

Modélisation Objet

Gestion de l'occupation du domaine public

Venditti Chloé et Zerban Yassine

Sommaire :

Page 1 : Situation et contexte de la réalisation du projet

Page 1 : Mise en lumière des fonctionnalités

Page 6/7 : Diagramme de Séquence : Une possible intervention du surveillant des travaux

Page 8/9 : Diagramme d'Activité : comment est géré la facturation pour les Propriétaires ?

Page 9/10 : Diagramme d'États-Transitions : le cycle de vie de la demande

Page 10/14 : Diagramme de classe : Une approche Orientée Objet

Page 15/17 : Résultat, critiques et optimisation du résultat obtenu



I - Situation et contexte de la réalisation du projet

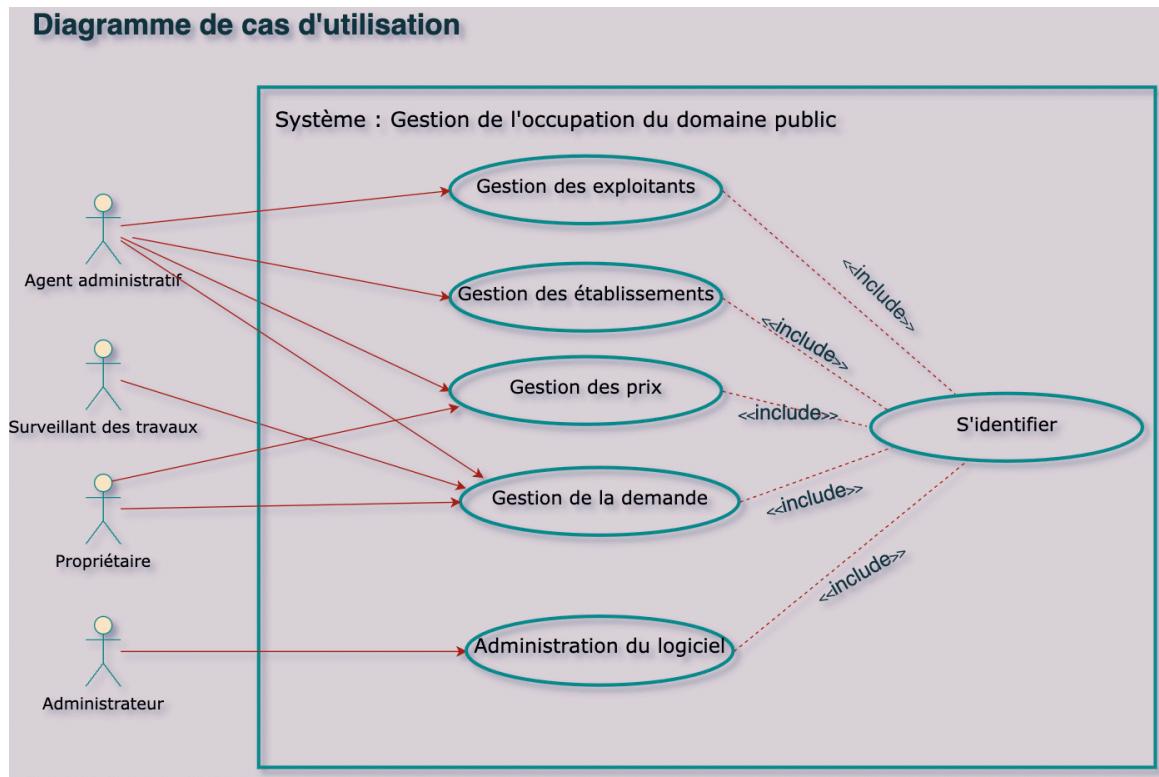
Dans le cadre de l'UE Modélisation Objet nous avons dû réaliser ce projet en binôme durant les séances de TP encadrés par nos enseignantes. Le projet consiste à réaliser une application afin d'aider une mairie avec la gestion de l'occupation du domaine public, c'est à dire, l'aider avec le processus de création de terrasses et/ou d'établissements. Pour réaliser cela, nous nous sommes aidé avec des diagrammes UML créer à partir du sujet donné pour par la suite pouvoir coder en java une fonctionnalité que nous avons choisi de mettre en oeuvre, ici nous avons choisi : Un propriétaire qui possède plusieurs établissement veut consulter sa facture pour une de ses terrasses installée dans l'un de ses établissements.

II - Mise en lumière des fonctionnalités

A- DCU Général

Avant même de commencer à penser au code java à réaliser il faut analyser le projet donné, les données le constituant et en tirer les fonctionnalités majeures. Pour cela, une grande analyse a été faite par nous même, de l'existant de la mairie, c'est à dire comment à l'heure actuelle procède t'elle pour gérer l'occupation du domaine public ? Mais également l'analyse du besoin, de quoi la mairie a t'elle besoin en plus de ce qu'elle a déjà aujourd'hui pour gérer l'occupation du domaine public de sa ville ? Une fois les grandes fonctionnalités trouvées nous avons réalisé un premier DCU (Diagramme de Cas d'Utilisation) Général.

Diagramme de cas d'utilisation



Pour ce premier diagramme nous avons tout d'abord relevé chaque **acteurs** qui sera amener à utilisateur la future application. Nous aurons :

- Agent Administratif
- Surveillant des travaux
- Propriétaire
- Administrateur

Chaque acteur aura au minimum une **grande fonctionnalité** attribuer. Ici, nous avons pu relever 5 grandes fonctionnalités, que nous allons par la suite détaillées une par une.

- Gestion des exploitants
- Gestion des établissements
- Gestion des prix
- Gestion de la demande
- Administration du logiciel

Pour nous, la Gestion des établissement contient également la gestion des terrasses et la Gestion des exploitants correspond à la gestion des propriétaire ayant une/des terrasse(s) et/ou un/des établissement(s).

Les liens parant de nos grandes fonctionnalités vers la fonctionnalité 'S'identifier' correspond à l'obligation de tous les acteurs accédants au logiciel de s'identifier avant de pouvoir accéder aux grande fonctionnalités.

B- DCU Détaillés

a. Gestion de la demande

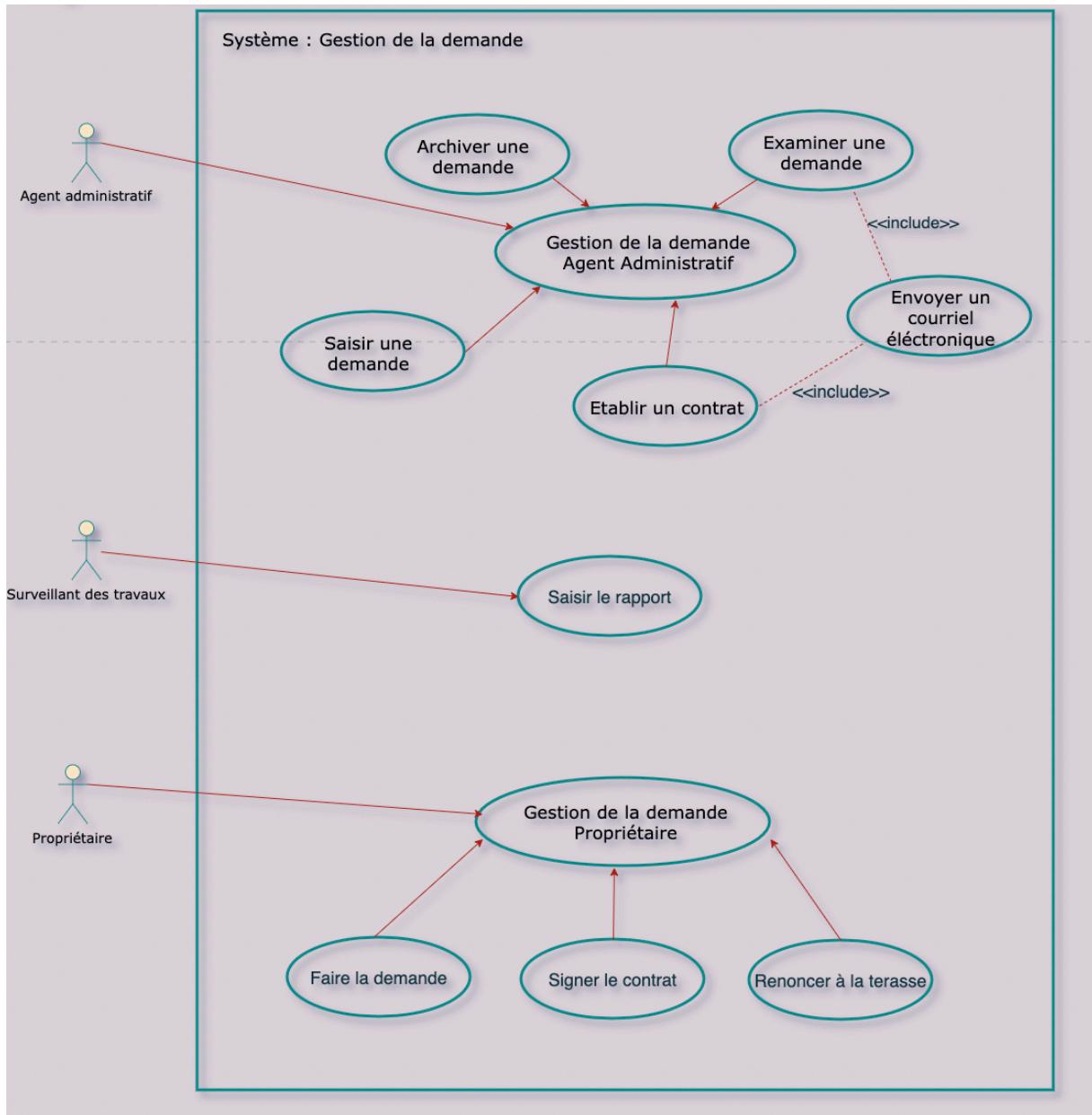
La première grande fonctionnalité que nous avons détaillée dans un DCU est la Gestion de la demande. Elle met en oeuvre 3 acteurs

- Agent Administratif
- Propriétaire
- Surveillant des travaux

Dans le processus de la demande chacun de ces acteurs aura un rôle.

Tout d'abord le **Propriétaire** devra **faire sa demande**, ensuite c'est **l'Agent administratif** qui prendra le relais en **saisissant la demande** faite par le propriétaire, il **l'examinera**, il **établira ensuite le contrat**. quand il sera à ces étapes un **courriel électronique sera envoyé** au propriétaire pour l'informer de l'avancer de sa demande. Le **Surveillant des travaux** effectuera ensuite des vérifications sur site, et une fois cela fait, il **saisira son rapport**. Toutefois, le **propriétaire** peut y **renoncer**, si c'est la cas, alors la demande sera archivée par l'agent administratif.

DCU détaillé Gestion de la demande

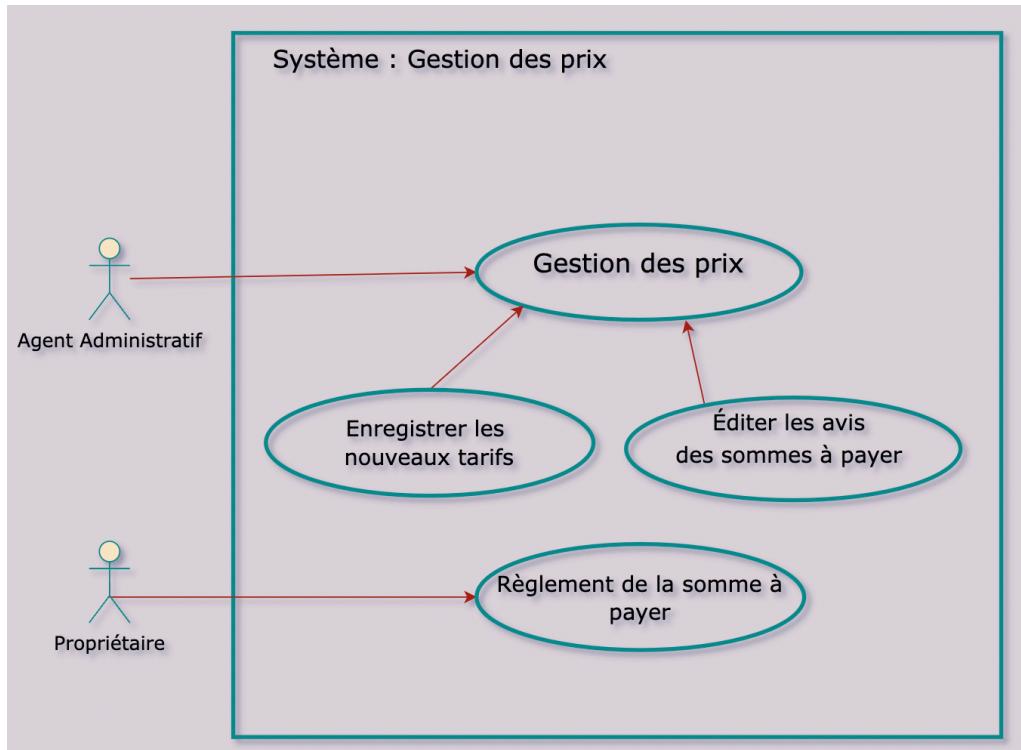


b. Gestion des prix

Ensuite, on a détaillé la fonctionnalité Gestion des prix. Deux acteurs entrent en jeu.

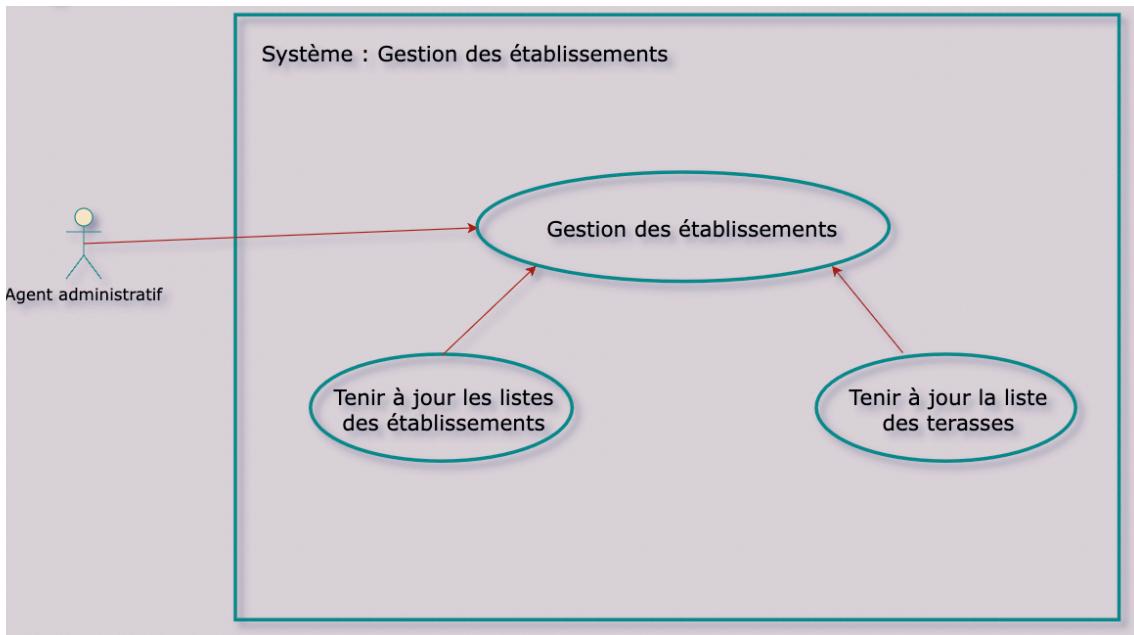
- Agent Administratif
- Propriétaire

L'**agent administratif** doit **enregistrer les nouveaux tarifs** mis en vigueur ou encore **éditer les avis des sommes à payer** par les **propriétaires**, qui eux doivent **régler la somme à payer**.



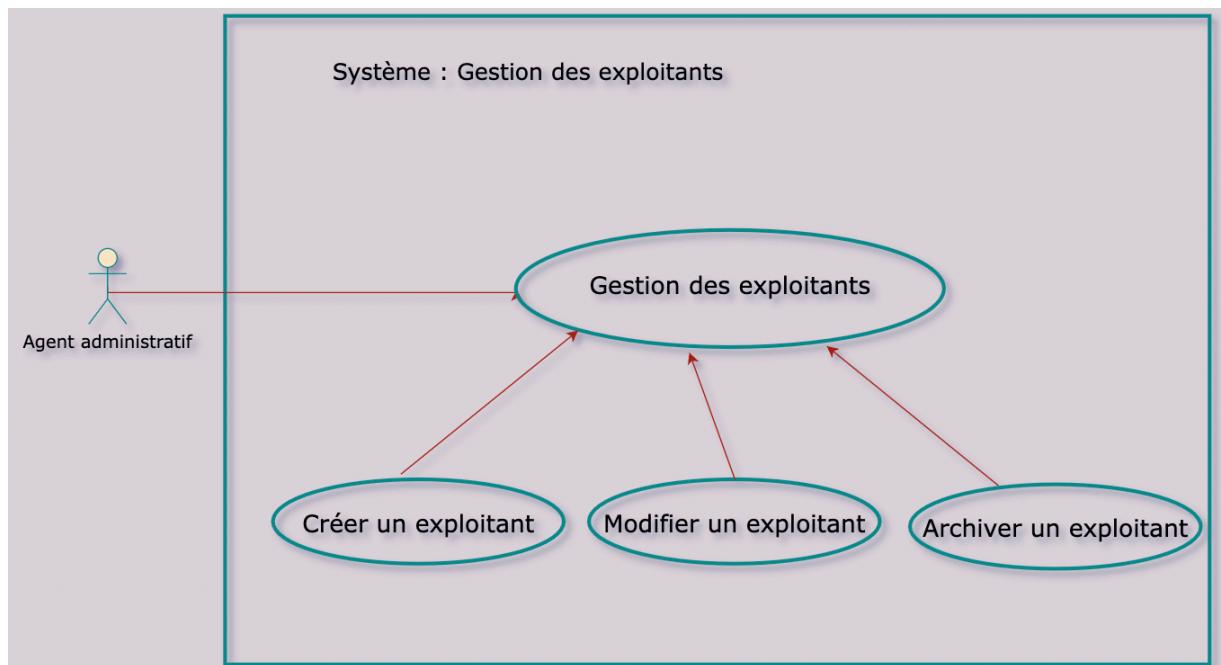
c. Gestion des établissements

Une de nos fonctionnalités les plus importantes est la gestion des établissements. **L'agent administratif** en est en charge. Il doit **tenir à jour la liste des établissements** et également **la liste des terrasses**.



d. Gestion des exploitants

Notre dernière grande fonctionnalité détaillée est la **gestion des exploitants**, par exploitants on entend propriétaire. C'est encore une fois, l'**agent administratif** qui peut **créer un exploitant**, **le modifier** ou **l'archiver**.

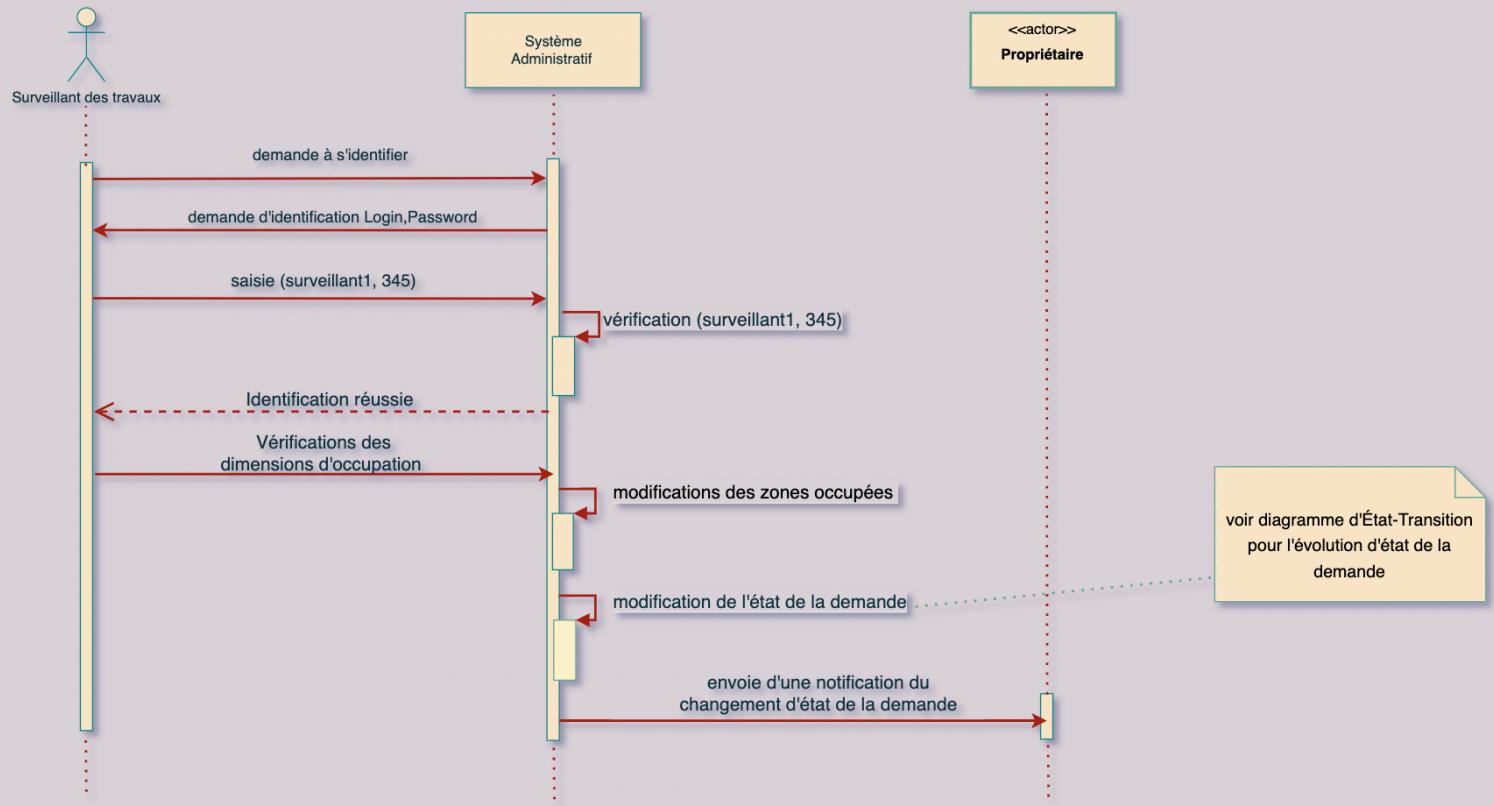


Dans chaque DCU Détailé, nous considérons que les acteurs sont déjà connecté au logiciel. (D'où l'absence de la liaison <<include>> vers identification)

III- Diagramme de Séquence : une possibilité d'intervention du surveillant des travaux

La réalisation d'un diagramme de séquence nous a permis de détaillé l'interaction des acteurs 'Surveillant des travaux' et 'Propriétaire' et le système. Comme vu précédemment le surveillant des travaux peut intervenir lors de la vérification des dimensions d'occupation.

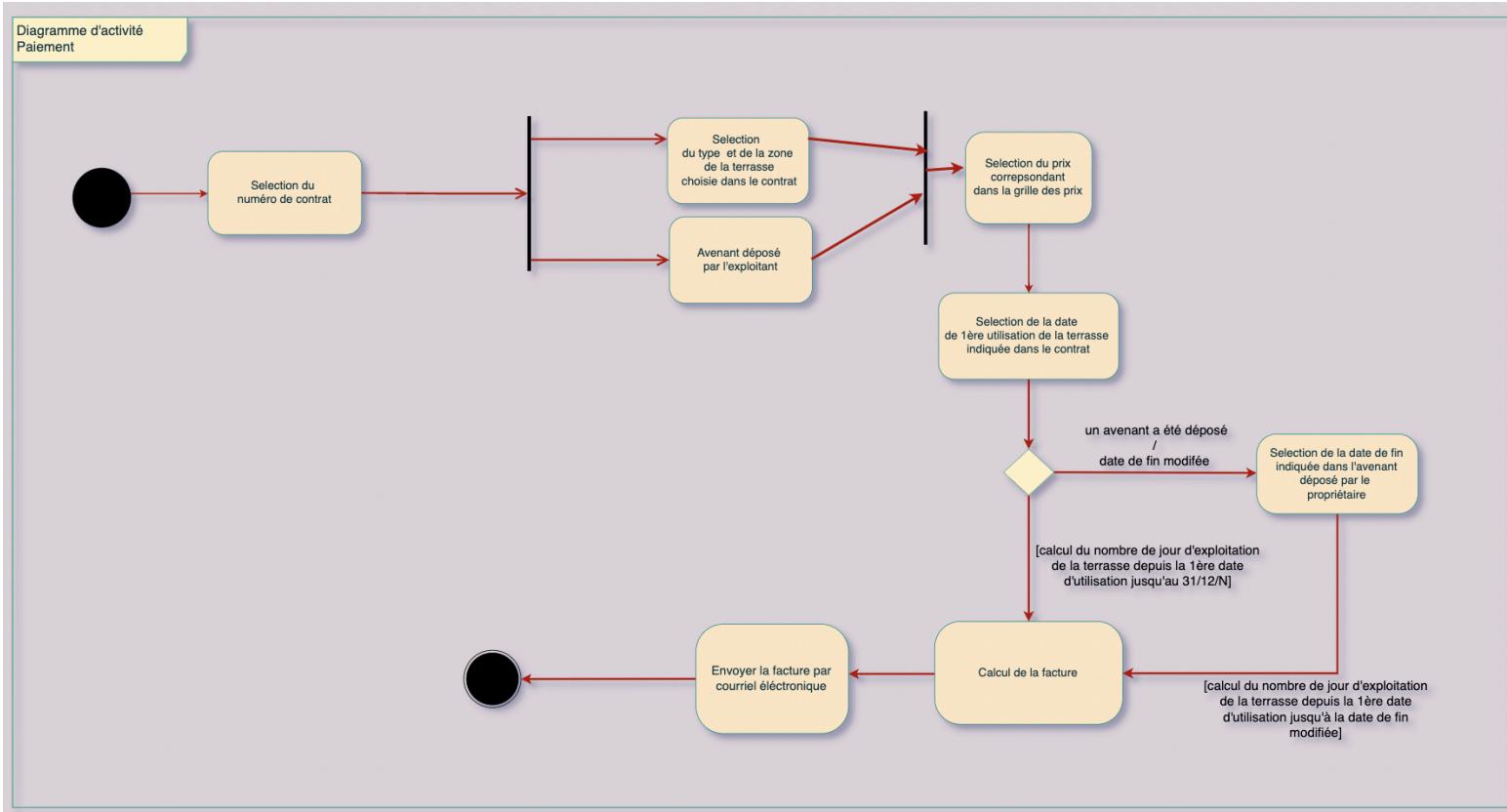
Diagramme de Séquence
Intervention du surveillant des travaux



Une fois la demande faite par le propriétaire et celle-ci examinée par l'agent administratif, le surveillant des travaux n'interviendra que lorsque la demande sera dans l'état 'recevable'. Le surveillant des travaux peut alors intervenir. Ici, on lit le diagramme de séquence réalisé. Une fois avoir rentrer ses informations (login, password) et une fois le système les ayant vérifiés, il pourra commencer à saisir son rapport (après avoir fait des vérifications sur terrain mais cela ne concerne en aucun cas notre logiciel). Il va donc faire une vérification des dimensions d'occupation. Cette vérification aura des conséquences sur, les zones occupées et sur l'état de la demande. En effet, une fois que le surveillant des travaux a fini de saisir son rapport alors, l'état de la demande passe à 'en cours'. (Nous détaillerons cela au diagramme d'États transition à la page 9/10). Une fois la modification de l'état de la demande, une notification sera envoyée au propriétaire pour le lui informer.

III- Diagramme d'Activité : Comment est gérer la facturation pour les Propriétaires ?

Nous avons réalisé un diagramme d'activité pour modéliser le processus de facturation de notre système. C'est un des processus les plus importants à comprendre et à mettre en place.



Evidemment, pour calculer une facture, il faut savoir de quel contrat il s'agit, on va donc sélectionner le contrat concerné puis, y extraire les données suivantes : Le type de terrasses et la zone de l'établissement dans laquelle la terrasse est exploitée. Un avenant peut être déposé par le propriétaire de l'établissement (et donc terrasse). Si le propriétaire en a déposé un, il faut le prendre en compte car il modifiera la date de fin d'exploitation. On peut alors récupérer dans la grille des prix le prix correspondant.

Avec le contrat que nous avons récupéré, on va en sélectionner la date de 1ère utilisation de la terrasse qui y est indiquée. Ici, nous avons deux cas de

Marie de P - Service Emplacements - Cellule 118 - Mail : emplacements@ville-c.fr
OCCUPATION COMMERCIALE DU DOMAINE PUBLIC

TARIFS 2005	
> Terrasse permanente (du 1 ^{er} janvier au 31 décembre - 365 jours)	
Zone	Tarif (en € / m ²)
A	126,50
B	106,36
C	74,74
> Terrasse semi-permanente (du 1 ^{er} avril au 31 octobre - 214 jours)	
Zone	Tarif (en € / m ²)
A	74,16
B	62,36
C	43,82
> Terrasse d'été (du 15 mai au 15 septembre - 123 jours)	
Zone	Tarif (en € / m ²)
A	42,97
B	36,14
C	25,39

Terrasses 2005

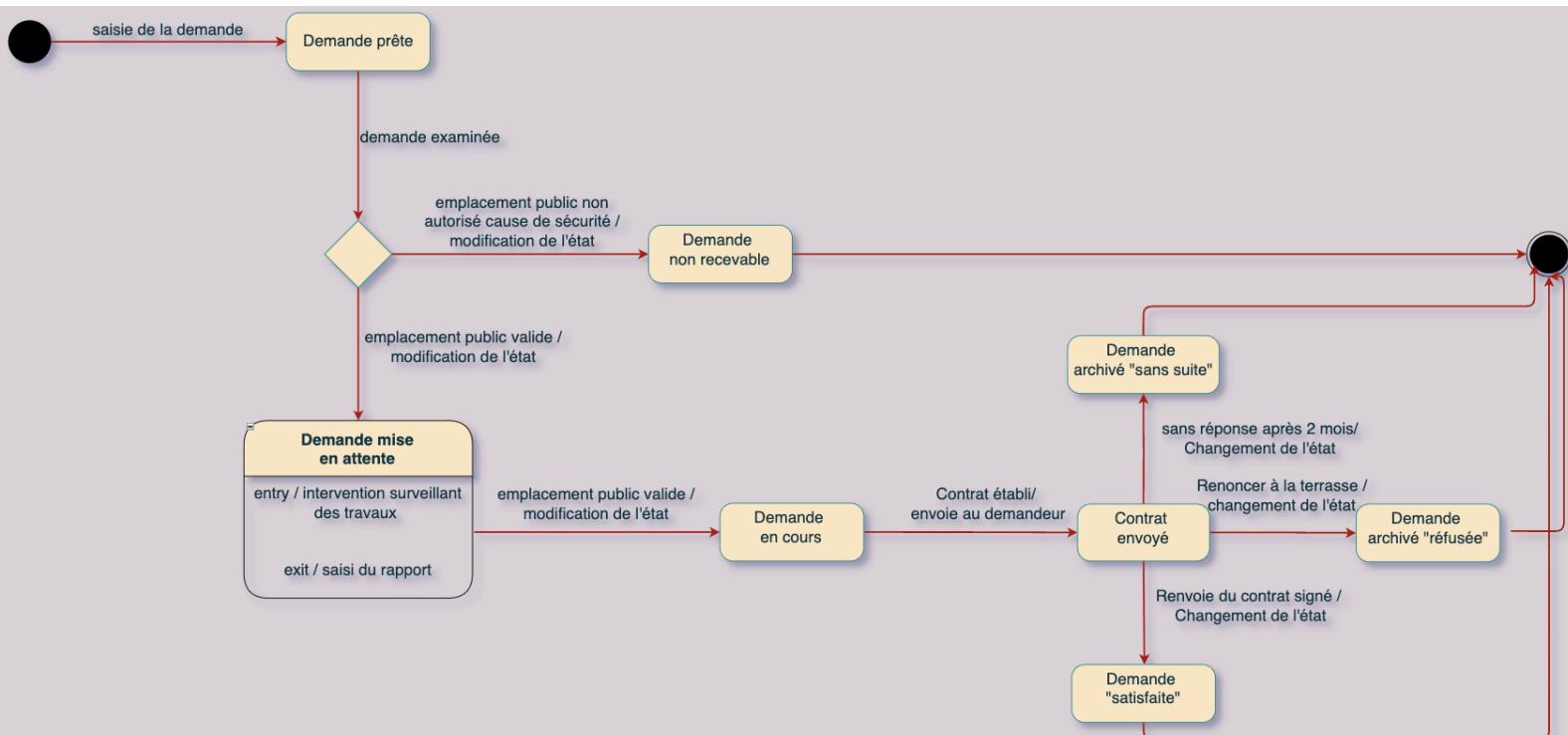
situation :

- Soit aucun avenant n'a été déposé et donc le paiement se fera de la 1ère date d'exploitation jusqu'au 31/12/N (jour de fin + paiement),
- Soit un avenant a été déposé par le propriétaire et donc il faut modifier la date de fin d'exploitation.

Suite au choix de calcul du nombre de jour avec ou sans avenant, on calcul la facture avec le montant à régler pour l'envoyer par courriel électronique au propriétaire de la terrasse qui pourra consulter puis régler sa facture.

IV- Diagramme d'États-Transitions : Le cycle de vie de la demande.

Lors de l'analyse des fonctionnalités on a vu que la gestion de la demande était une fonctionnalité importante. Dans le diagramme D'Activité juste ci-dessus, on a vu qu'avec l'intervention du surveillant des travaux, l'état de la demande changé. Grâce à un diagramme d'État-Transition on va représenté le cycle de vie d'un objet, ici, la demande. Elle est modifiée selon son « État » (8 états).



Le cycle de vie de la demande commence au rond noir qui veut dire 'état initial'. Elle va donc être en premier saisie. Une fois la demande correctement saisie, pour en informer l'agent administratif elle passera à l'état « **Prête** ». L'agent administratif va ensuite l'examinée. La demande

peut être passée à deux états différents selon l'examen mené par l'agent administratif.

- L'emplacement public de la future terrasse (donc de l'établissement actuel) n'est pas autorisé, pour cause de sécurité, alors l'état passe à la demande « **Non recevable** » et on passera à l'état final directement.
- L'emplacement public est valide, l'état de la demande passera donc à l'état « **Mise en attente** ».

Donc, on est dans l'état « Mise en attente » car si la demande est en état « Non recevable », la demande prend fin.

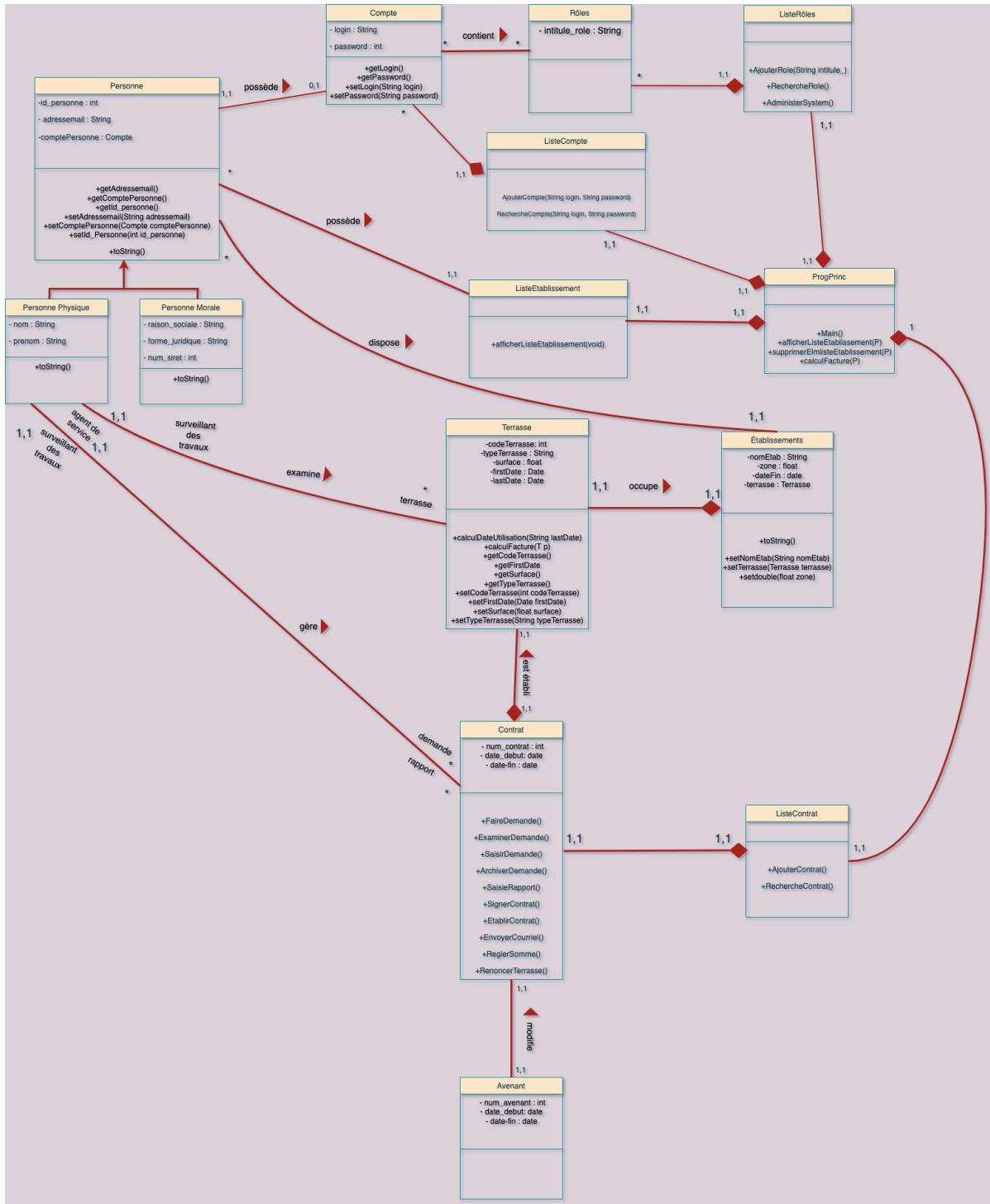
L'état « Mise en attente » nécessite l'intervention du surveillant de travaux puis, une fois la saisi de son rapport on quitte cet état, pour passer à l'état « **En cours** ». Si on est arrivé à cet état c'est que le rapport du surveillant des travaux a bien été saisi. Un contrat va être ensuite établi, puis envoyé au demandeur (propriétaire par exemple). L'état de la demande sera alors à « **Contrat envoyé** ». Ici, trois choix sont possibles.

- Le propriétaire ne donne aucune réponse au contrat envoyé par l'agent administratif sous 2 mois. La demande sera archivée sous l'état « **Sans suite** ».
- Le propriétaire souhaite renoncé à la terrasses, le contrat ne lui convient plus. La demande est archivée avec l'état « **Refusée** ».
- Le propriétaire est satisfait, la demande sera archivée à l'état « **Satisfait** ».

V- Diagramme de Classe : Une approche Orientée Objet

On a identifier toutes les fonctionnalités de notre projet.

Maintenant il faut en faire émergé tous les objets et leurs caractéristiques. L'approche orientée objet considère notre futur logiciel comme une collection de ces objets. Ils auront des caractéristiques, attribut. On notera également les interactions entre ces différents objets. Cette approche nous aidera lors de l'étape du développement, mais, lors du développement du logiciel, on peut être amené à adapter nos diagrammes de classe en fonction de l'évolution du code.



Le diagramme de classe ci dessus est le diagramme de classe détaillé. Avec toutes les classes nécessaires selon nous.

Les classes « Personne Physiques » par exemple un propriétaire avec un nom et prénom et la classe « Personne Morale » une entreprise avec une raison sociale, une forme juridique et un numéro de siret hériteront de la classe « Personne ». Ils auront donc en commun les attributs : id de la personne, une adresse mail et un compte associé. Une personne possède un compte ou non

(obligation d'en posséder un pour faire une demande comme vu sur le diagramme de Cas d'Utilisation). Un compte est donc relié à une unique personne. Une personne peut avoir différent rôle, un propriétaire par exemple peut également être surveillant des travaux ? C'est une possibilité à ne pas négliger, d'où les cardinalités entre les classes « Compte » et « Rôles », '*', qui signifie plusieurs. Pour gérer tous les comptes une classe « ListeCompte » est nécessaire. Si celle-ci vient à être supprimer, les comptes le seront aussi. L'objet compte dépend de cette liste, d'où l'association de composition entre les deux classes. De même que pour les comptes, les rôles sont gérés dans une classe « ListeRoles », encore une fois l'objet rôle dépend de la liste des rôles, d'où l'association de composition. Ces deux classes Listes seront directement relié avec notre classe « ProgPrinc » contenant notre main, si le main n'existe pas les listes non plus, donc les deux classes Liste seront associé avec une composition avec la classe de notre programme principal.

On a donc vu que la classe Personne donner un héritage aux classes « PersonnePhysique » et Morale. La classe « Personne Physique » est associé à la classe « Terrasse ». Effectivement, on a vu grâce aux différents diagrammes précédent que le surveillant des travaux aller sur le terrain pour faire un examen. Donc une association entre « PersonnePhysique » et « Terrasse » avec pour association : le surveillant des travaux qui examine la terrasse. Il peut en examiner plusieurs mais un seul surveillant des travaux est nécessaire pour en examiner une.

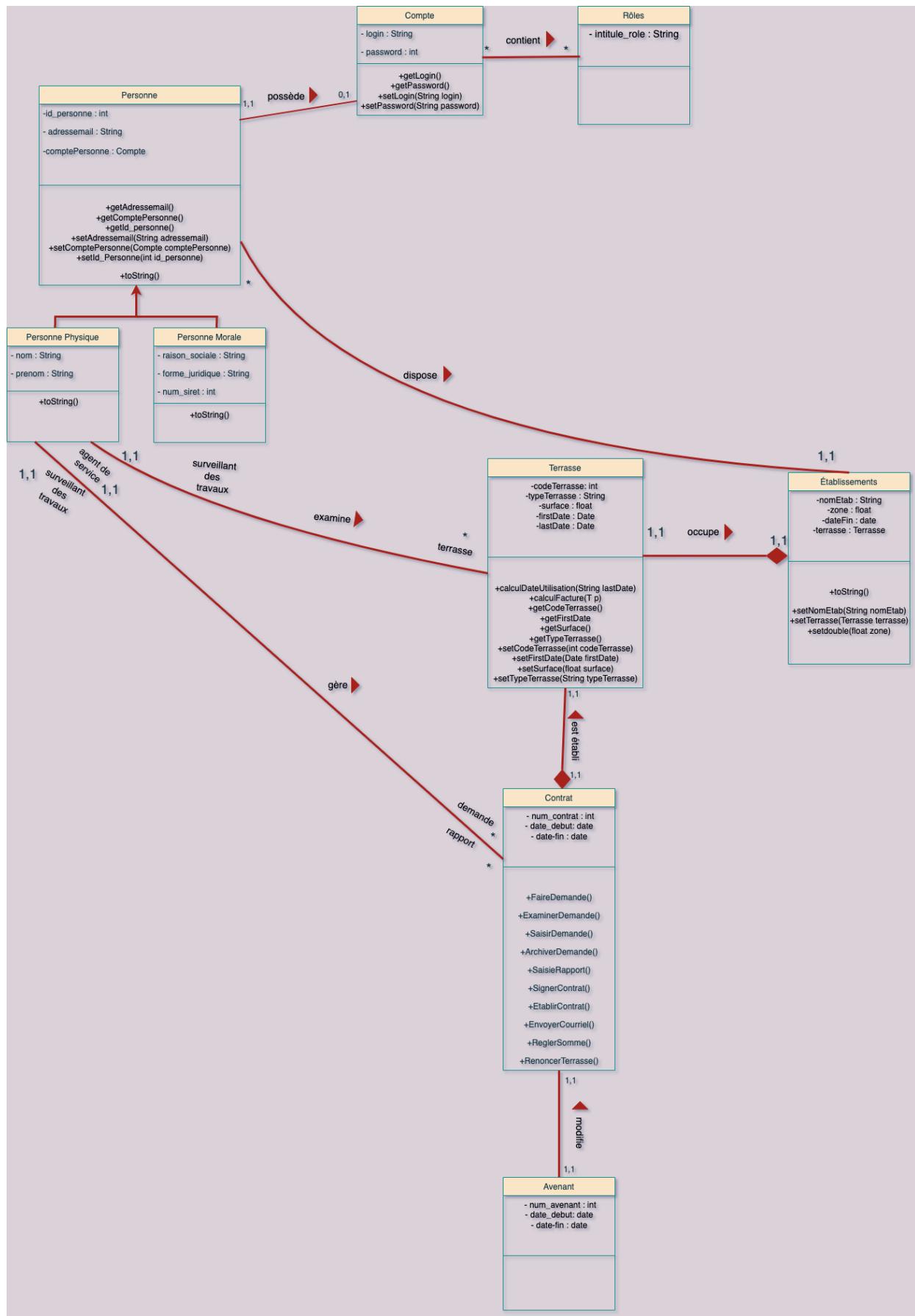
Il peut également saisir un rapport, qui à donc un lien avec la classe « Contrat ». Deux associations sont présentes entre ces deux classes, le surveillant des travaux qui donc, gère son rapport; mais également, l'agent administratif qui gère la demande associé au contrat. Tout comme les rôles ou les comptes le contrat et gérer par une liste, la classe « ListeContrat » la représente. Une association de composition est nécessaire, au même titre que pour

rôles ou comptes et elle sera relié également à la classe « ProgPrinc » avec une association de composition. Grâce au diagramme d'États-Transitions on a vu qu'un avenant pouvait être modifier. Une classe « Avenant » sera donc associé à notre classe « Contrat » qui ira donc modifié celui-ci.

Une terrasse occupe évidemment un établissement, et si celui-ci est supprimé, la terrasse sera inévitablement elle aussi supprimé, une association de composition de plus est présente donc entre la classe « Terrasse » et « Etablissement ». Un établissement est possédé par une Personne, donc cette personne dispose de l'établissement en question. Mais, il est possible qu'une personne puisse avoir plusieurs établissement. Imaginons un propriétaire de deux cafés, chacun des cafés peut avoir une terrasse. Une liste des établissement est nécessaire pour stocké tous les établissements d'une personne. La classe « ListeEtablissement » est donc relié à Personne, celle-ci sera relié à notre programme principal avec une association de composition.

Ce diagramme est donc le diagramme de classe détaillé qui a été construit à partir du diagramme de classe général. Il est relativement similaire au diagramme de classe détaillé, celui-ci étant fait à partir du diagramme de classe général. Les classes représentants les listes ne sont pas présente ainsi que le programme principal.

.



VI- Résultat, critiques et optimisation du résultat obtenu.

Une fois toutes les analyses finies, nous avons pu passer au développement d'une fonctionnalité du projet. Nous avons choisi de travailler sur la fonctionnalités du calcul de la facture. On a aussi permis au code d'afficher la liste des établissements d'un propriétaire et de supprimer un établissement s'il le souhaite. Le résultat obtenu est celui souhaité et attendu. Nous avons voulu un échange entre l'utilisateur et le système en plus. L'utilisateur peut se connecter en tant que personne physique ou morale, puis choisir l'action qu'il veut réaliser.

- Afficher la liste de ses établissements
- Supprimer un établissement de sa liste
- Calculer la facture d'un de ses établissements.

Il choisit la facture. Il doit sélectionner pour quel établissement (donc quelle terrasse), puis l'affichage se fait, les informations sur la terrasse et l'établissement, le nombre de jour exploités et le montant de la facture en euros.

Affichage du menu :

```
Veuillez choisir si vous êtes une personne Morale/Physique
1: Morale,
2: Physique
1
Veuillez saisir votre identifiant
a
Veuillez saisir votre mot de passe
1234
Vous êtes bien connecté !
Id : 1 Raison Sociale: SAARL Forme Juridique: SS Num SIRET: 1234 Adresse: apple@icloud.com Compte: Username: a Password: 1234
1 - Afficher les listes des établissements de l'utilisateur
2 - Supprimer un établissement à l'utilisateur
3 - Demander une facture de terrasse
0 - Appuyer sur 0 pour quitter
```

Affichage de la liste des établissements (contient 1) de la personne morale.

```
Veuillez choisir si vous êtes une personne Morale/Physique
1: Morale,
2: Physique
1
Veuillez saisir votre identifiant
a
Veuillez saisir votre mot de passe
1234
Vous êtes bien connecté !
Id : 1 Raison Sociale: SAARL Forme Juridique: SS Num SIRET: 1234 Adresse: apple@icloud.com Compte: Username: a Password: 1234
1 - Afficher les listes des établissements de l'utilisateur
2 - Supprimer un établissement à l'utilisateur
3 - Demander une facture de terrasse
0 - Appuyer sur 0 pour quitter
1
class projet.PersonneMorale
1 - Nom :Café La Gare Zone: 126.5 Type: Permanente
■
```

Affichage de la facture de la terrasse permanente de 4m² en zone à 126.5 Euros (A) de l'établissement Café La Gare dont le propriétaire est :

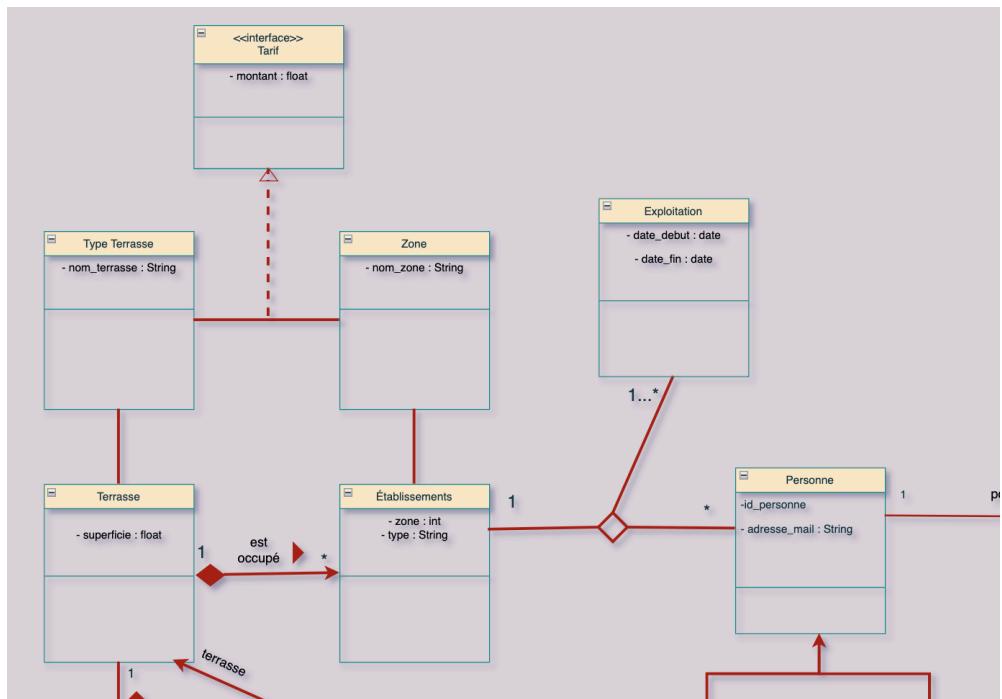
Id :1 Nom :Zerban Prenom :Yassine, yassine@pm.me, login : user1 password : 1234.

```
Veuillez choisir si vous êtes une personne Morale/Physique
1: Morale,
2: Physique
2
Veuillez saisir votre identifiant
user1
Veuillez saisir votre mot de passe
1234
Vous êtes bien connecté !
Id : 1 Nom: Zerban Prenom: Yassine AdresseMail: yassine@pm.me Compte: Username: user1 Password: 1234
1 - Afficher les listes des établissements de l'utilisateur
2 - Supprimer un établissement à l'utilisateur
3 - Demander une facture de terrasse
0 - Appuyer sur 0 pour quitter
3
1 - Nom :Café La Gare Zone: 126.5 Type: Permanente
Choisissez l'établissement possédant la terrasse dont vous voulez la facture
1
_____
Periode occupée : 365.0 jours.
Facture à payer: 506.0 euros.
yassinezerban@Air-de-Yassine projet % ■
```

La période occupée est de 365 jours et la facture est donc de 506 euros. (4x126.5 = 506euros).

Nous avons une critique à apporter à notre travail. Nous avions dessiner un diagramme de classe détaillé, mais, lors du développement du logiciel, nous avons du le modifier, car nous

avons rencontré des problèmes concernant les classes initiales « Zone », « Période d'exploitation ».



Voici la partie initiale, qui a donc été modifiée. Nous avons rencontré des problèmes lors de codage au niveau de l'association ternaire avec « Etablissement », « Personne » et « Exploitation ». Nous avons donc décidé que la classe « Terrasse » contiendrait cela, ainsi que le type de terrasse et que la classe « Etablissement » contiendrait la zone. L'interface initialement prévue « Tarif » n'a pas été faite, la tarif, donc la facture est géré dans la classe « Terrasses » en fonction du calcul de la date d'exploitation et des informations de la terrasse et de l'établissement. C'est un changement qui n'était pas prévu mais nous avons adapté notre diagramme en fonction de l'avancé du développement de notre logiciel.

Concernant l'optimisation du résultat obtenu, une interface graphique est à imaginer pour l'étape supérieure, et une amélioration du code JAVA, ainsi que des classes présentes.