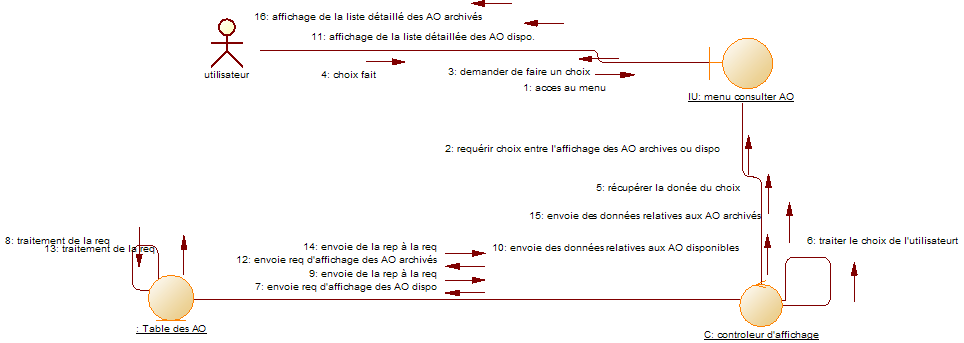
**Description :** le commanditaire accède au menu (1), le contrôleur de saisie demande l’envoie de données saisies à l’interface, entrant ainsi dans la phase de saisie qui qui se terminera après la récupération des données du formulaire (2,3,4,5 ).

Le contrôleur réalise les tests de validités des données saisies(6). Si les données saisie sont valides alors celles-ci sont enregistrées (7,8) et la réussite de l’enregistrement est signalée au commanditaire (9,10,11). Sinon l’échec de la saisie est signalé lui aussi (13).  
 Aussi, le commanditaire peut répéter en cas d’échec la séquence de saisie 3 fois.

1. **Diagrammes de séquence et de collaboration « Consulter AO)**



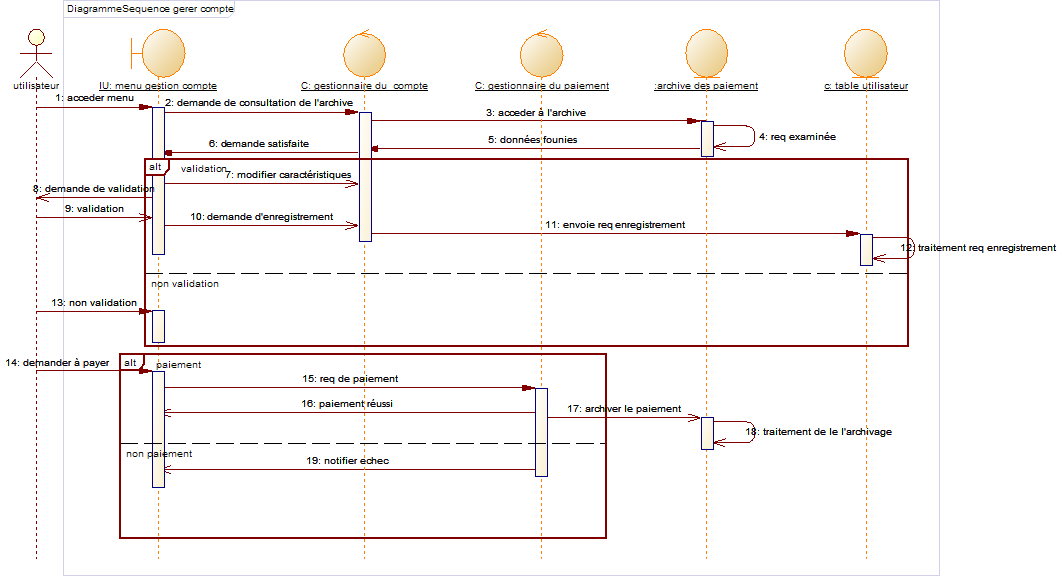
**Figure III.10 :** diagramme de séquence « consulter AO »



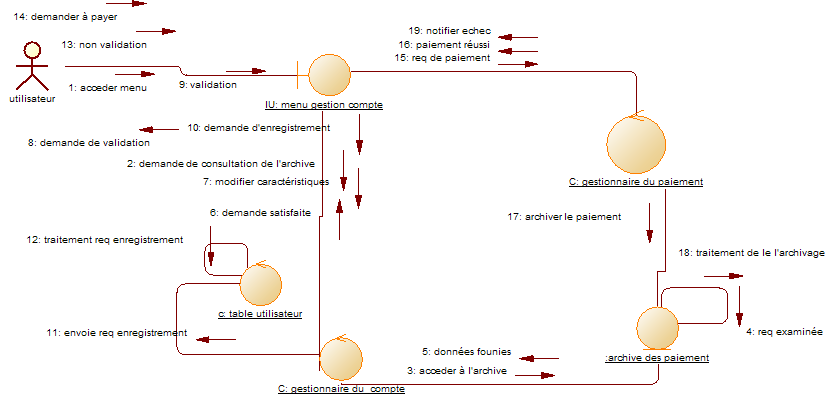
**Figure III.11 :** diagramme de collaboration « consulter AO »

**Description :** Après son accès au menu (1), l’utilisateur entre dans la phase de choix ou il choisit de consulter les AO disponibles en vue de trouver une réponse et ceux qui sont déjà archivés (2,3,4,5,6). Si l’utilisateur choisit de de consulter les AO disponibles, une requête est envoyée en ce sens par le contrôleur à la base, ce qui aboutir par l’affichage d’une liste détaillée(7,8,9,10,11). Sinon une action équivalente est lancée pour les AO déjà archivé (12,13,14,15,16)

1. Diagrammes de séquence et de collaboration « gérer compte »



**Figure III.12 :** diagramme de séquence « gérer compte »

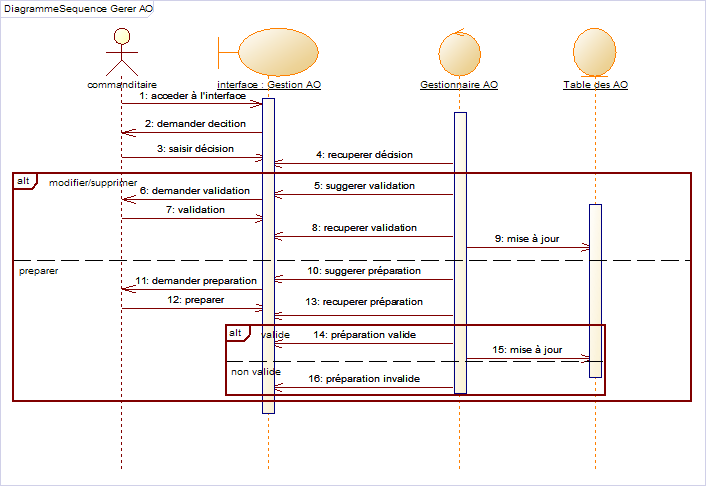


**Figure III.13 :** diagramme de collaboration « gérer compte »

**Description :** Suite à son accès à l’interface du menu de gestion de compte (1), plusieurs choix s’offrent à l’utilisateur :

* + La consultation des anciens paiements des abonnements par le biais de d’une requête lancée à la table des paiements via le contrôleur « gestionnaire du compte » (2, 3,4,5,6)
  + La modification des paramètres du compte et des informations concernant l’utilisateur qui seront validés(9) pour permettre leur enregistrement (8,8,10,11,12) ou non(13).
  + Enfin, une procédure de payement (15,16,17,18) qui prendra en compte l’éventualité de l’échec de la procédure(19).

1. **Diagrammes de séquence et de collaboration « gérer AO »**



**Figure III.14 :** diagramme de séquence « gérer AO »

**Figure III.15 :** diagramme de collaboration « gérer AO »

**Description :** suite à l’accès à l’interface de gestion des AO (1), le commanditaire est amené à :

* + Modifier/supprimer (parce que cela suit la presque même procédure) qui suivra une séquence commençant par la validation du choix et se terminant par une mise à jour de la table AO
  + Préparer un AO, autrement plus aboutie que ce qui est fait ci-dessus.

1. **Conclusion**

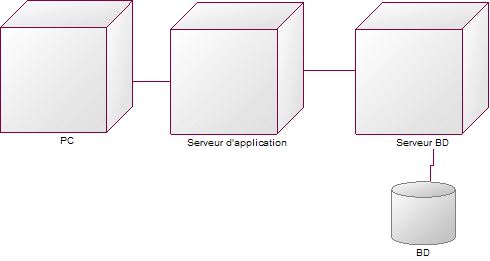
L’étude conséquente des diagrammes d’activités, de séquence et de collaboration nous a permis de mettre en valeur les principales fonctionnalités de l’application, qui ainsi élaboré et implémenté, nous facilite une transition simple et effective à la phase de construction.

*CHAPITRE IV :***Phase de construction**

**Introduction :**

L’objectif essentiel dans la phase de construction, est de centrer le développement de l’application web, prête à être exploitée par les différents utilisateurs. Nous terminons dans ce chapitre ci, la conception et l’implémentation.

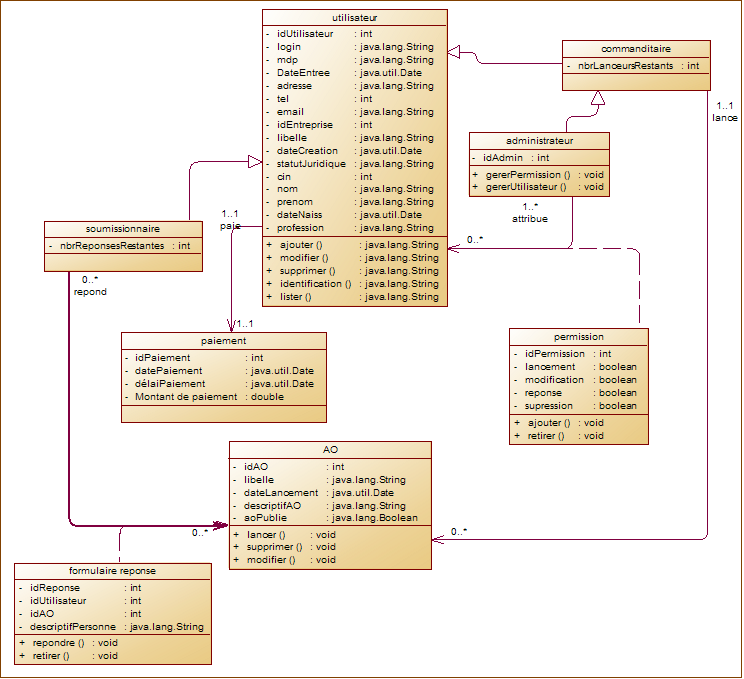
1. **Conception**
2. **Architecture matérielle : Diagramme de déploiement**



**Figure IV.1 :** diagramme de déploiement

**Description :** Vue la nature de notre projet, qui est la conception et la réalisation d’une plateforme de gestion d’AO, nous avons opté pour l’architecture 3 tiers qui sépare 3 couche différentes. La couche métier, la couche donnée et la couche présentation. Ce fut préférable à l’architecture 2 tiers, dès lors que l’on a constaté que la sécurité accrue du fait de sa définition indépendamment réalisée de chaque service. Ce choix se fit aussi, en raison d’une plus grande flexibilité et de meilleures performances, étant donné le partage des tâches entre les différents serveurs.

1. **Architecture Logicielle : Diagramme de classe**

****

**Figure IV.2 :** diagramme de classe

**Description :**

**Classe utilisateur :** représente une entreprise, un organisme public, ou un particulier qui se connecte en vue de naviguer sur la plateforme et effectuer plusieurs opération qui lui sont réservées.

**Classe formulaire réponse :** représente le formulaire de réponse qu’un soumissionnaire remplit et envoie, dans le but de répondre à n appel d’offre.

**Classe commanditaire :** représente une personne morale ou physique qui lance un AO.

**Classe soumissionnaire :** représente une personne physique ou morale qui répond à un AO

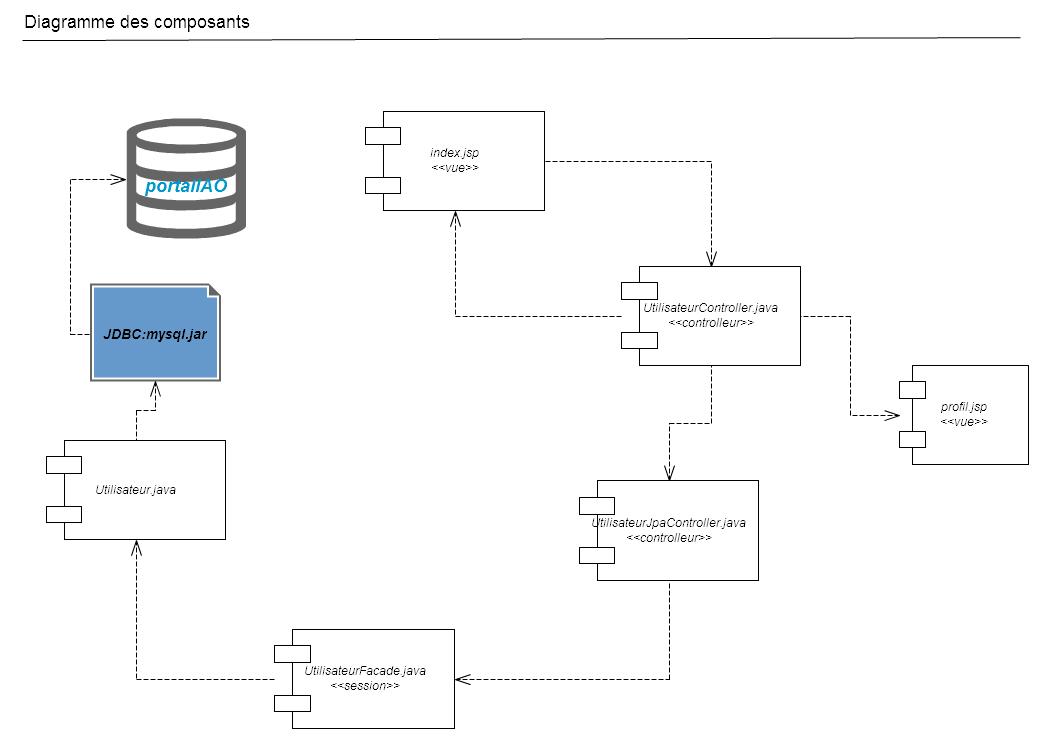
**Classe administrateur :** représente un administrateur de la plateforme

**Classe paiement :** représente les payements d’abonnement effectués.

**Classe AO :** représente les AO

**Classe permission :** permet la gestion des différentes permissions pour les utilisateurs

1. **Implémentation des cas d’utilisation prioritaires :**

**1 implémentation du cas d’utilisation «inscription»** 

**Figure IV.3:** diagramme des composants de l’inscription.

L’implémentation de ce cas d’utilisation se présente comme suit :

* + Dans l’interface inscription, nous avons implémenté un formulaire d’inscription et un bouton “s’inscrire” qui permet de confirmer sa validation.

- la page index.jsp invoque les classes métiers propres a l’utilisateur et en cas de succès, le contrôleur fait une redirection vers la page du profil. Mais en cas d'échec, un message d’erreur sera affiché dans la même page.

**2. implémentation du cas d’utilisation «lancement d’un appel d’offre» :**

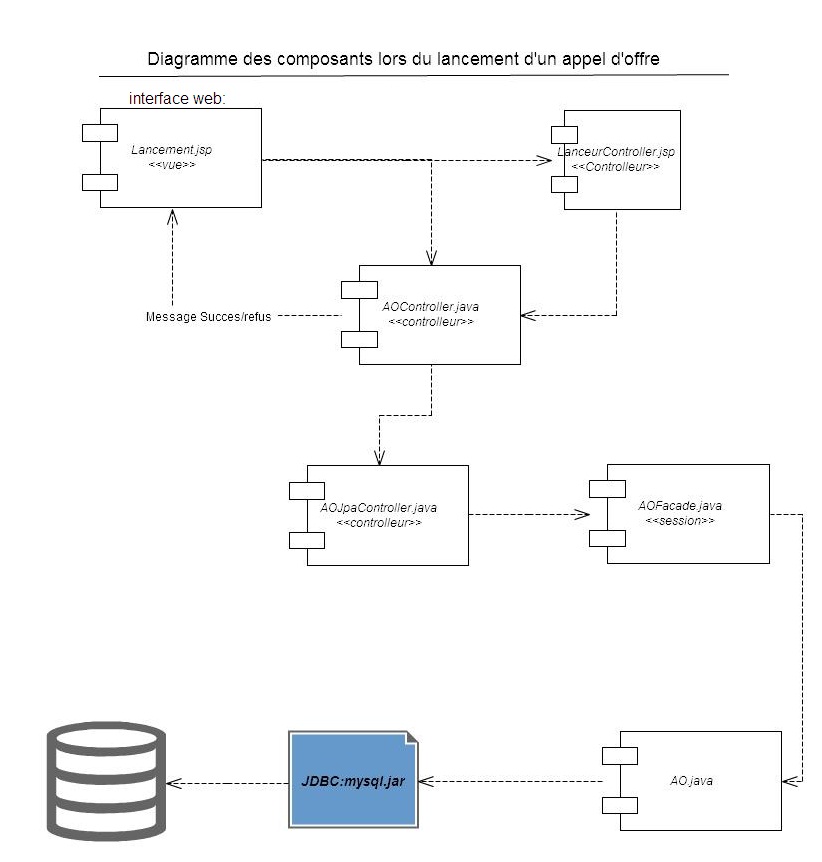
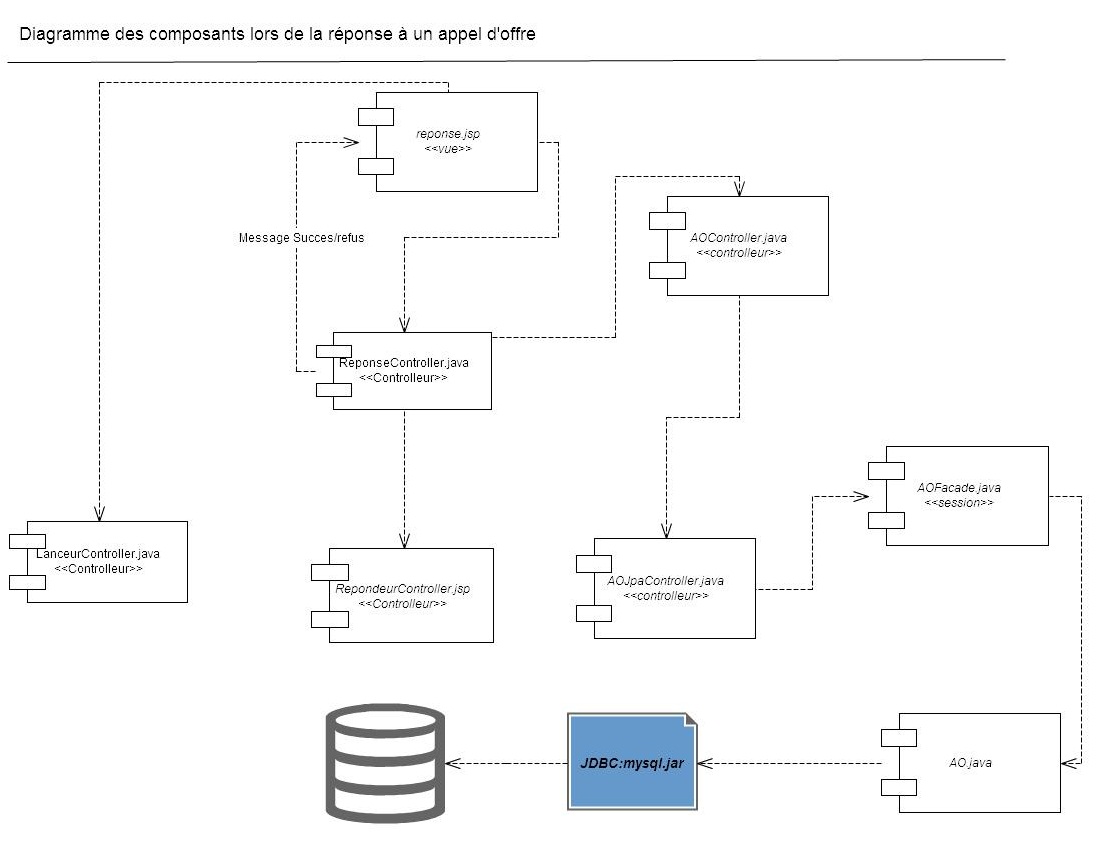


figure .2: diagramme des composants du lancement d’un appel d’offre.

Lors du lancement de l’appel d’offre, le commanditaire rempli un formulaire et clique sur le lien du lancement, ce lien invoquera les deux contrôleurs utilisateur et AO,un message de succès ou d'échec va être affiché en fonction du succès de la transaction.

**3. Implémentation du cas d’utilisation «réponses a un appel d’offre» :**

**Figure III.3:** diagramme des composants de la réponse à un appel d’offre.

Quand un soumissionnaire répond a un appel d’offre, il doit remplir le formulaire de réponse, la page reponse.jsp invoquera le contrôleur de la réponse et celui du lanceur. Ce Premier invoquera les contrôleurs de l’appel d’offre et celui du répondeur.

En cas de succès ou d'échec, un message va être affiché dans la même page.

**III. Mise en place du système**

Les outils informatiques utilisés pour le développement de notre application sont les suivants :

**Environnement de développement Intégré Netbeans:**

L'EDI NetBeans est un environnement de développement - un outil pour les programmeurs pour écrire, compiler, déboguer et déployer des programmes. Il est écrit en Java - mais peut supporter n'importe quel langage de programmation. Il y a également un grand nombre de modules pour étendre l'EDI NetBeans. L'EDI NetBeans est un produit gratuit, sans aucune restriction quant à son usage.

Également disponible, La Plateforme NetBeans; une fondation modulable extensible utilisée comme brique logicielle pour la création d'applications bureautiques. Les partenaires privilégiés fournissent des modules à valeurs rajoutées qui s'intègrent facilement à la Plateforme et peuvent être utilisés pour développer ses propres outils et solutions.

Les deux produits sont open source et gratuits pour un usage commercial et non-commercial. Le code source est disponible pour réutilisation sous la Common Développent and Distribution License (CDDL).

NetBeans IDE 7.1.est le premier IDE supportant les tout derniers standards et spécifications de la plate-forme Java, avec notamment le support complet du développement pour JavaFX 2.0 et JDK7.

Cette nouvelle version offre également le support d'Oracle WebLogic Server 12c, nouvelle édition récemment annoncée par Oracle de son serveur applicatif, conçue pour offrir le support de Java Platform Enterprise Edition (Java EE) 6 aux développeurs qui réalisent des applications sur la plate-forme Oracle WebLogic Server.

NetBeans IDE 7.1 intègre également des améliorations importantes pour Swing GUI Builder, le support de CSS3 ainsi que des outils pour le débogage visuel des interfaces utilisateur JavaFX et Swing.

**Serveur d’application Glassfish:**

GlassFish 3.0.1 repose sur une exécution modulaire et flexible basée sur la norme OSGi. Il permet aux entreprises de créer et de déployer des applications Web à l'aide du profil Web Java EE 6 léger et de facilement exploiter la puissance de la plate-forme Java EE 6 complète pour les applications d'entreprise. Les développeurs bénéficient également du modèle de programmation simplifié et des améliorations de productivité proposés par Java EE 6. Le résultat est une plate-forme flexible apte à offrir une solution ciblée à tout problème, réduisant ainsi le coût et la complexité. GlassFish 3.0.1 utilisant une architecture de micro-noyau basée sur OSGi, les développeurs peuvent commencer avec le profil Web Java EE 6 et utiliser le centre de mise à jour pour effectuer une mise à niveau dynamique vers la plate-forme Java EE 6.

GlassFish 3.0.1 propose également un alignement avec NetBeans 6.9 , la prise en charge d'Eclipse , la prise en charge des scripts JRuby/Ruby et Groovy/Graal, une console d'administration conviviale et la connectivité au centre de mise à jour. Une fois que vous aurez installé et commencé à utiliser GlassFish 3.0.1, n'hésitez pas à nous faire part de vos impressions et commentaires.

Présentation de la licence :

Les distributions de la communauté Glassfish sont disponibles sous licence double composée de la licence Common Development and Distribution License (CDDL) v1.0 et de la licence GNU General Public License (GPL) v2. Reportez-vous aux fichiers lisez-moi tiers et webprofile-third-party-readme pour connaître les composants externes inclus dans GlassFish 3.0.1 et les licences associées.

**MySQL:**

MySQL, le plus populaire des serveurs de bases de données SQL Open Source, est développé, distribué et supporté par MySQL AB. MySQL AB est une société commerciale, fondée par les développeurs de MySQL, qui développent leur activité en fournissant des services autour de MySQL.

Le logiciel MySQL utilise la licence GPL (GNU General Public License), http://www.gnu.org/licenses/ , pour définir ce que vous pouvez et ne pouvez pas faire avec ce logiciel, dans différentes situations. Si vous ne vous sentez pas confortable avec la licence GPL ou bien que vous devez intégrer MySQL dans une application commerciale, vous pouvez acheter une licence commerciale auprès de MySQL AB.

Le serveur de bases de données MySQL est très rapide, fiable et facile à utiliser Si c'est ce que vous recherchez, vous devriez faire un essai. Le serveur de bases de données MySQL dispose aussi de fonctionnalités pratiques, développées en coopération avec nos utilisateurs. Vous pouvez trouver une comparaison des performances du serveur MySQL avec d'autres systèmes de bases de données dans nos pages de tests de performances. See Section 7.1.4, « La suite de tests MySQL ».

MySQL Server fonctionne en mode client/serveur ou en système embarqué. Le serveur MySQL est un système client / serveur qui est constitué d'un serveur SQL multi-threadé qui supporte différentes interfaces, clients, bibliothèques et outils d'administration, ainsi qu'une large gamme de pilotes pour différents langages (API).

**IV. Le processus unifié**

Le processus unifié est un ensemble de principes génériques adaptables en fonction des spécificités des projets. Il répond aux préoccupations suivantes :

– Qui participe au projet ?

– Quoi, qu’est-ce qui est produit durant le projet ?

– Comment doit-il être réalisé ?

– Quand est réalisé chaque livrable ?

C’est donc un patron de processus pouvant être adapté à une large classe de systèmes logiciels, à différents domaines d’application, à différents types d’entreprises, à différents niveaux de compétences et à différentes tailles de projets.

**IV.1 Le processus unifié : concept et intérêts**

Le processus unifié a pour objectif de maîtriser la complexité des projets informatiques en en diminuant les risques. Pour ce faire, il se base sur les concepts suivants :

**Itératif et incrémental :**

Dans le but de réduire la complexité en la maîtrisant, un projet informatique se trouve décomposé en sous projets qui représentent chacun une itération. Les itérations désignent des étapes de l’enchaînement d’activités, tandis que les incréments correspondent à des stades de développement du produit.

L’utilisation d’un processus itératifs présente de nombreux avantages :

- Limiter les coûts, en termes de risques, aux strictes dépenses liées à une seule itération.

- Limiter les risques de retard de mise sur le marché du produit développé grâce à un feed-back rapide.

- Accélérer le rythme de l’ensemble du développement, car il permet aux développeurs de travailler plus efficacement vers des objectifs clairs, à court terme, plutôt qu’en fonction d’un planning à long terme soumis à d’inévitables dépassements et surtout changements.

- Prendre en compte le fait que les besoins des utilisateurs et les exigences correspondantes ne peuvent être intégralement définis à l’avance. Grâce à un développement itératif, ces besoins se dégagent peu à peu des itérations successives pour faciliter l’adaptation à l’évolution des besoins.

**III.1.a Basé sur les cas d’utilisation :**

Le processus unifié utilise le langage UML pour la création des plans d’élaboration et de construction du système logiciel. UML fait partie intégrante du processus unifié, ils ne sont pas sélectionnés de façon isolée, mais ils sont développés en tandem avec l’architecture du système. Les cas d’utilisation guident la conception de l’architecture du système, qui influence, à son tour, leur sélection pour telle ou telle itération. L’architecture et les cas d’utilisation évoluent de façon parallèle au cours du cycle de vie du développement.

**III.1.b Centré sur l’architecture :**

L’architecture d’un système logiciel peut être décrite comme étant les différentes vues du système. À partir d’une vision au niveau des besoins l’architecte se focalise sur une partie en l’affinant et en créant un système. Après avoir affiné successivement les modèles produits l’architecte se précise.

L’architecture fournit la structure qui servira de cadre au travail effectué au cours des itérations, tandis que les cas d’utilisation définissent les objectifs et orientent le travail de chaque itération. La décomposition suivant un mode itératif et incrémental permet de mieux gérer les risques et de limiter la complexité des applications.

**IV.2 Vie du processus unifié :**

Le processus unifié répète un certain nombre de fois une série de cycles constituant la construction d’une génération du système. Tout cycle se conclut par la livraison d’une version du produit aux clients et s’articule en quatre phases :

– L’incubation

– L’élaboration

– La construction

– La transition

Chacune de ces phases se subdivise à son tour en itérations.

Pour mener efficacement un tel cycle les développeurs ont besoin de toutes les représentations du produit logiciel :

– Un modèle des cas d’utilisation exposant tous les cas d’utilisation et leurs relations avec les utilisateurs

– Un modèle d’analyse poursuivant deux objectifs : détailler les cas d’utilisation et procéder à une première répartition du comportement du système entre divers objets appropriés.

– Un modèle de conception définissant la structure statique du système sous forme de sous-systèmes, classes et interfaces, et les cas d’utilisation réalisés sous forme d’interactions entre les sous-systèmes, les classes et les interfaces.

– Un modèle d’implémentation intégrant les composants (représentant le code source) et la correspondance entre les classes et les composants.

– Un modèle de déploiement définissant les noeuds physiques des ordinateurs et l’affectation de ces composants sur ces noeuds.

* Un modèle de tests décrivant les cas de test vérifiant les cas d’utilisation.

Un modèle est une abstraction du système. Tous les modèles sont liés ensemble. Ils représentent le système comme un tout. Les éléments de chacun des modèles présentent des dépendances de traçabilité.

**IV.3 Les phases du PU :**

**IV.3.a La phase d’incubation :**

Cette phase répond aux questions suivantes :

– Que va faire le système ? Par rapport aux utilisateurs principaux, quels services va-t-il rendre ?

– Quelle va être l’architecture générale (cible) de ce système ?

– Quels vont être : les délais, les coûts, les ressources, les moyens à déployer ?

Au cours de cette phase les risques majeurs sont identifiés et hiérarchisés, une planification est apportée avec une estimation approximative du projet dans son ensemble.

La phase d’inception ne propose pas une étude complète du système envisagé. Elle ne vise qu’à dégager quelques cas d’utilisation qui vont être fondamentaux pour l’étude initiale.

L’objectif, à ce niveau, est d’avoir une vue d’ensemble du système pour arriver à la conclusion que le système est réellement faisable et qu’il est souhaitable de le réaliser.

**IV.3.b La phase d’élaboration :**

-La phase d’élaboration permet de préciser la plupart des cas d’utilisation et de concevoir et valider l’architecture du système. La stabilisation de l’architecture est l’objectif principal de cette phase.

Cette architecture se réalise par itérations de manière à arriver progressivement à la cible définie dans ses grandes lignes lors de la phase d’inception.

**IV.3.c La phase de construction :**

L’architecture est aboutie et le produit s’apparente à l’application satisfaisant les exigences. Le but est de coller parfaitement aux exigences. La phase de construction comprend les activités suivantes :

– Extension de l’identification, de la description et de la réalisation des cas d’utilisation à l’ensemble des cas d’utilisation

– Finalisation de l’analyse, de la conception, de l’implémentation et des tests.

– Préservation de l’intégrité de l’architecture.

– Surveillance des risques critiques et significatifs identifiés dans les deux premières phases et réduction des risques.

**IV.3.d La phase de transition :**

Dans cette phase le produit est mis en production, il est déployé chez le client avec les recommandations nécessaires.

Chapitre V:  
Phase de transition

Introduction :

La phase de transition constitue la dernière phase du processus unifié, dans laquelle l’application est en version bêta. Nous terminons par la représentation de quelques interfaces de notre application web.

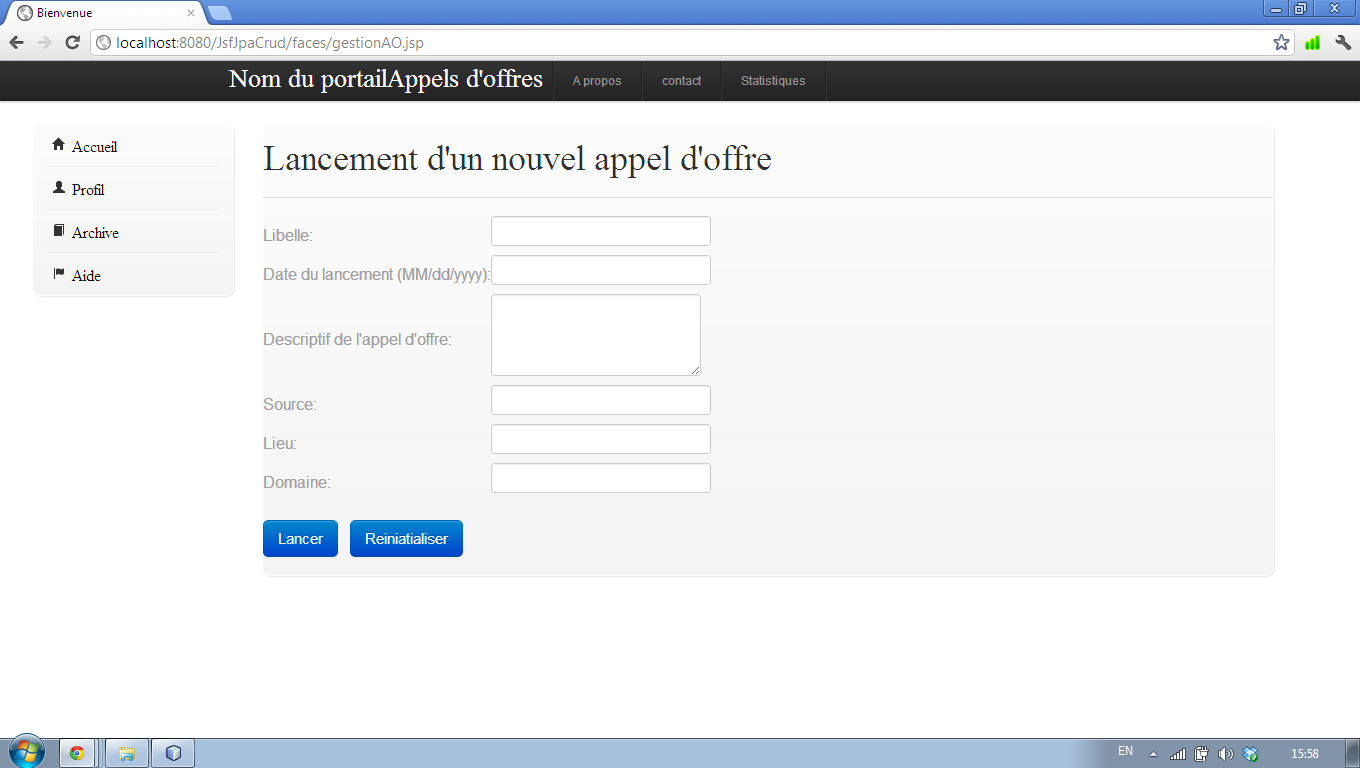
Page d'accueil



**Figure V.1:** Interface d’accueil.

La page d'accueil permet a l'utilisateur de se connecter au site web ainsi permet aux nouveaux utilisateurs de s'inscrire. Une présentation des principaux domaines des appels d'offres est disponible en cliquant sur les liens.

Page lancement:



**FigureV.2 :** interface Lancement.

Chaque commanditaire peut lancer des appels d'offres comme décrit dans cette interface.

Administration du site:

1. **La gestion des utilisateurs:**



**Figure V.3:** interface administration des utilisateurs.

Tous les utilisateurs sont visibles par les administrateurs ainsi ces derniers peuvent ajouter, modifier ou bien supprimer un des utilisateurs.

**Page : gestion des appels d'offres:**

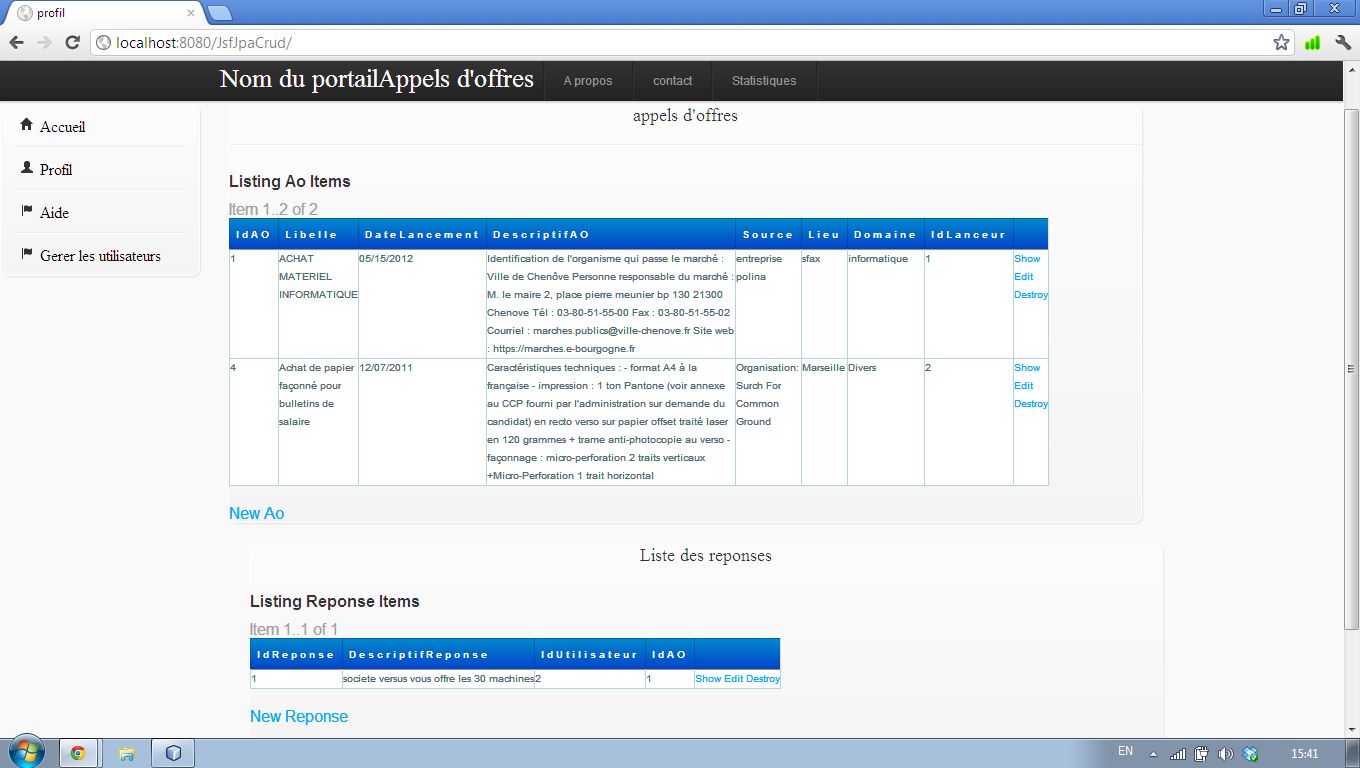


figure V.4: interface administration des appels d’offres.

Comme la figure (V.4) le montre, tous les appels d'offre du portail sont visibles par les administrateurs,ils peuvent aussi lancer, modifier ou bien supprimer des appels d'offres ainsi que les réponses.

Page contact:

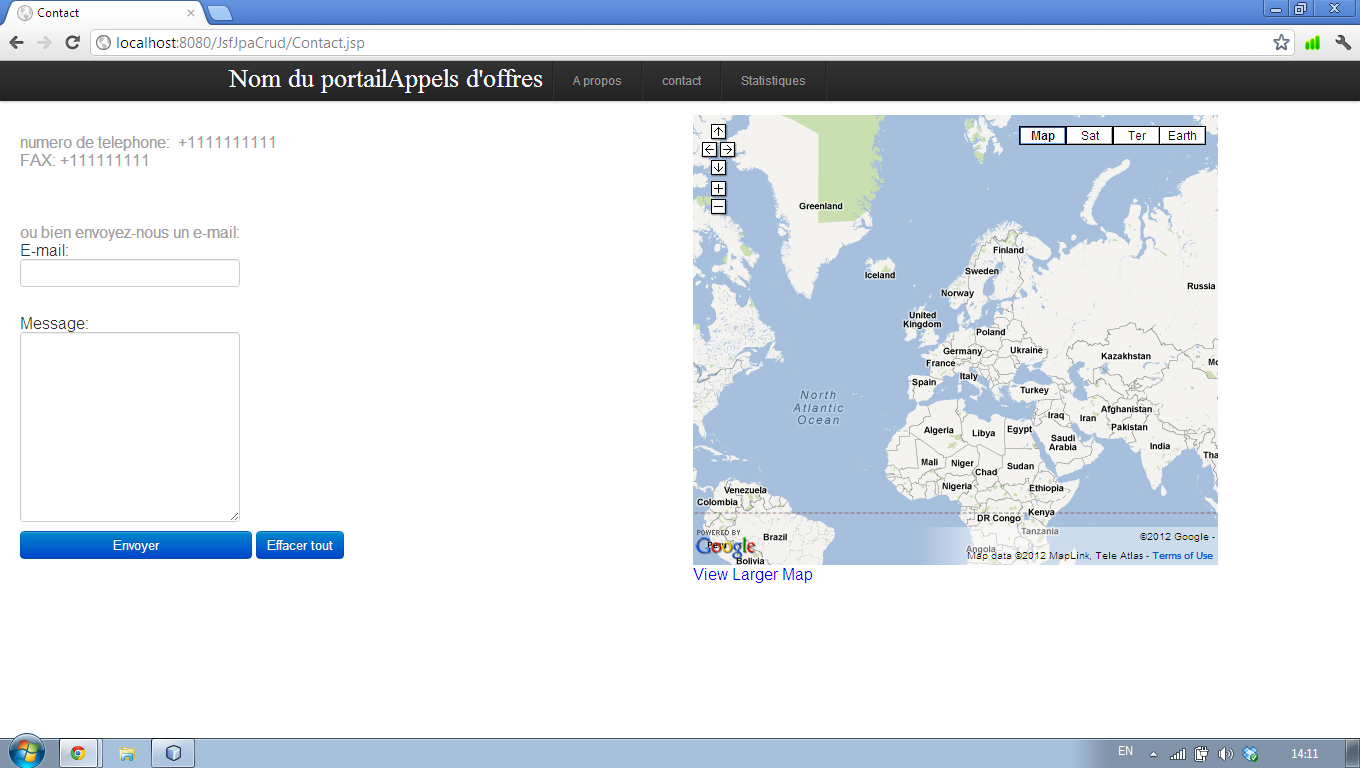
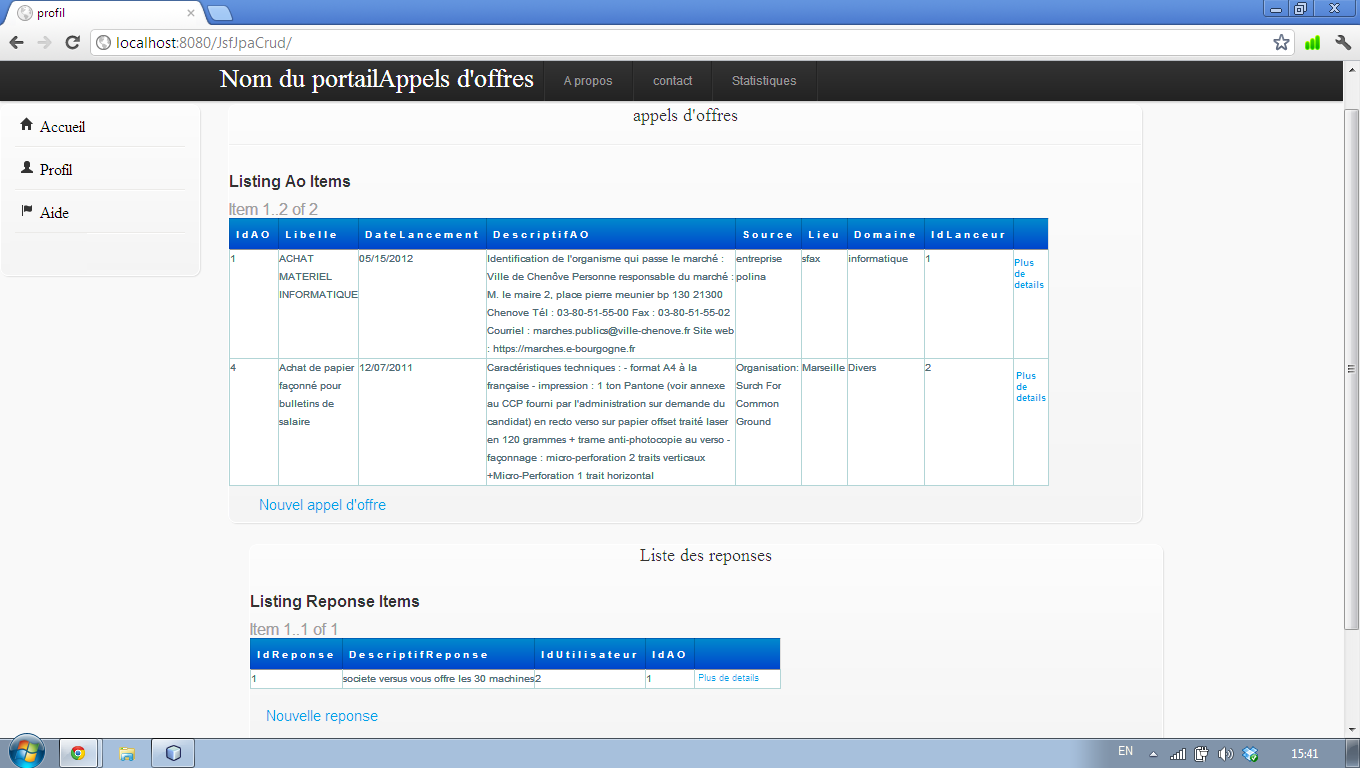


figure V.5: interface Contact.

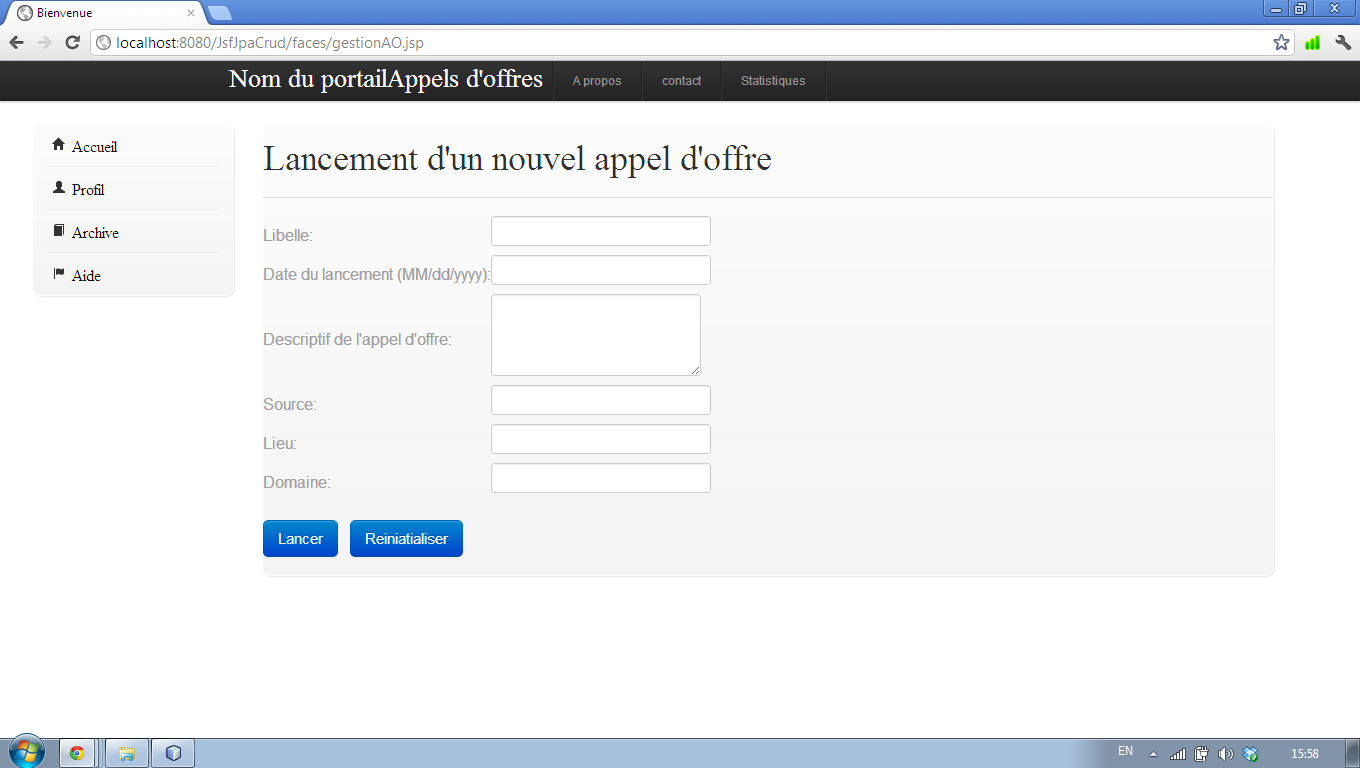
Cette page permet de contacter le propriétaire de ce portail par téléphone, par Fax ou bien par mail quand le visiteur saisi son adresse électronique et le corps du message, un e-mail va être envoyer automatiquement vers l'adresse -mail de l'entreprise.Notre site web offre aussi le service de google maps.

Page du profil commanditaire:



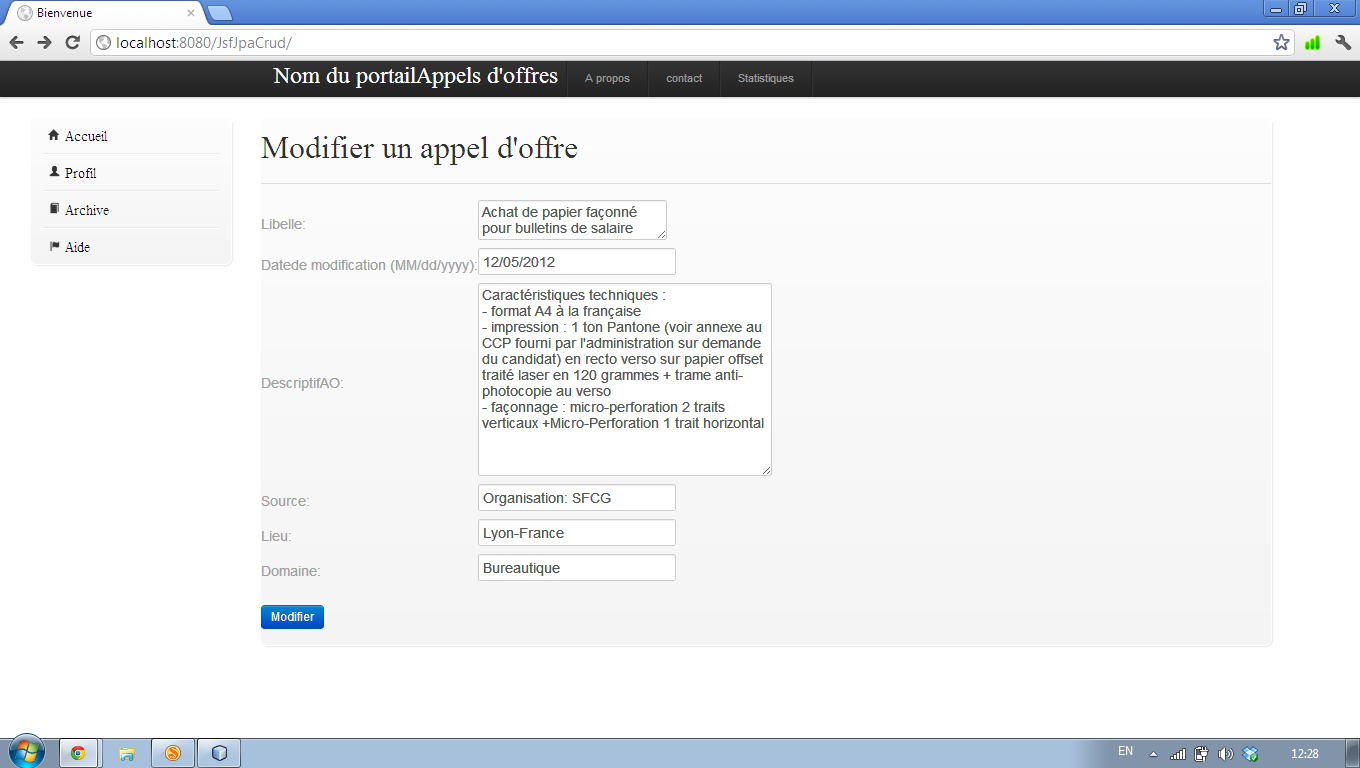
**Figure V.6:** interface profil.

Lorsqu'un utilisateur est commanditaire, il lance les appels d'offres et il visualise toutes les réponses qui ont été faites par les soumissionnaires.



**Figure V.7:** interface du lancement d’un appel d’offre.

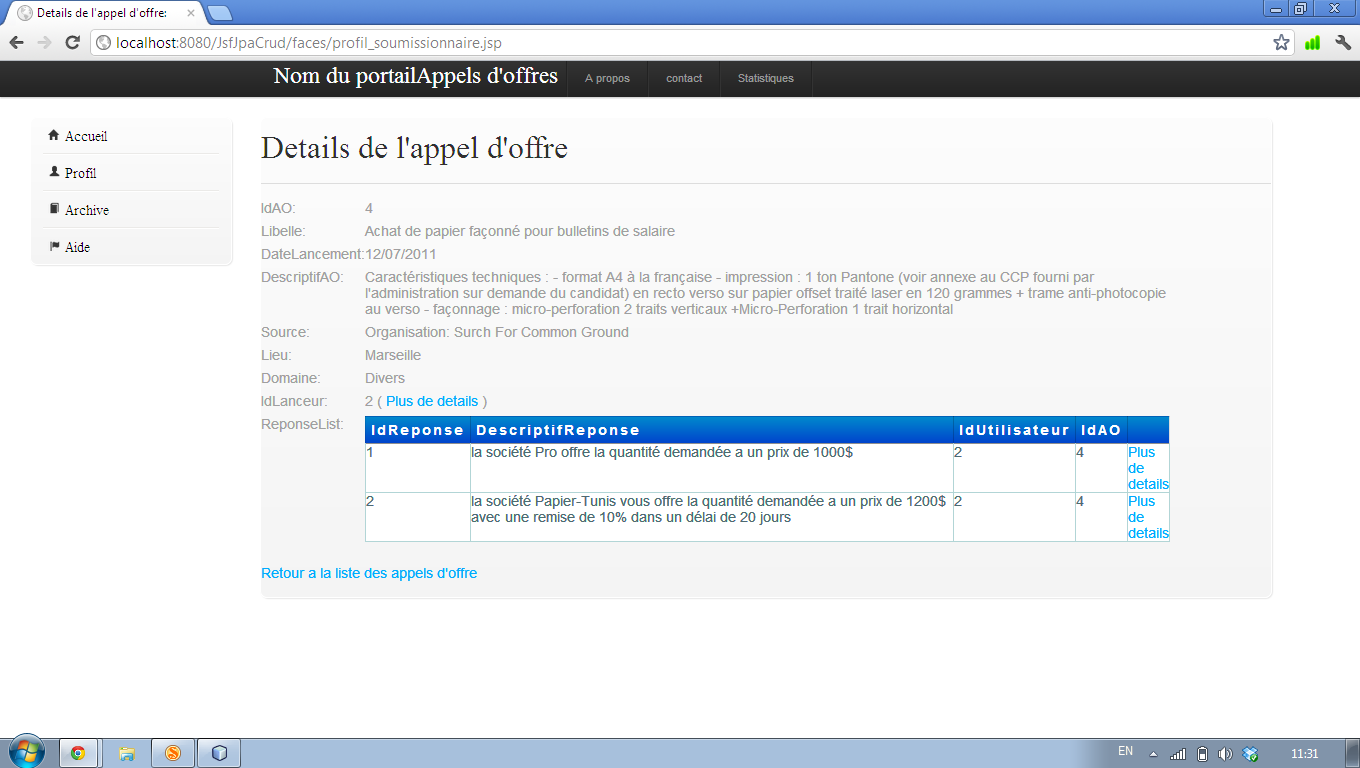
Modification d’un appel d’offre lancé:



**Figure V.8:** interface modification de l’appel d’offre.

Page du profil soumissionnaire:

Un soumissionnaire,tout comme un commanditaire, peut voir tous les appels d'offres et aussi y répondre.( La figure V.6 décrit le profil du soumissionnaire).il peut aussi afficher les details des appels d’offres disponibles.



**Figure V.10:** interface details d’un appel d’offre.

**Conclusion :**

Au cours de ce chapitre nous avons testé notre application, présenté quelques interfaces graphiques et à partir de cette phase nous pouvons modifier l’application selon les besoins des utilisateurs.

Notre stage de fin d’étude s’est inscrit dans le cas de la réalisation d’une application web qui permet la gestion d’appels d’offres.

Ce travail fut d’un intérêt considérable, dans le cas où il nous a permis d’atteindre la majorité des objectifs que nous nous sommes fixés. En effet, ce fut pour nous un passage du cadre théorique vers un environnement réel de travail, afin d’exercer nos connaissances acquises lors de notre cursus universitaire et une initiation à la vie professionnelle.

Ainsi, en effectuant une étude théorique sur les processus de gestion des appels d’offres, nous avons pu constater qu’ils étaient rudimentaires et primitifs ne permettant pas une accessibilité grand-public au différent AO lancé.

Notre application et notre travail méthodique avait pour but de faciliter, de rendre plus aisé et fluide la gestion des appels d’offres aux administrateurs mais aussi aux autres utilisateurs, de lancer et de répondre aux différents AO disponibles.

Pour accomplir notre travail, nous avons suivi la démarche du « processus unifié », à savoir la capture des besoins, l’analyse, la conception, l’implémentation et la réalisation, tout en sachant que nous avons changé ici ou là, quelques aspect de cette démarche à l’endroit où cela nous est paru adéquat.

Et nonobstant, ces changements, nous avons en premier lieu, rassemblé les informations nécessaires à la compréhension du nouveau système et spécifié les fonctionnalités pertinentes de l’application, les utilisateurs, ainsi que les contraintes majeures.

En second lieu, nous avons approfondi la compréhension de ce système, conçus, implémentés et développés les besoins de l’application

Enfin, nous avons testé l’intégralité de notre application et proposé quelques interfaces utilisateurs.

A l’issue de cette réalisation, les fonctionnalités principales ont été énoncées et auparavant ont été implémentées et achevées. Et en termes de perspectives, d’autres modules et fonctionnalités peuvent être ajoutés tel que un module de paiement, un autre permettant la création d’une newsletter et enfin une application mobile.