

Rapport d'itération 2

|  |  |
| --- | --- |
| **Laboratoire** | 02 |
| **Étudiants** | Dominic Roberge  André Koolen  Anthony Martin Coallier |
| **Codes permanents** | ROBD03129209  KOOA23039101  MARA19129107 |
| **Cours** | LOG210 |
| **Session** | Été 2015 |
| **Groupe** | 03 |
| **Professeur** | Yvan Ross |
| **Chargé de laboratoire** | Charbonneau, Philippe |
| **Date** | 08-06-2015 |

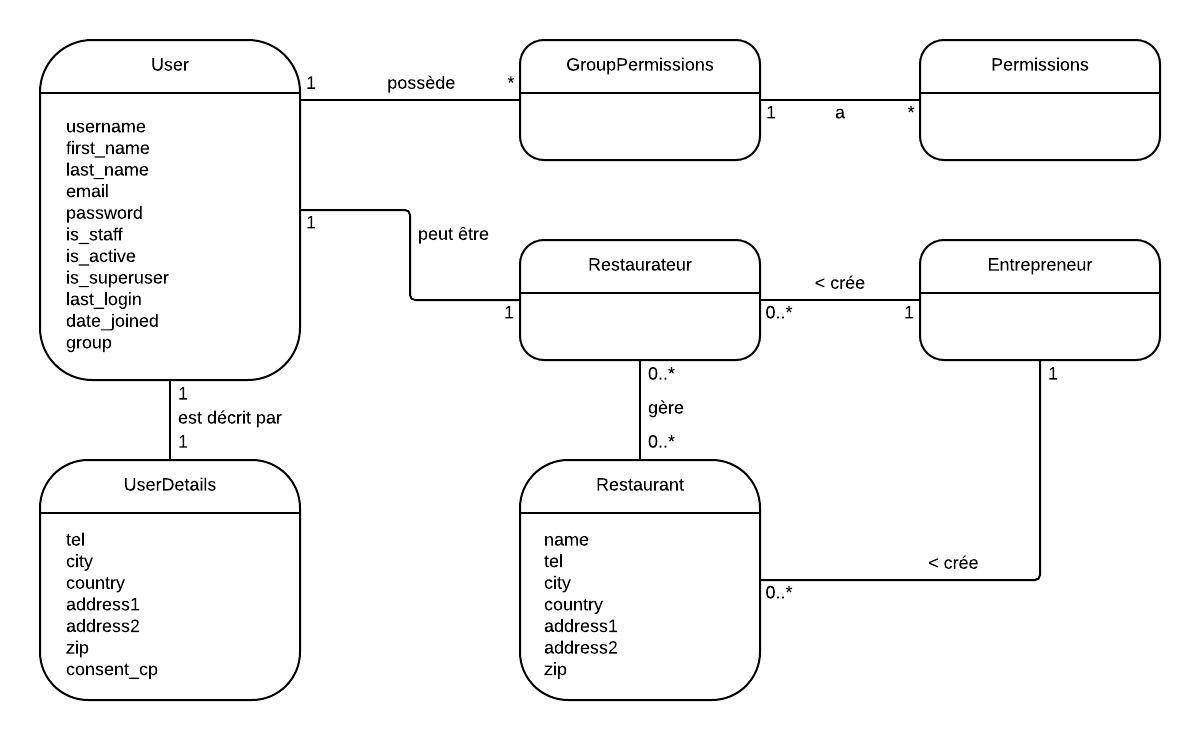
Table des matières

Introduction

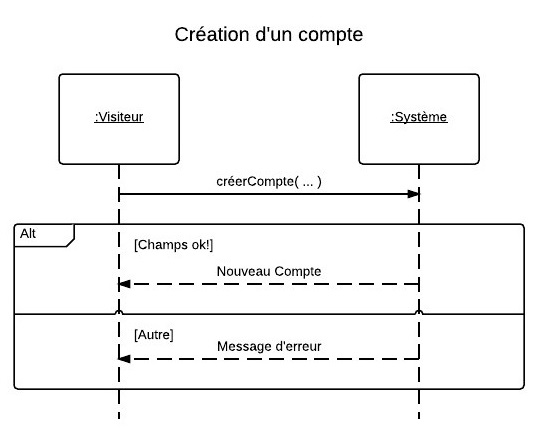
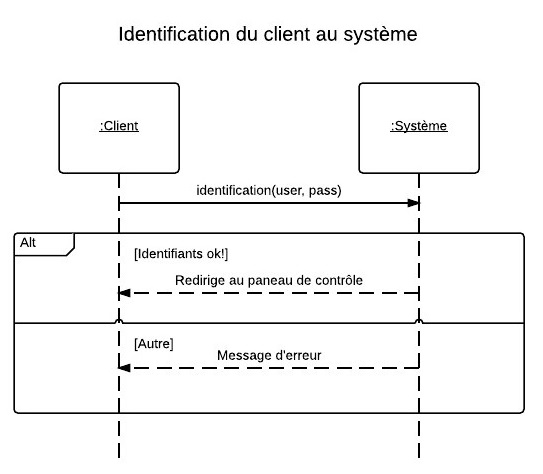
Durant la dernière itération, nous avons préparé le projet en installant Django, appliqué le CSS de Bootstrap 3, etc. De cette structure, nous avons l'implanté la création et la gestion des comptes des utilisateurs, visiteurs.

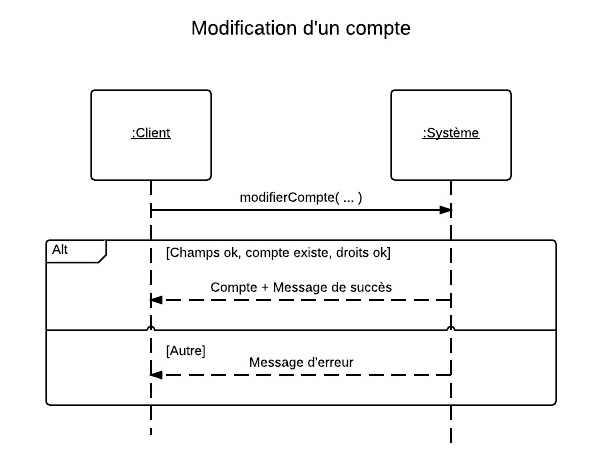
Maintenant, nous entamons la deuxième itération qui portera principalement sur la gestion des restaurants et des restaurateurs. Nous devons également corriger l'identifiant des visiteurs lors de la connexion et une amélioration de l'interface sera apporté.

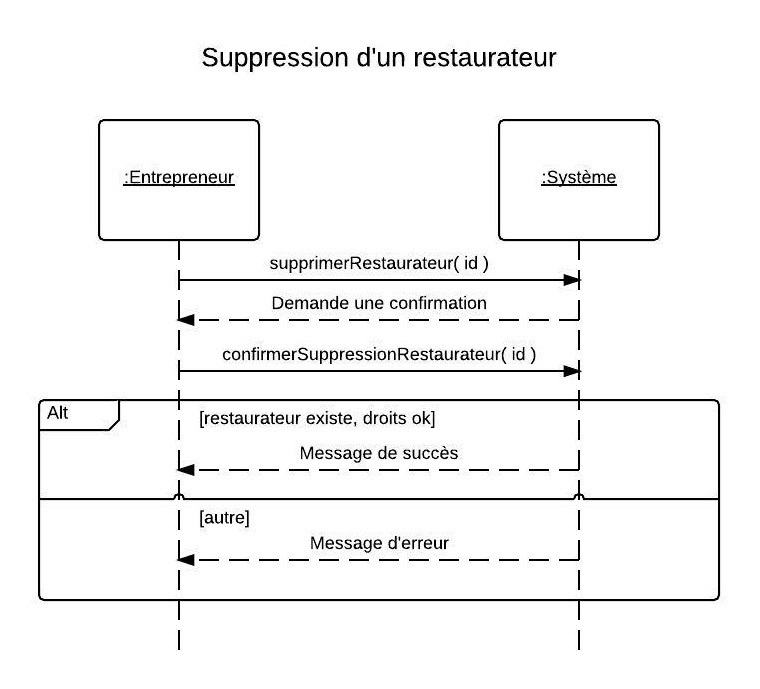
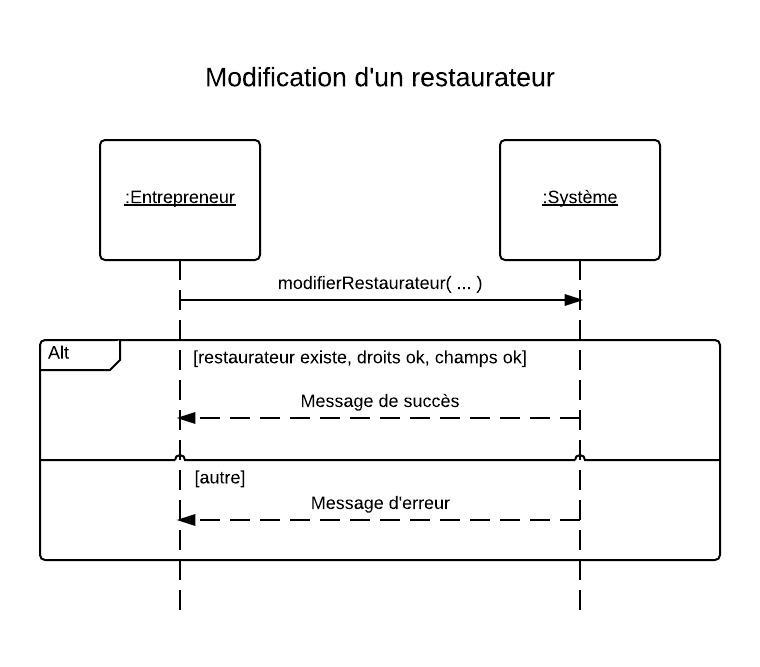
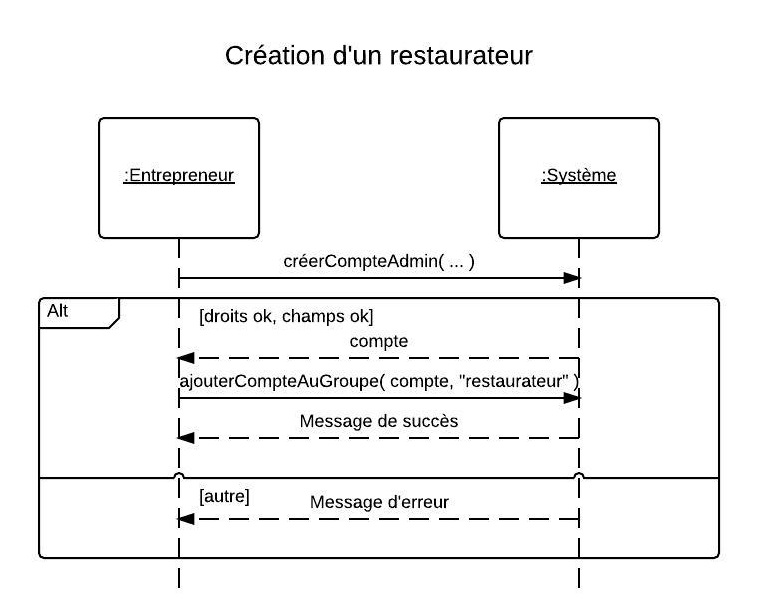
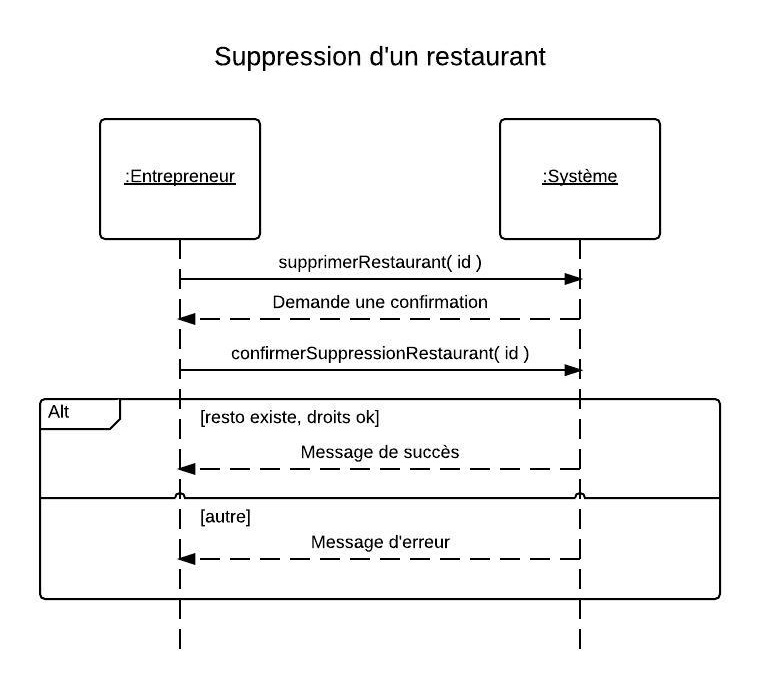
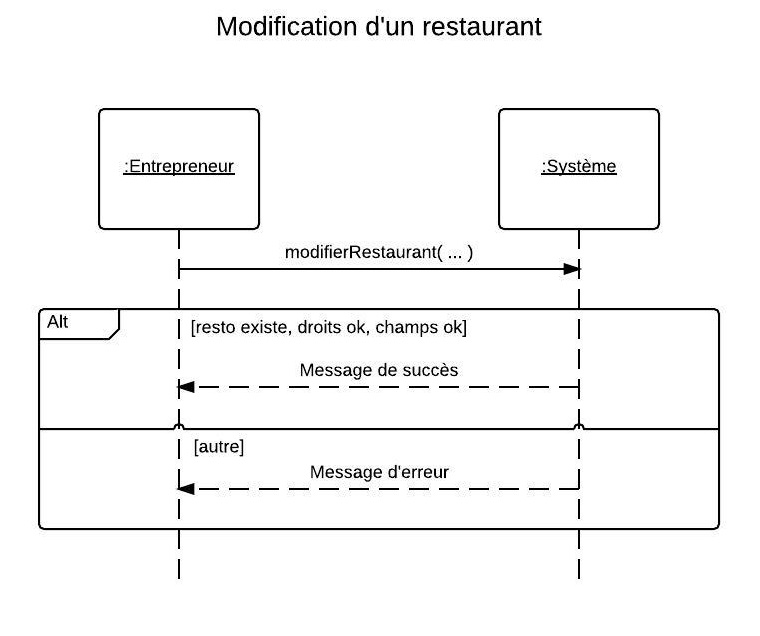
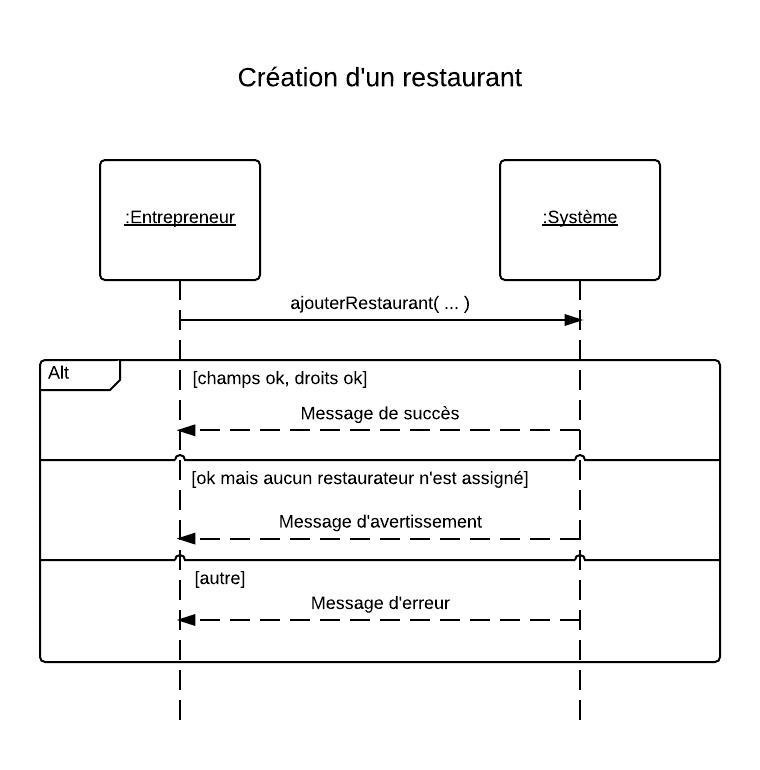
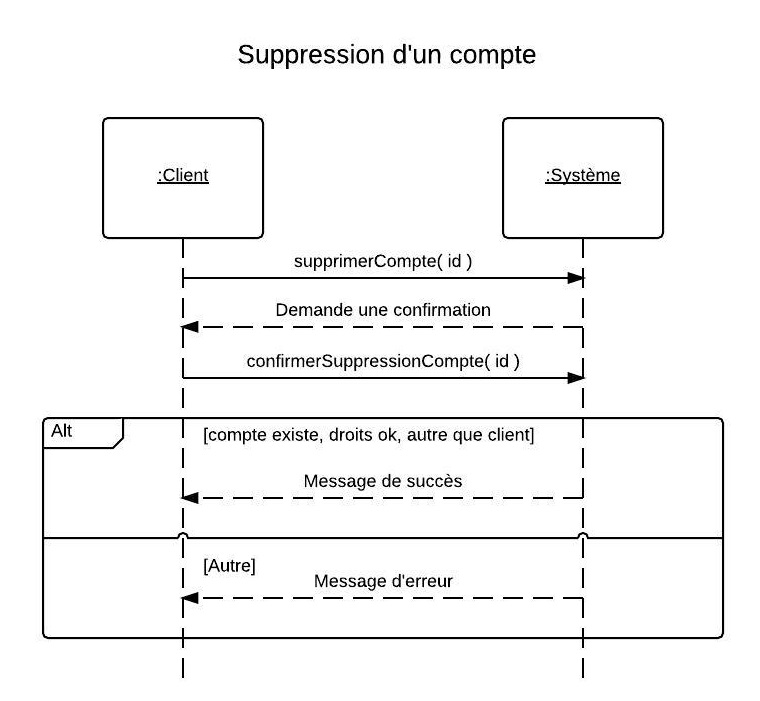
Modèle du domaine



Diagrammes de séquence système







Contrats d'opérations

|  |  |
| --- | --- |
| **Opération** | identification(user:String, pass:String) |
| **Référence croisée** | Identification du client |
| **Préconditions** | Le compte existe |
| **Postconditions** | * Un objet Session a été créé * Attribut de l'objet User last\_login a été modifié |

|  |  |
| --- | --- |
| **Opération** | créerCompte(user:String, pass:String, nom:String, prénom:String, courriel:String, tel:String, city:String, country:String, address1:String, address2:String, zip:String, consent:Boolean) |
| **Référence croisée** | Création d'un compte |
| **Préconditions** | Une visite du site est en cours |
| **Postconditions** | * Un objet User a été créé * Un objet UserDetails a été créé * Une association entre le User et son UserDetails a été créé * User.username est devenu user * User.password est devenu pass * User.last\_name est devenu nom * User.first\_name est devenu prénom * User.email est devenu courriel * User.tel est devenu tel * User.city est devenu city * User.country est devenu country * User.address1 est devenu address1 * User.address2 est devenu address2 * User.zip est devenu zip * User.consent\_cp est devenu consent |

|  |  |
| --- | --- |
| **Opération** | modifierCompte(id: integer, user:String, pass:String, nom:String, prénom:String, courriel:String, tel:String, city:String, country:String, address1:String, address2:String, zip:String) |
| **Référence croisée** | Modification d'un compte |
| **Préconditions** | * Le compte existe et le client est authentifié * Le client est sur la page de ce formulaire |
| **Postconditions** | * Les attributs du User ont été modifiés * User.username est devenu user * User.password est devenu pass * User.last\_name est devenu nom * User.first\_name est devenu prénom * User.email est devenu courriel * User.tel est devenu tel * User.city est devenu city * User.country est devenu country * User.address1 est devenu address1 * User.address2 est devenu address2 * User.zip est devenu zip |

|  |  |
| --- | --- |
| **Opération** | supprimerCompte(id:Int) |
| **Référence croisée** | Suppression d'un compte |
| **Préconditions** | * Le compte existe |
| **Postconditions** | Aucune |

|  |  |
| --- | --- |
| **Opération** | confirmerSupression(id:Int) |
| **Référence croisée** | Suppression d'un compte |
| **Préconditions** | * Le User existe * Le client est authentifié |
| **Postconditions** | * L'attribut de User "is\_active", a été mis à False   *NB. Django déconseille la suppression d'utilisateur pour l'effet de suppression en Cascade des éléments qui y sont associés* |

|  |  |
| --- | --- |
| **Opération** | ajouterRestaurant(name:String, tel:String, city:String, country:String, address1:String, address2:String, zip:String, restaurateur:Restaurateur) |
| **Référence croisée** | Création d'un restaurant |
| **Préconditions** | * L'entrepreneur est identifié comme super admin |
| **Postconditions** | * Création d'un objet Restaurant * Association facultative d'un Restaurant avec un User du groupe "Restaurateur" * Restaurant.name est devenu name * Restaurant.tel est devenu tel * Restaurant.city est devenu city * Restaurant.country est devenu country * Restaurant.address1 est devenu address1 * Restaurant.address2 est devenu address2 * Restaurant.zip est devenu zip |

|  |  |
| --- | --- |
| **Opération** | modifierRestaurant(id: integer, name:String, tel:String, city:String, country:String, address1:String, address2:String, zip:String, restaurateur:Restaurateur) |
| **Référence croisée** | Modification d'un restaurant |
| **Préconditions** | * L'entrepreneur est identifié comme super admin * Le restaurant existe |
| **Postconditions** | * Les attributs du Restaurent sont modifiés * Association facultative du Restaurant avec un User du groupe "Restaurateur" * Restaurant.name est devenu name * Restaurant.tel est devenu tel * Restaurant.city est devenu city * Restaurant.country est devenu country * Restaurant.address1 est devenu address1 * Restaurant.address2 est devenu address2 * Restaurant.zip est devenu zip |

|  |  |
| --- | --- |
| **Opération** | supprimerRestaurant(id:Integer) |
| **Référence croisée** | Suppression d'un restaurant |
| **Préconditions** | * L'entrepreneur est identifié comme super admin * Le restaurent existe |
| **Postconditions** | Aucune |

|  |  |
| --- | --- |
| **Opération** | confirmerSuppressionRestaurant(id:Integer) |
| **Référence croisée** | Suppression d'un restaurant |
| **Préconditions** | * L'entrepreneur est identifié comme super admin * Le restaurent existe |
| **Postconditions** | * Le Restaurant est supprimé |

|  |  |
| --- | --- |
| **Opération** | créerCompteAdmin(username:String, password1:String, password2:String) |
| **Référence croisée** | Création d'un restaurateur |
| **Préconditions** | * L'entrepreneur est identifié comme super admin |
| **Postconditions** | * Un objet User a été créé   *NB. Le module Admin de Django n'associe pas les droits d'accès à l'utilisateur par défaut. L'entrepreneur est redirigé vers la modification du compte (opération* ***modifierRestaurateur****) et dois cocher "* *Statut équipe".*   * Un objet Restaurateur a été créé * Une association entre User et Restaurateur a été créée * Restaurateur.username est devenu username * Restaurateur.password est devenu password1 |

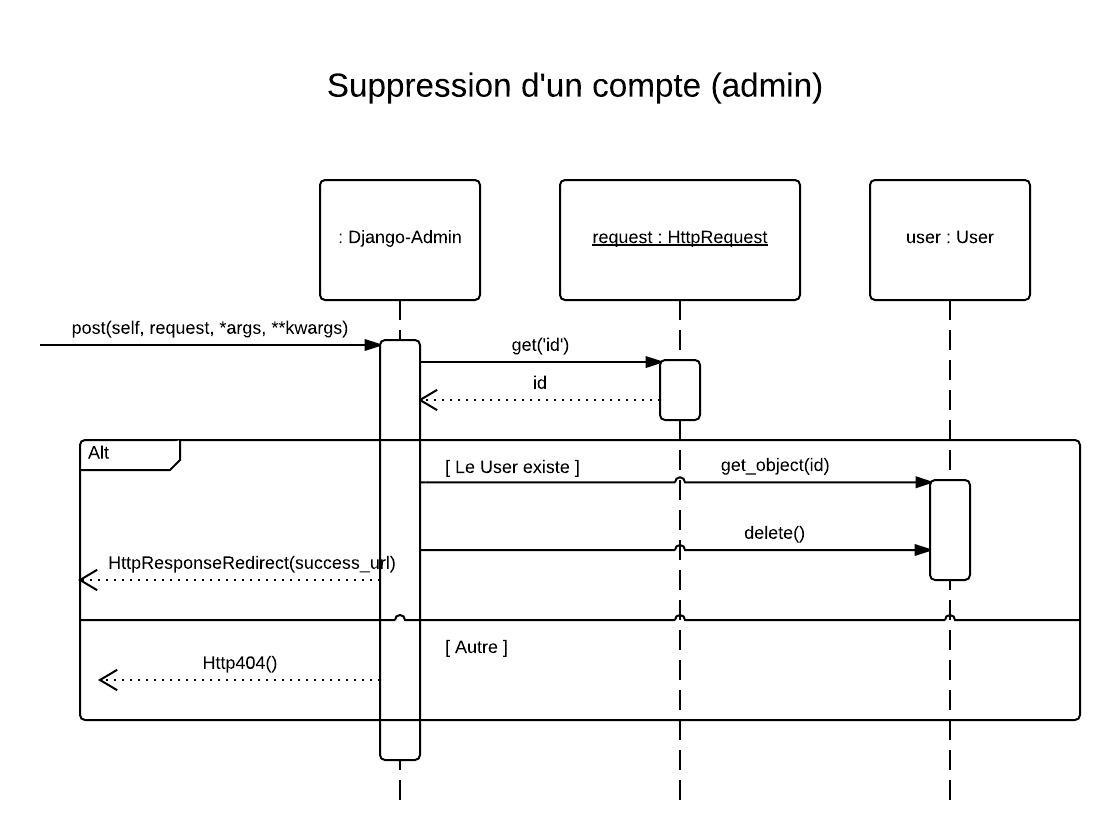
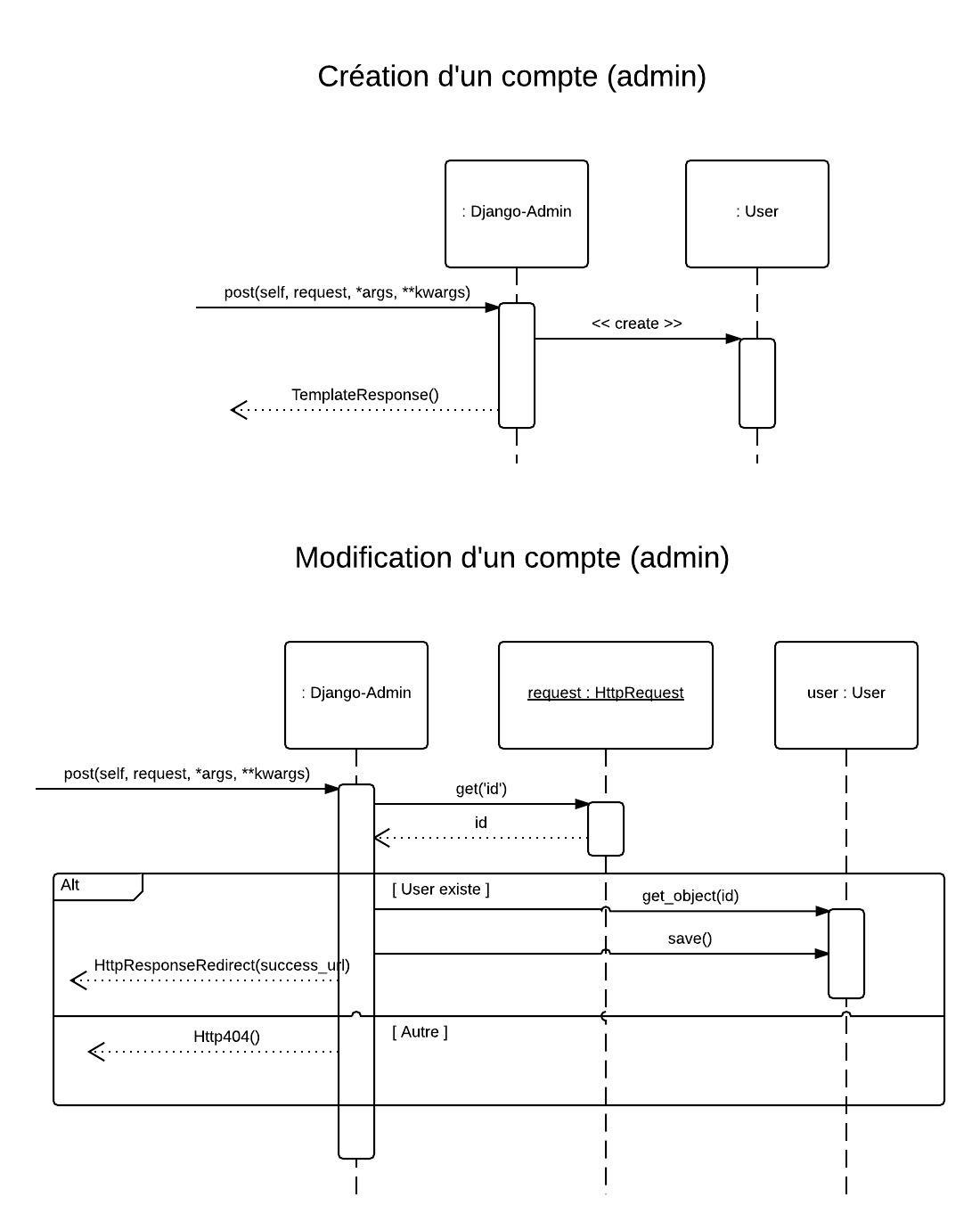
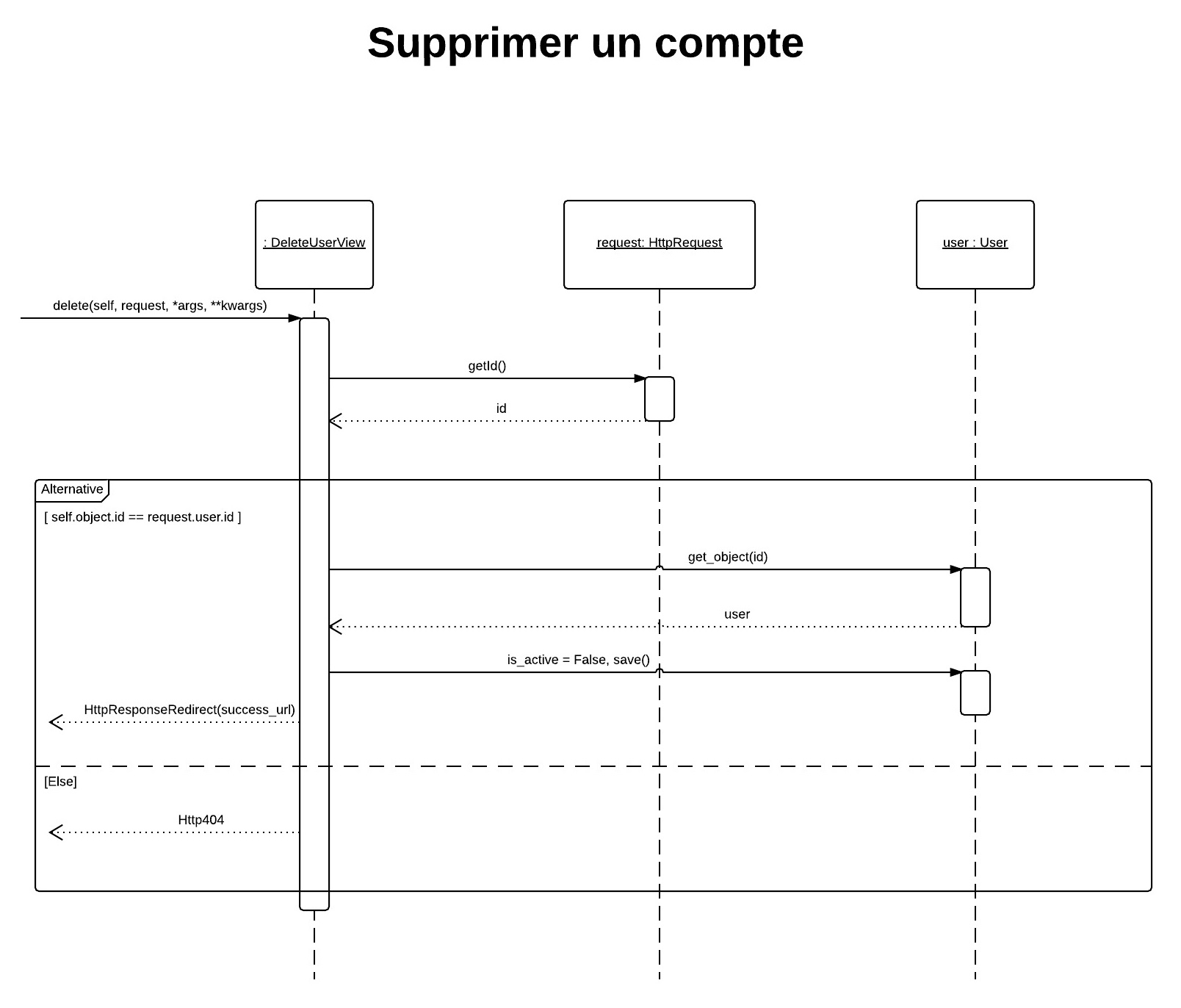
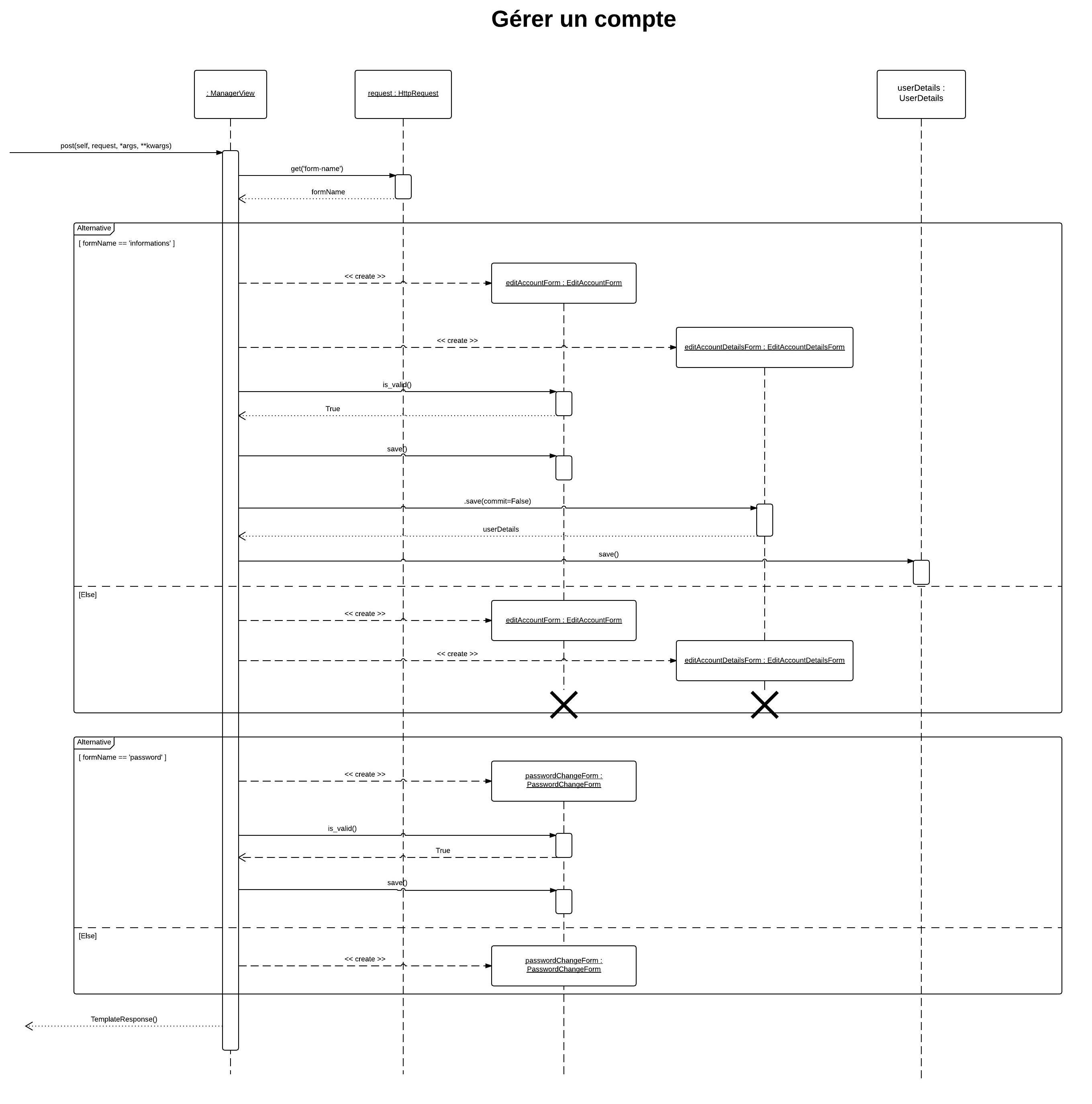
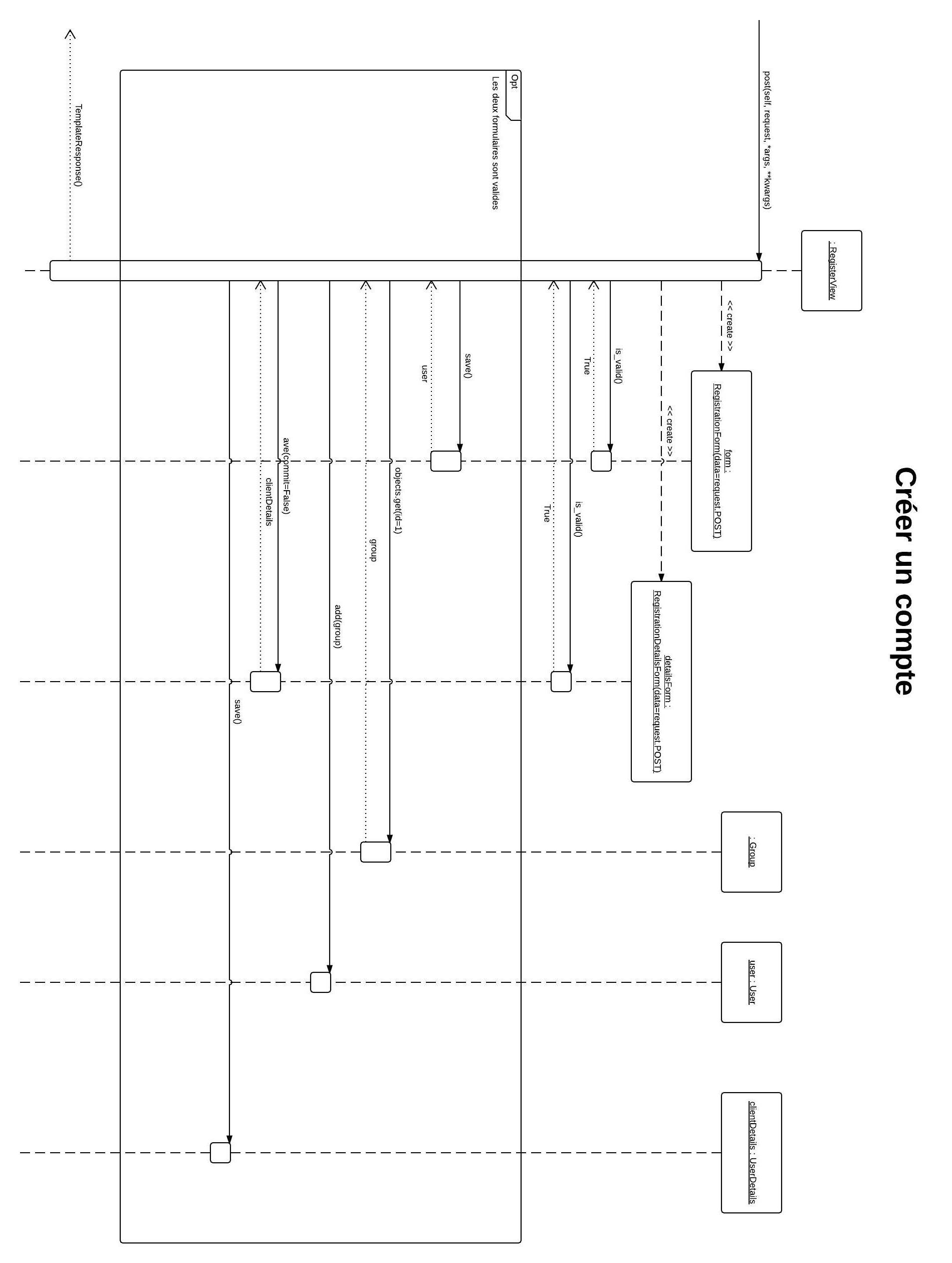
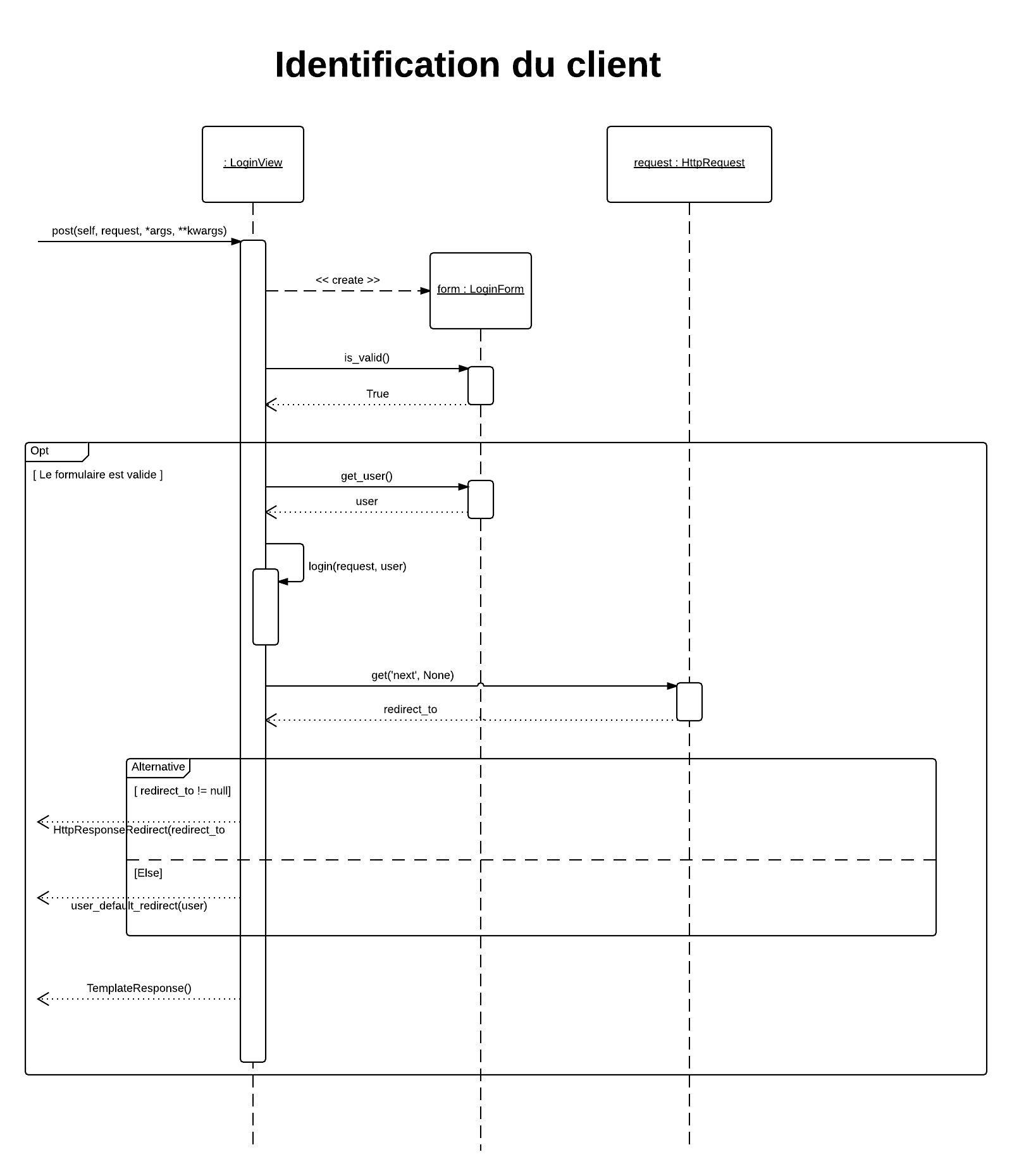
|  |  |
| --- | --- |
| **Opération** | ajouterCompteAuGroupe(id:Integer, groupe:String) |
| **Référence croisée** | Création d'un restaurateur |
| **Préconditions** | * L'entrepreneur est identifié comme super admin * Le User existe |
| **Postconditions** | * Une association a été créée entre l'instance de Groupe "restaurateur" et le User concerné |

|  |  |
| --- | --- |
| **Opération** | modifierRestaurateur(id:Integer, username:String, password1:String, password2:String, first\_name:String, last\_name:String, email:String, is\_staff:Boolean, is\_active:Boolean) |
| **Référence croisée** | Modification d'un restaurateur |
| **Préconditions** | * L'entrepreneur est identifié comme super admin * Le User existe |
| **Postconditions** | * Restaurateur.username est devenu user * Restaurateur.password est devenu password1 * Restaurateur.last\_name est devenu nom * Restaurateur.first\_name est devenu prénom * Restaurateur.email est devenu courriel * Restaurateur.is\_staff est devenu is\_staff * Restaurateur.is\_active est devenu is\_active |

|  |  |
| --- | --- |
| **Opération** | supprimerRestaurateur(id:Integer) |
| **Référence croisée** | Suppression d'un restaurateur |
| **Préconditions** | * L'entrepreneur est identifié comme super admin * Le Restaurant existe |
| **Postconditions** | Aucune |

|  |  |
| --- | --- |
| **Opération** | confirmerSuppressionRestaurateur(id:Integer) |
| **Référence croisée** | Suppression d'un restaurateur |
| **Préconditions** | * Une demande d'annulation est en cours |
| **Postconditions** | * Le Restaurant a étéest supprimé |

Diagrammes de séquence



Conclusion

Durant cette itération, nous avons implémenté les fonctionnalités nécessaires pour la gestion et la création de compte restaurant et celle d'un restaurateur. Une association entre restaurant et restaurateur est possible soit lors de la création d'un des comptes ou lors de modification dans l'un des deux comptes, mais seul l'entrepreneur possède les droits pour cette modification. Le tout est fonctionnel par le module administratif de Django. Nous avons également entamé le travail sur le design *front-end* du produit pour les visiteurs.

La prochaine itération qui commencera à partir du 15 juin traitera de la gestion des menus dans le module pour les restaurants ainsi que d'autres points qui seront décidés lors de la planification.