

Dossier Technique

Je*i* Time

Gestion de la saisie des heures

Par Sandra Trino

CFPA Bordelongue Juillet 2009

Sommaire

I) Définition du cahier des charges.....	3
1) Présentation de l'association JEInnov.....	3
2) Définition d'une JEI (Jeune Entreprise Innovante).....	3
3) Présentation du cahier des charges.....	3
4) Schéma descriptif de la conception de l'application.....	3
II) Diagrammes des Uses Cases, règles de gestion et d'organisation.....	4
1) Diagramme général des Uses Cases.....	4
2) Diagramme des sous-scénarios.....	4
Sous scénario : Gestion des heures.....	4
Sous-scénario : Gestion des Projets.....	5
Sous-scénario : Gestion des Utilisateurs.....	5
Sous-scénario : Bilan.....	6
Sous-scénario : Organisation du temps de travail.....	6
3) Présentation du tableau des droits des différents profils.....	7
4) Règles de gestion et d'organisation	8
Gestion de Projet :	8
Gestion des Utilisateurs.....	8
Les bilans.....	9
Gestion des Heures :	9
Organisation du temps de travail.....	9
III) La Base de données.....	11
1) MCD de Vect'Heures.....	11
2) Passage de l'une à l'autre.....	11
3) MCD de JEI Time	12
IV) Description phase de développement.....	13
1) Technologies utilisées.....	13
Les technologies :	13
Présentation du framework OpenSuit :	13
2) Tableau récapitulatif des itérations.....	17
V) Diagrammes de conceptions détaillées.....	18
1) Description de l'architecture de l'application	18
2) Diagrammes de classes.....	19
3) Enchaînement d'écrans.....	21
Schéma montrant toutes les pages et leurs liens dans le menu.....	21
Captures d'écran de la page d'accueil pour les différentes profils.....	22
Annexes 1	26
Ensemble des diagrammes de séquence détaillée.....	27
Gestion Projet.....	27
Gestion des utilisateurs	37
Bilans.....	39
Gestion des heures	42
Organisation du temps de travail :	43

I) Définition du cahier des charges

1) Présentation de l'association JEInnov

L'application JEI Time a été développée dans le cadre de l'association JEInnov. Cette association regroupe des Jeunes entreprises spécialisées dans la recherche (JEI jeune entreprise innovante). Ses buts sont d'aider les JEI, les promouvoir et les aider dans les différentes démarches qu'elles peuvent rencontrer.

2) Définition d'une JEI (Jeune Entreprise Innovante)

Pour avoir le statut JEI, il faut réunir différents critères qui sont :

- Etre une PME (de moins de 250 salariés)
- Exister depuis moins de 8 ans
- Consacrer une partie de l'activité à la recherche (appliquée, expérimentale, fondamentale).

3) Présentation du cahier des charges

Une première application Vect'Heures fut développée par une bio-informaticienne appartenant à l'une des entreprises membre de l'association.

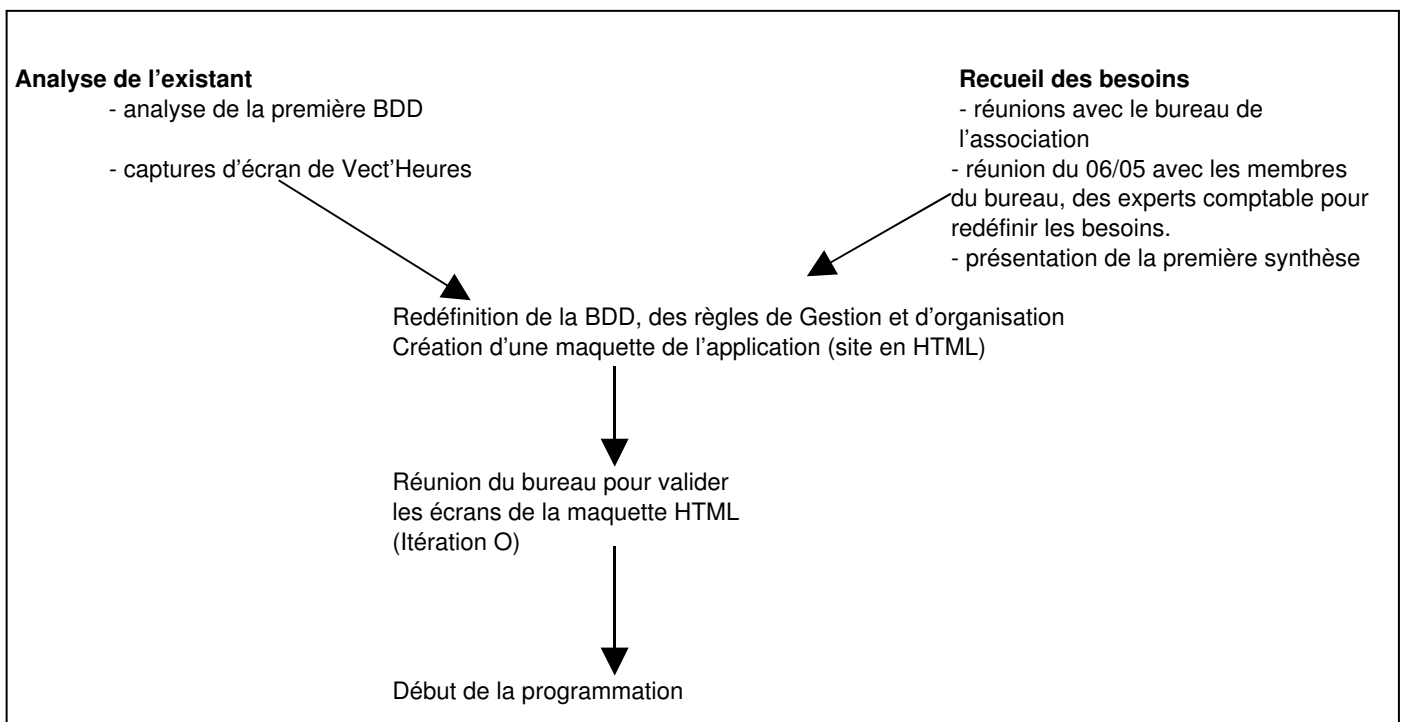
JEI Time est une application web permettant à des collaborateurs de saisir des temps (heures) sur des tâches appartenant à des projets. Ses saisies peuvent être rassemblées dans des tableaux récapitulatifs permettant ensuite de faire les demandes de CIR (crédit impôt recherche).

L'administrateur doit pouvoir contrôler les saisies des collaborateurs, gérer ses projets, ses collaborateurs, générer ses tableaux.

Cette application doit répondre à quatre impératifs :

- être adaptée pour la demande de CIR (crédit impôt recherche)
- permettre une souplesse dans la définition des projets
- être simple d'utilisation
- être simple d'installation

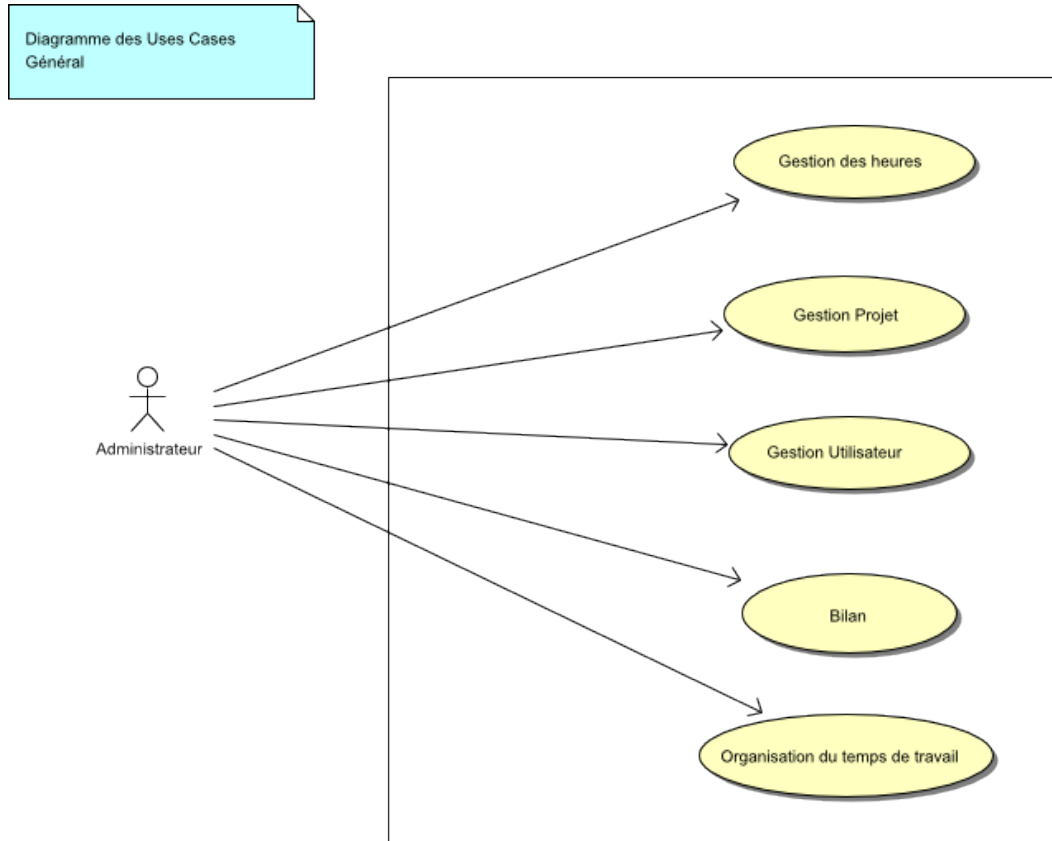
4) Schéma descriptif de la conception de l'application



II) Diagrammes des Uses Cases, règles de gestion et d'organisation

Cinq cas d'utilisation ont été identifiés la gestion des heures, la gestion des projets, la gestion des utilisateurs, les bilans, l'organisation du temps de travail. Chacun d'eux regroupe des sous-scénarios.

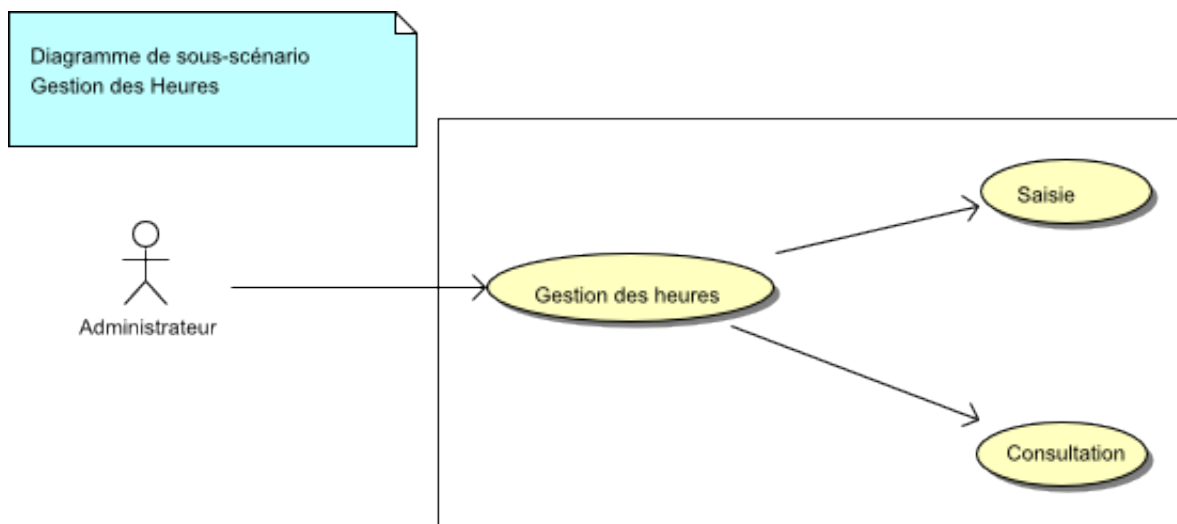
1) Diagramme général des Uses Cases



2) Diagramme des sous-scénarios

Sous scénario : Gestion des heures

La gestion des heures s'organise selon deux sous-ensembles la saisie des heures et la consultation du tableau de bord.

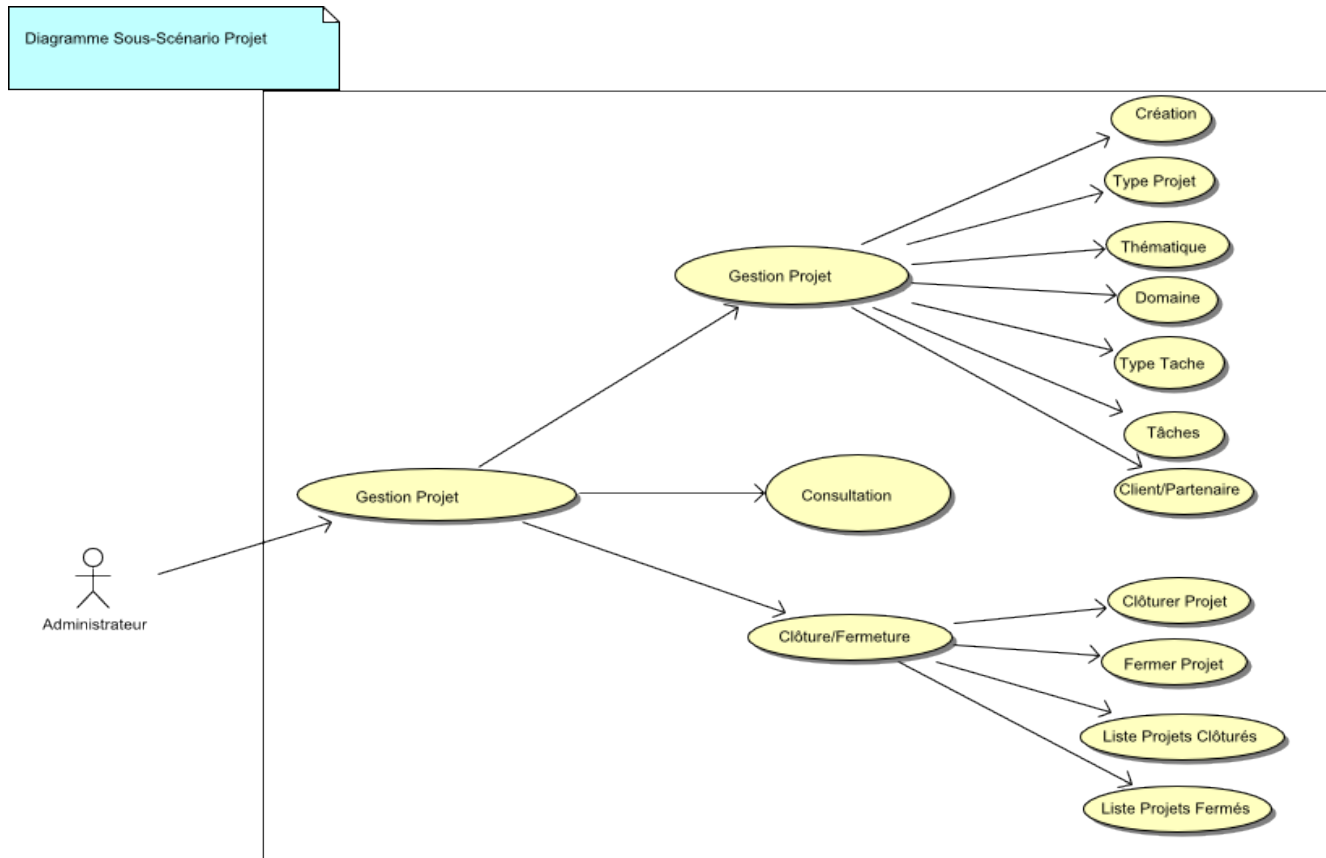


Sous-scénario : Gestion des Projets

La gestion de projet se décline selon trois sous-ensembles la gestion projet, la consultation d'un projet et la clôture/fermeture d'un projet.

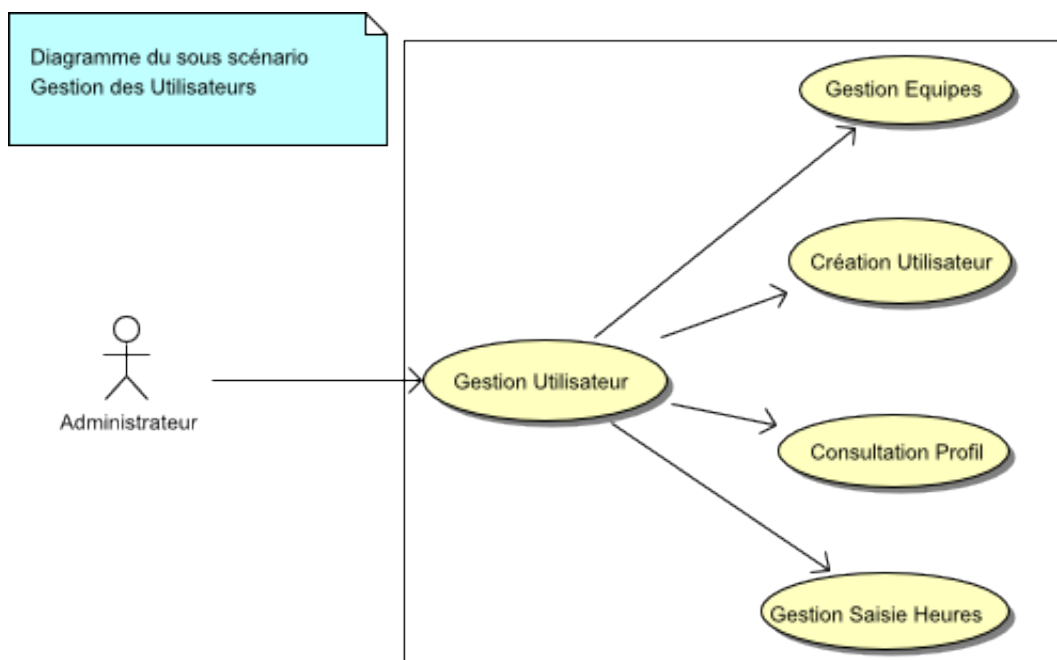
La gestion projet se décompose en sept modules création projet, gestion type de projet, gestion des thématiques, gestion des domaines, gestion des types de tâche, gestion des tâches, gestion des clients/partenaires.

La clôture/fermeture de projet se décompose en quatre modules clôturer un projet, fermer un projet, voir la liste des projets clôturés, voir la liste des projets fermés.



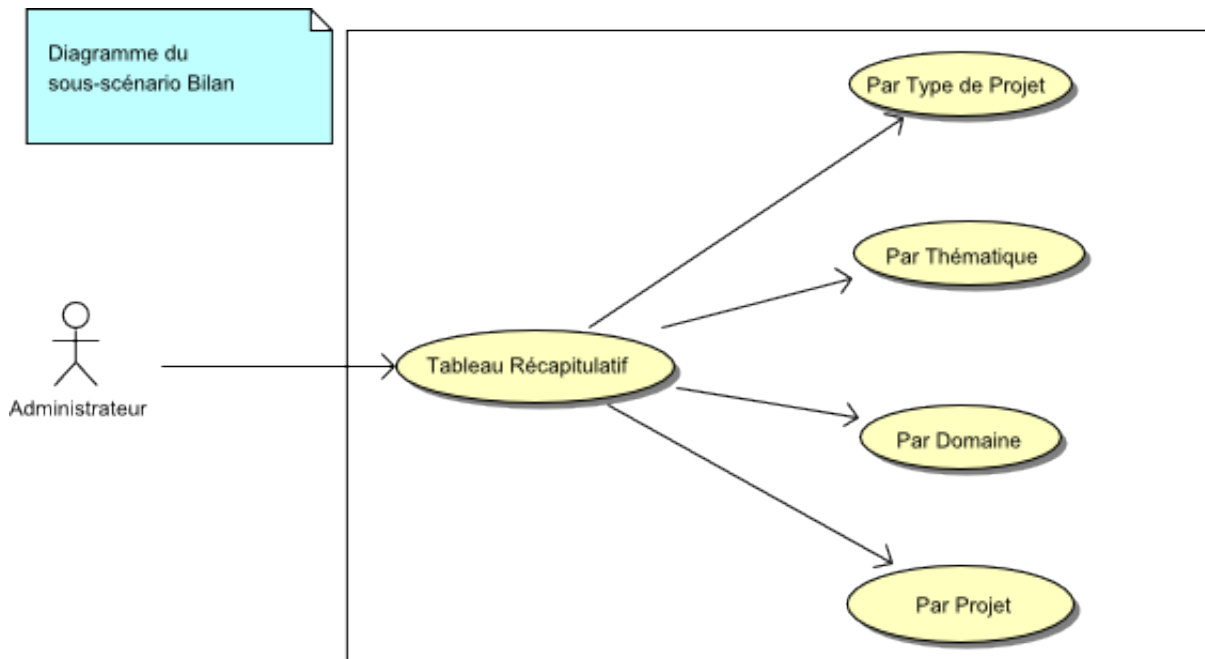
Sous-scénario : Gestion des Utilisateurs

La gestion des utilisateurs se décompose en quatre sous-ensembles la gestion des équipes, la création d'un utilisateur, la consultation d'un profil, la gestion des Saisies des heures.



Sous-scénario : Bilan

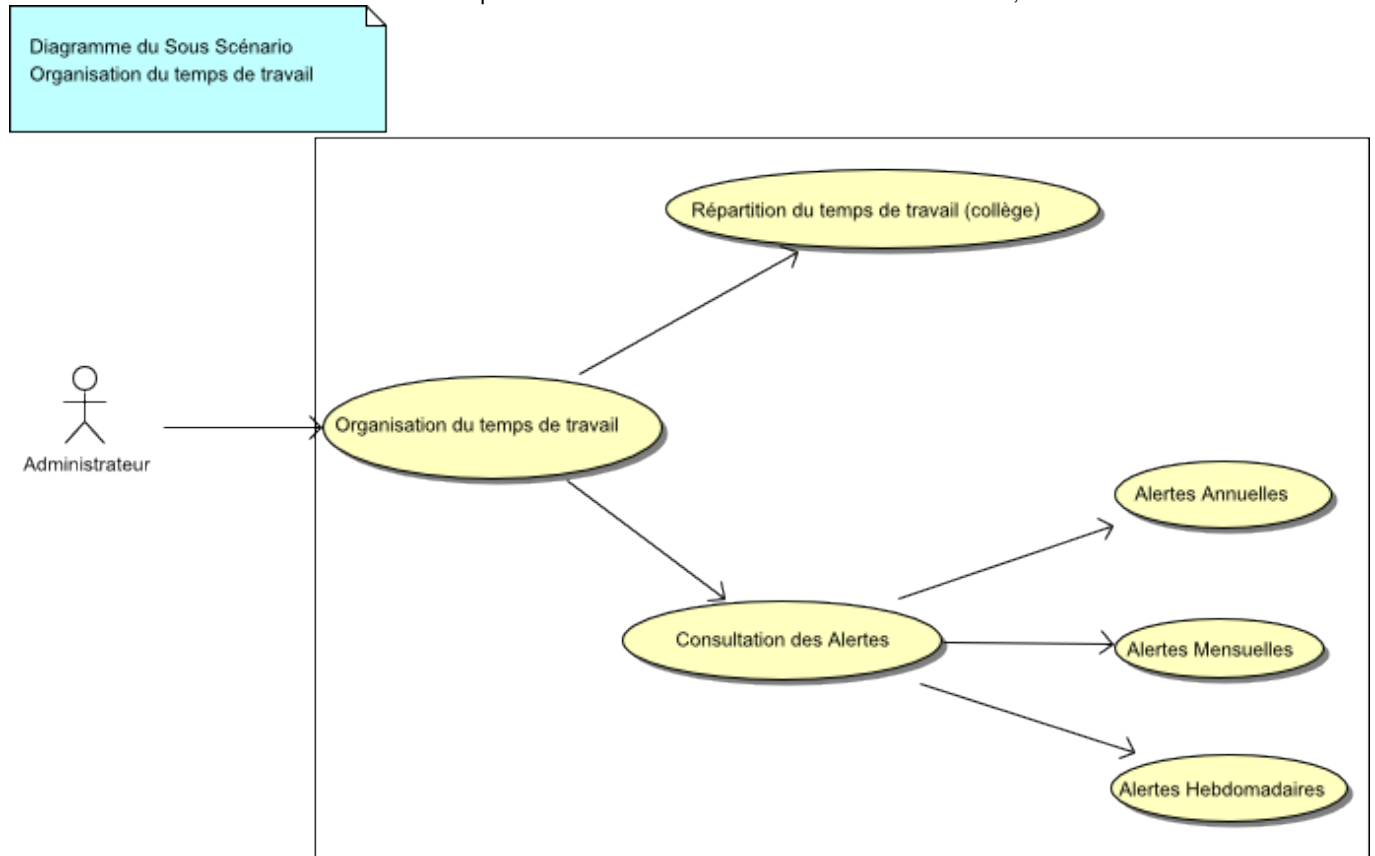
La gestion des bilans comprend quatre sous ensembles tableau récapitulatif par type de projet, tableau récapitulatif par thématique, tableau récapitulatif par domaine et tableau récapitulatif par projet



Sous-scénario : Organisation du temps de travail

L'organisation du temps de travail s'organise selon deux sous-ensembles la répartition du temps de travail (collège c'est dans ce module que l'on définit les différents éléments du nombre d'heures travaillées par jour, par mois et par année, par exemple le collège des cadres ou des techniciens) et la consultation des alertes.

La consultation des alertes est composée de trois sous-modules les alertes annuelles, mensuelles et hebdomadaire.



3) Présentation du tableau des droits des différents profils

Quatre profils ont été identifiés, chacun ayant des droits différents sur l'application :

L'administrateur, le chef de projet, le gestionnaire financier et le collaborateur.

Le tableau suivant présente la matrice fonction/droit.

			Administrateur	Chef de Projet	Gestionnaire Financier	Collaborateur
Gestion Projet						
Gestion Projet	Creation		X	X		
	Type Projet		X			
	Thématique		X			
	Domaine		X			
	Type Tâche		X			
	Tâche		X			
	Client/Partenaire		X			
	Consultation Projet		X	X	X	
Clôturer ou Fermer un projet						
	Clôturer		X			
	Fermer		X	X		
	Liste projet Clôturés		X			
	Liste projet Fermés		X	X		

Gestion Utilisateur

Equipes		X			
Création Utilisateur		X			
Consultation		X	X	X	m
Consulter, Modifier, Saisir Heures		X	X		

Bilan

Tableau Récapitulatifs					
	Par Type Projet		X		X
	Par Thématique		X		X
	Par Domaine		X		X
	Par Projet		X		X

Gestion des Heures

Saisies des heures		X	X	X	X
Consultation		X	X	X	X

Organisation du temps de travail

Répartition du temps de travail (collège)			X		
Consulter les alertes					
	Alertes Annuelles		X	X	
	Alertes Mensuelles		X	X	
	Alertes Hebdomadaires		X	X	

4) Règles de gestion et d'organisation

Liste des différentes règles de gestion et d'organisation selon les cas d'utilisation : gestion projet, gestion utilisateur, bilans, gestion des heures et organisation du temps de travail.

Gestion de Projet :

Un projet

- est défini par un nom
- doit être rattaché à un type de projet
- peut être rattaché à un domaine, une thématique ou à un autre projet
- doit avoir une date de début
- peut avoir une date prévue de fin
- peut avoir aucun ou plusieurs Client/Partenaire
- peut avoir un budget prévisionnel
- peut avoir un temps prévisionnel
- doit avoir de une à plusieurs tâches
- Une tâche doit avoir une priorité (faible, moyenne, forte)
- Une tâche doit être R&D ou non R&D
- Un budget prévisionnel peut être défini pour une tâche
- Un temps prévisionnel peut être défini pour une tâche
- peut être fermé (il disparaît des listes de sélection des saisies des heures pour les collaborateurs)
- fermé peut être rouvert
- peut être clôturé (il devient seulement consultable et disparaît des listes de sélection des saisies des heures pour les collaborateurs et l'interface de gestion administrateur)
- peut-être modifié tant qu'il n'est pas clôturé.
- peut-être supprimé tant qu'il n'est ni clôturé, ni qu'aucune heure n'a été saisie sur ses tâches
- Quatre types de projet (Hors-Projet, Recherche Appliquée, Recherche Expérimentale, Recherche fondamentale) sont déjà créés au départ de l'application car ils sont communs à toutes les entreprises qui utiliseront l'application.
- la création de thématique, ou de domaine ou de client/partenaire n'est pas obligatoire pour la création d'un projet.
- Un type de tâche autre est définie au départ de l'application.
- Sept tâches (Autre, Congés, Formation (non scientifique) Fériés, Maladie / Congés Maternité, RTT, Réunion (non scientifique)) sont définies au départ de l'application.

Gestion des Domaines, Thématiques et Type de tâche :

- sont définis par un nom
- peuvent être modifiés
- peuvent être supprimés tant qu'ils ne sont pas rattachés à un projet (domaine et thématique) ou à une tâche (type de tâche)

Gestion des Tâches :

- doivent être rattachées à un type de tâche
- sont définies par un nom
- peuvent être modifiées
- peuvent être supprimées si elles ne sont pas rattachées à des projets

Gestion des Utilisateurs

Gestion des équipes :

Une équipe

- est définie par un nom
- peut avoir une fonction
- peut être modifiée
- peut être supprimée si aucun utilisateur n'y est rattaché.
- la création d'au moins une équipe n'est pas obligatoire pour la création d'un utilisateur.

Gestion utilisateur :

Un Utilisateur :

- est définie par un nom, prénom et login
- a un mot de passe (qui est généré automatiquement par l'application, c'est la première lettre du prénom + le nom tout en minuscule (exemple : Jean Dupond mot de passe : jdupond))
- peut être rattaché à une équipe
- a obligatoirement un statut (administrateur, chef de projet, gestionnaire financier ou collaborateur)
- est obligatoirement rattaché à un collège
- doit être à temps plein (les valeurs de ses heures journalières sont celles du collège auquel il appartient) ou à temps partiel (il faut définir les heures journalières)
- a un salaire brut annuel
- montant des charges obligatoires annuelles (hors mutuelle, ticket resto, etc ...)
- peut être modifié

- peut être supprimé, uniquement s'il n'a jamais fait de saisie d'heures.
- lorsqu'un collègue est modifié, il faut modifier les collaborateurs qui appartiennent à ce collègue
- peut appartenir à plusieurs collègue (historique) seule le lien le plus récent apparaît.
- est définie au départ de l'application (login admin mot de passe : admin avec les droits d'administration)

Les bilans

Tous les tableaux récapitulatifs sont exportables au format csv.

Par type de projet :

- il faut définir une période (date de début, date de fin)
- il faut sélectionner les types de projets
- un premier tableau récapitulatif regroupant tous les projets du type concerné peut être visualisé
- une sélection par projet est possible
- un deuxième tableau récapitulatif regroupant les projets sélectionnés et les types concernés peut être visualisé
- une sélection par tâche est possible
- un troisième tableau récapitulatif regroupant les projets sélectionnés, les types concernés et les tâches choisies peut être visualisé
- une sélection par type de tâche est possible
- un quatrième tableau récapitulatif regroupant les projets sélectionnés, les types concernés et les types de tâche choisis peut être visualisé
- une sélection par tâche est possible
- un cinquième tableau récapitulatif regroupant les projets sélectionnés, les types concernés, les types de tâches choisis et les tâches sélectionnées est visualisable
- l'affichage des tableaux se fait par regroupement de type de projet, nom du projet, les collaborateurs et le nombre d'heures saisies

Par domaine ou thématique

- présélection par tâche R&D, non R&D ou toutes
- sélection des domaines ou thématiques
- affichage du tableau récapitulatif regroupant les domaines ou thématiques, les noms de projets, les collaborateurs et le nombre d'heures saisies.

Par projet

- présélection par tâche R&D, non R&D ou toutes
- sélection des projets
- affichage du tableau récapitulatif regroupant les projets, les noms des tâches, les collaborateurs et le nombre d'heures saisies.

Gestion des Heures :

Saisie des heures :

- Interface administrateur : choix du collaborateur au préalable
- choix d'une date
- possibilité de valider pour voir le tableau de bord de la semaine du jour concerné
- choix d'un projet
- choix d'un type de tâche
- choix d'une tâche
- choix du nombre d'heures travaillées sur la tâche
- possibilité d'écrire un commentaire
- affichage d'un tableau récapitulatif toutes les heures saisies pour des projets du jour concerné.
- validation de ce tableau récapitulatif pour l'enregistrement
- possibilité de modifier des heures saisies en cliquant sur l'heure concerné dans le tableau de bord hebdomadaire
- pour supprimer une saisie réinitialiser une heure à zéro

Consultation tableau de bord

- choix d'une date
- affichage du tableau récapitulatif hebdomadaire

Organisation du temps de travail

Répartition du temps de travail : Un Collège

- est défini par un nom
- un nombre d'heure pour le lundi, mardi, mercredi, jeudi, vendredi
- un nombre d'heure mensuel
- un nombre d'heure annuel
- un nombre de congés et RTT annuel
- possibilité de faire une sélection du type d'alerte concernant ce collège (annuelle, mensuelle, hebdomadaire)
- possibilité d'écrire la liste de sélection des saisies des heures (chaque heure doit être séparée par le signe /)

Alertes :

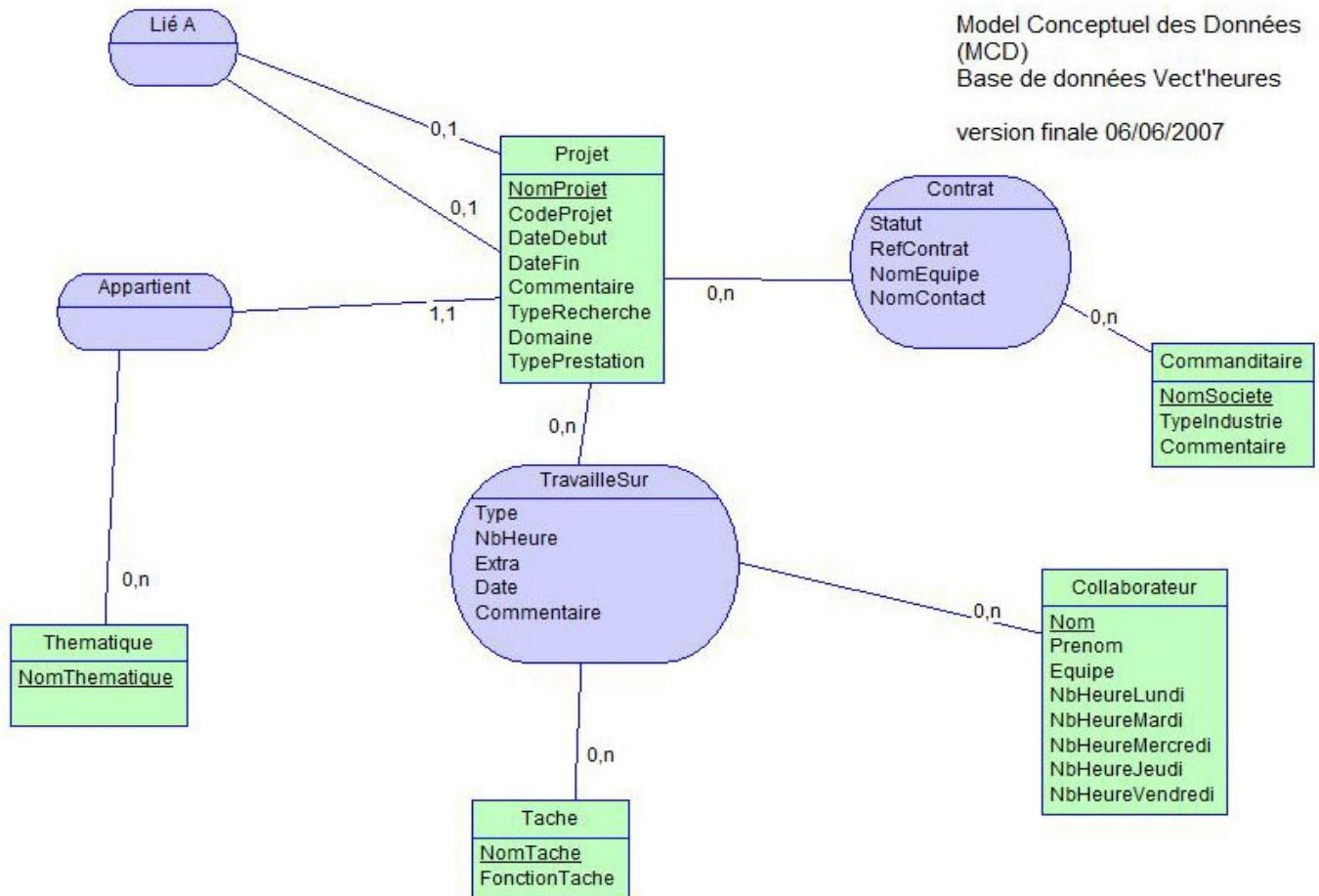
- elles peuvent être annuelles, mensuelles ou hebdomadaires

- choix de la période année (si pas indiquée, trie pour l'année en cours), mois ou jour.
- affichage d'un tableau regroupant le nombre d'heures la période sélectionnée, le nom des collaborateurs concernés, le nombre d'heures défini dans le collège, le nombre d'heures saisi et la différence.

III) La Base de données

1) MCD de Vect'Heures

L'application Vect'heures a été conçue pour répondre aux besoins des JEI pour la saisie des temps sur des tâches d'un projet, pour permettre la génération de tableaux récapitulatifs et faire les demandes de CIR (crédit impôt recherche). Au fil de l'utilisation, différents inconvénients sont apparus, comme le manque de portabilité de l'application (grande difficulté d'installation), un manque de certaines fonctionnalités comme la gestion des utilisateurs et trop orientée génie biologique.



2) Passage de l'une à l'autre

La base de données de Vect'Heures représente la base de l'application.

La ternaire entre le projet, la tâche et le collaborateur n'a pas été gardée. Un collaborateur travaille sur des tâches qui appartiennent à des projets.

Le domaine et le type de recherche sont sortis des projets pour devenir des entités séparées.

De nouvelles entités ont été ajoutées, pour répondre aux besoins :

Role : permet de définir le rôle de l'utilisateur

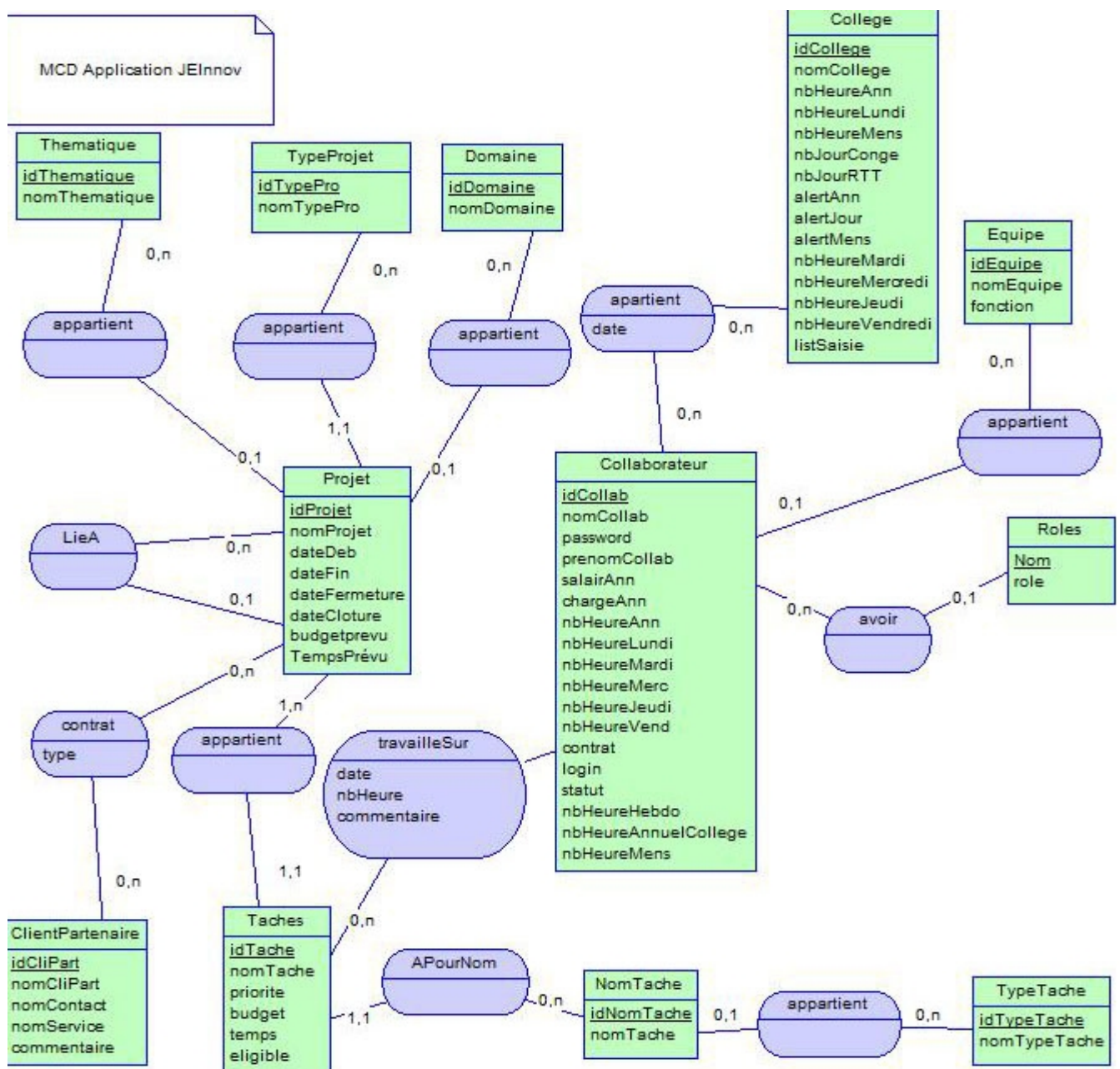
Nom Tâche : dissociation du nom de la tâche car une tâche est rattachée en relation 1,1 avec un projet, mais un même nom de tâche peut apparaître dans différents projets.

Type Tâche : permet d'organiser les tâches et faciliter la recherche lors de la création d'un projet ou de la saisie des heures.

Collège : définit la répartition du temps de travail. Vect'Heures avait des contrôles stricts c'est-à-dire que les collaborateurs ne pouvaient saisir plus de 8 heures par jour. Ces contrôles n'ont pas été gardés, car ils ne correspondaient pas à la réalité de certaines entreprises (par exemple des entreprises permettent les heures supplémentaires, dans Vect'Heures cette notion n'était pas possible).

Equipe : possibilité d'organiser les collaborateurs en équipe.

3) MCD de JEI Time



IV) Description phase de développement

1) Technologies utilisées

Les technologies :

- OpenSUIT (framework dédié à la création des IHM)
- Langage de programmation Java
- Hibernate (framework ORM permettant le dialogue entre l'appli java et le système de base de données)
- HSQLDB (base de données embarquées faite en java)
- Serveur Apache Tomcat
- API Jaas et Realm (Tomcat) pour l'authentification

Présentation du framework OpenSUIT :

- il a été développé par l'entreprise EBM Websourcing
- il permet la génération rapide d'IHM
- il est basé sur des Bean Java
- il permet d'avoir un cadre fixe pour les composants graphiques
- gain de temps et de qualité dans le développement d'une application

Un fichier base.xsd permet de générer les fichiers xml et de spécifier leur nature (fichier de mapping, frame ou page.composite).

Le framework s'organise autour de fichier xml :

Un premier fichier nomDeAppli.xml qui gère le mapping de toutes les pages.

Un second fichier frame.xml qui est un conteneur. C'est dans ce fichier xml que l'on définit l'emplacement de chaque élément qui apparaîtra à l'écran (Entête, menu de gauche, le pied de page).

Un ensemble de fichier xml représentant les pages qui s'afficheront à l'écran (exemple les formulaires).

a) exemple de fichier de mapping : jeitime.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<base:Application HomePage="Accueil" xmlns:base="org.ow2.opensuit.xml.base" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="org.ow2.opensuit.xml.base base.xsd">
  <SupportedLocales>
    <base:locale.LocaleConfig Language="en"/>
    <base:locale.LocaleConfig DateOutputFormat="dd/MM/yyyy" Language="fr"/>
  </SupportedLocales>
  <LocaleSelector>
    <base:locale.Fixed Language="fr"/>
  </LocaleSelector>
  <MessagesProviders>
    <base:msg.ResourceBundle BaseName="appliJEI2.AppliJEInnovMessageRessources" Name="default"/>
  </MessagesProviders>

  <GlobalBeans>
    <base:binding.JavaBeans>
      <base:binding.Bean Class="appliJEI2.collaborateur.uiBeans.UserConfig" LifeScope="Session" Name="userConfig"/>
    </base:binding.JavaBeans>
  </GlobalBeans>

  <Frames>
    <base:Import_html.IFrame File="frames/frame.xml"/>
  </Frames>
  <Pages>
    <base:Import_page.IPage File="Pages/Accueil.xml"/>

    <!-- Pages Pour la Gestion des Projets (Projet, Type Projet, Domaine, Thématique, Client/Partenaire, Type Tâche) -->
    <base:Import_page.IPage File="Pages/Projet/CreationProjet.xml"/>
    <base:Import_page.IPage File="Pages/Projet/ChoiProjet.xml"/>
    <!-- Pages Pour la Gestion des Utilisateurs -->
    <base:Import_page.IPage File="Pages/Collaborateurs/ConsultUtilisateur.xml"/>
  </Pages>
</base:Application>
```

b) exemple de fichier frame.xml

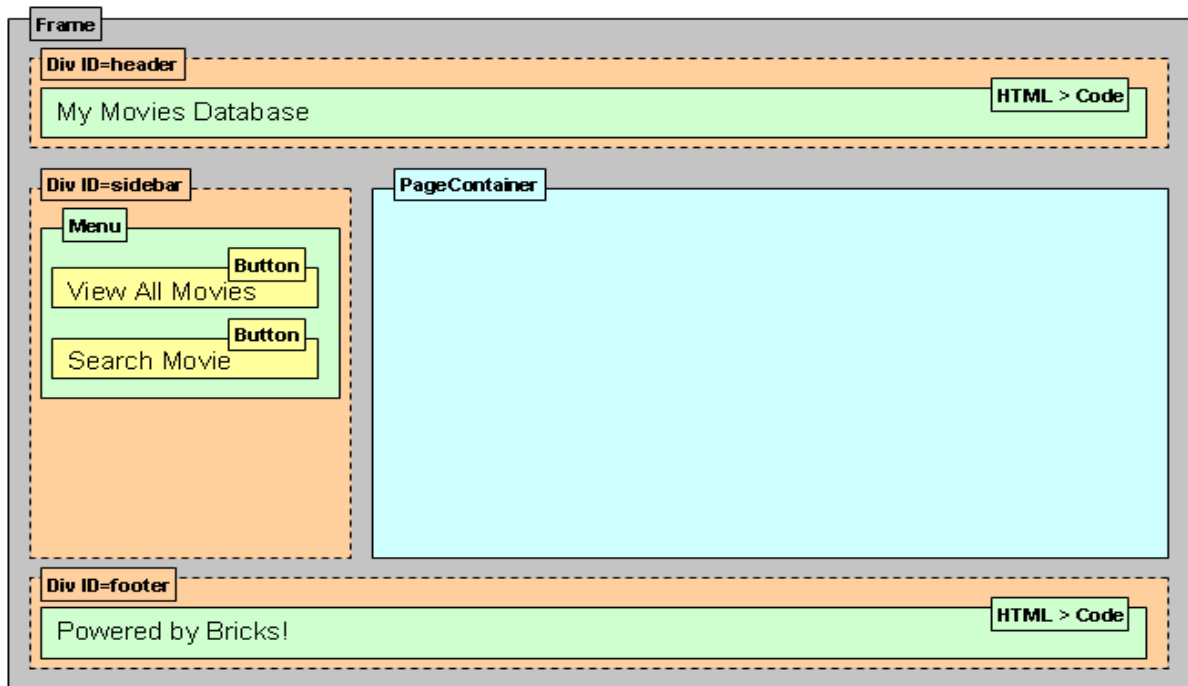
```
<?xml version="1.0" encoding="UTF-8"?>
<base:html.Frame Name="fenetre" xmlns:base="org.ow2.opensuit.xml.base"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="org.ow2.opensuit.xml.base ../base.xsd">

  <StyleSheets> <!--Définition du fichier css qui va servir à la mise en page de la page HTML →
    <base:html.CssStyleSheet File="resources/css/style2.css"/>
  </StyleSheets>
  <base:html.layout.Div ID="header"> <!--Définition de l'entête de l'application pour nous le logo →
    <base:html.HTML>
      <Code> </Code>
    </base:html.HTML>
  </base:html.layout.Div>
  <base:html.menu.Menu CssClass="HeaderMenu"> <!--Définition des boutons qui vont apparaître en haut de la page →
    <Items>
      <base:html.menu.Button>
        <Title>${userConfig.identity}</Title>
        <Icon>resources/icons/user.gif</Icon>
        <OnClick>
          <base:action.SwitchAction>
            <Switch>${empty userConfig.collab}</Switch>
            <Cases>
              <base:action.Case Name="false">
                <base:action.ShowPage Page="ConsultUtilisateur">
                  <Parameters>
                    <base:param.DynParam Name="id">
                      <Value>${userConfig.collab.idColl}</Value>
                    </base:param.DynParam>
                  </Parameters>
                </base:action.ShowPage>
              </base:action.Case>
            </Cases>
          </base:action.SwitchAction>
        </OnClick>
      </base:html.menu.Button>
    </Items>
  </base:html.menu.Menu>
  <base:html.layout.Div ID="sidebar"> <!--Définition du menu de gauche →
    <base:html.menu.Menu>
      <Title>Gestion de projet</Title>
      <Visible>${request.isUserInRole("administrateur")}|| request.isUserInRole("chef_de_projet")}</Visible>
      <Items>
        <base:html.menu.SubMenu>
          <Title>Gestion Projet</Title>
          <Items>
            <base:html.menu.Button>
              <Title>Création Projet</Title>
              <OnClick>
                <base:action.ShowPage Page="CreationProjet">
                  </base:action.ShowPage>
              </OnClick>
              <Visible>${request.isUserInRole("administrateur")} ||
                request.isUserInRole("chef_de_projet")}</Visible>
            </base:html.menu.Button>
          </Items>
        </base:html.menu.SubMenu>
        <base:html.menu.Button>
          <Title>Consultation Projet</Title>
          <OnClick>
            <base:action.ShowPage Page="ChoiProjet"/>
          </OnClick>
          <Visible>${request.isUserInRole("administrateur")} || request.isUserInRole("chef_de_projet")} ||
            request.isUserInRole("gestionnaire")}</Visible>
        </base:html.menu.Button>
      </Items>
    </base:html.menu.Menu>
  </base:html.layout.Div>
  <base:html.PageContainer />
  <base:html.layout.Div ID="footer"><!--Définition du pied de page →
    <base:html.HTML>
      <Code> <![CDATA[Créé et développé par JEInnov]]> </Code>
    </base:html.HTML>
  </base:html.layout.Div>
</base:html.Frame>
```

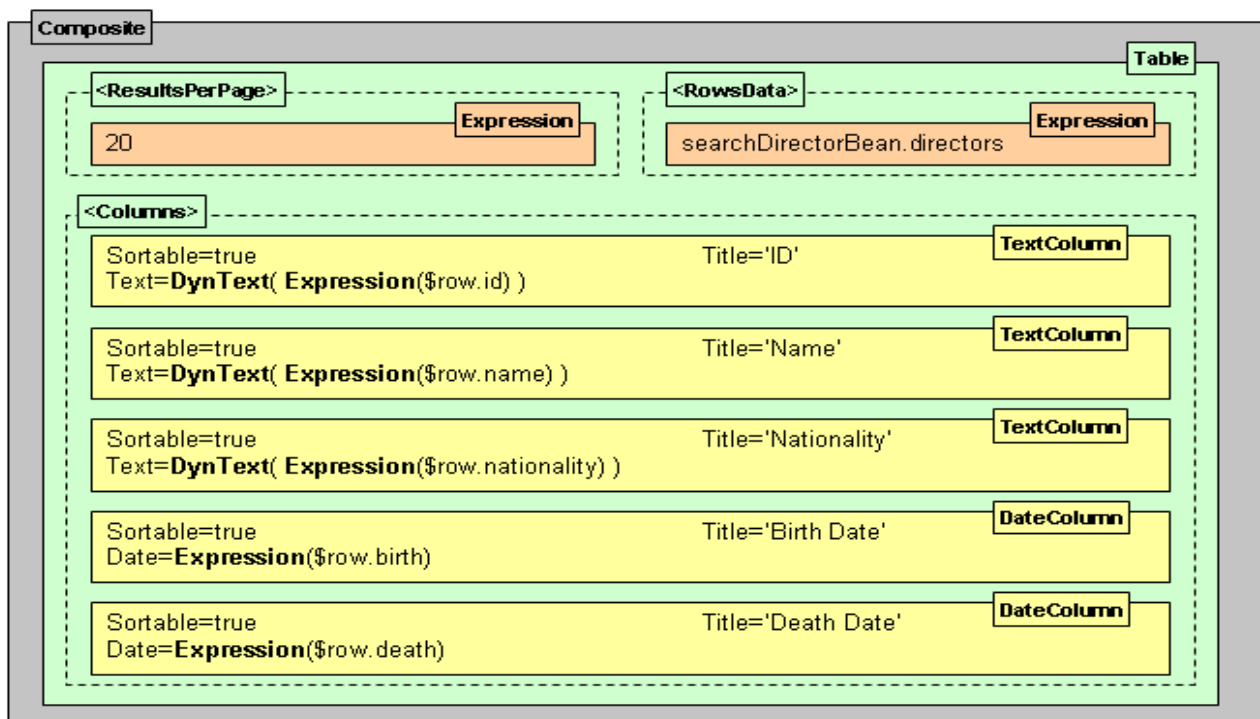
c) exemple de fichier choixProjet.xml contenant un tableau

```
<?xml version="1.0" encoding="UTF-8"?>
<base:page.Composite Name="ChoiProjet" xmlns:base="org.ow2.opensuit.xml.base"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="org.ow2.opensuit.xml.base ../../base.xsd">
  <Context>
    <base:page.ctx.Definition> <!-- Définition de différents éléments nécessaire à la page Autorisation d'accès à cette page, l'UIBean
qui va permettre l'affichage des informations et le chargement des informations nécessaires à l'affichage -->
      <Authorization>${request.isUserInRole("administrateur")} || request.isUserInRole("chef_de_projet")} ||
request.isUserInRole("gestionnaire")}</Authorization>
      <LocalBeans>
        <base:binding.JavaBeans> <!-- UIBean de référence -->
          <base:binding.Bean
            Class="appliJEE2.projet.uiBeans.ProjetUIBean"
            LifeScope="PageContext" Name="projetBean" />
          </base:binding.JavaBeans>
        </LocalBeans>
        <OnLoad> <!-- Chargement des methodes informations necessaires à l'affichage du tableau de la page -->
        <base:binding.Do>${projetBean.load(request)}</base:binding.Do>
        </OnLoad>
      </base:page.ctx.Definition>
    </Context>
    <Title>List des Projets </Title>
    <base:html.table.Table ID="tableCollab" SortAndPaginateLocally="true">
      <ResultsPerPage>${20}</ResultsPerPage>
      <RowsData> ${projetBean.allprojet} </RowsData> <!-- ArrayList qui va remplir le tableau>
      <Columns>
        <base:html.table.TextColumn>
          <Title> Nom du projet </Title>
          <Text>${$row.nomProjet} </Text>
          <OnClick>
            <base:action.ShowPage Page="ModifProjet">
              <Parameters>
                <base:param.DynParam Name="id">
                  <Value>${$row.idProjet}</Value>
                </base:param.DynParam>
              </Parameters>
            </base:action.ShowPage>
          </OnClick>
        </base:html.table.TextColumn>
        <base:html.table.DateColumn>
          <Title>Date de début</Title>
          <Date>${$row.dateDeb} </Date>
        </base:html.table.DateColumn>
        <base:html.table.DateColumn>
          <Title>Date prévu de fin </Title>
          <Date>${$row.dateFin}</Date>
        </base:html.table.DateColumn>
        <base:html.table.DateColumn>
          <Title> Date Fermeture</Title>
          <Date>${$row.dateFermeture}</Date>
        </base:html.table.DateColumn>
      </Columns>
    </base:html.table.Table>
  </base:page.Composite>
```

d) image représentant le découpage de la fenêtre qui apparaît à l'écran.



e) Schéma représentant un tableau :



2) Tableau récapitulatif des itérations

Pour le déroulement du développement de l'application une méthode par itération a été choisie.
Le tableau suivant présente les itérations et les composants développés pour chacune d'elles.

			Itération 1	Itération 2	Itération 3	Itération 4
Gestion Projet						
Gestion Projet	Creation		X	m		m
	Type Projet		X			
	Thématique		X			
	Domaine		X			
	Type Tâche		X			
	Tâche		X			
	Client/Partenaire		X	m		
Consultation Projet				X		
Clôturer ou Fermer un projet						
	Clôturer			X		
	Fermer			X		
	Liste projet Clôturés			X		
	Liste projet Fermés			X		
Gestion Utilisateur						
Equipes			X			
Création Utilisateur			X	m		
Consultation				X		
Consulter, Modier, Saisir Heures				X	m	m
Bilan						
Tableau Récapitulatifs	Par Type Projet				X	m
	Par Thématique				X	
	Par Domaine				X	
	Par Projet				X	
Gestion des Heues						
Saisies des heures				X	m	m
Consultation				X		
Organisation du temps de travail						
Répartition du temps de travail (collège)			X	m	m	m
Consulter les alertes						
	Alertes Annuelles					X
	Alertes Mensuelles					X
	Alertes Hebdomadaires					X
Apparences						
Definition du nom				X	m	
Définition de la charte					X	

X : première programmation d'un composant
m : modification d'un composant

V) Diagrammes de conceptions détaillées

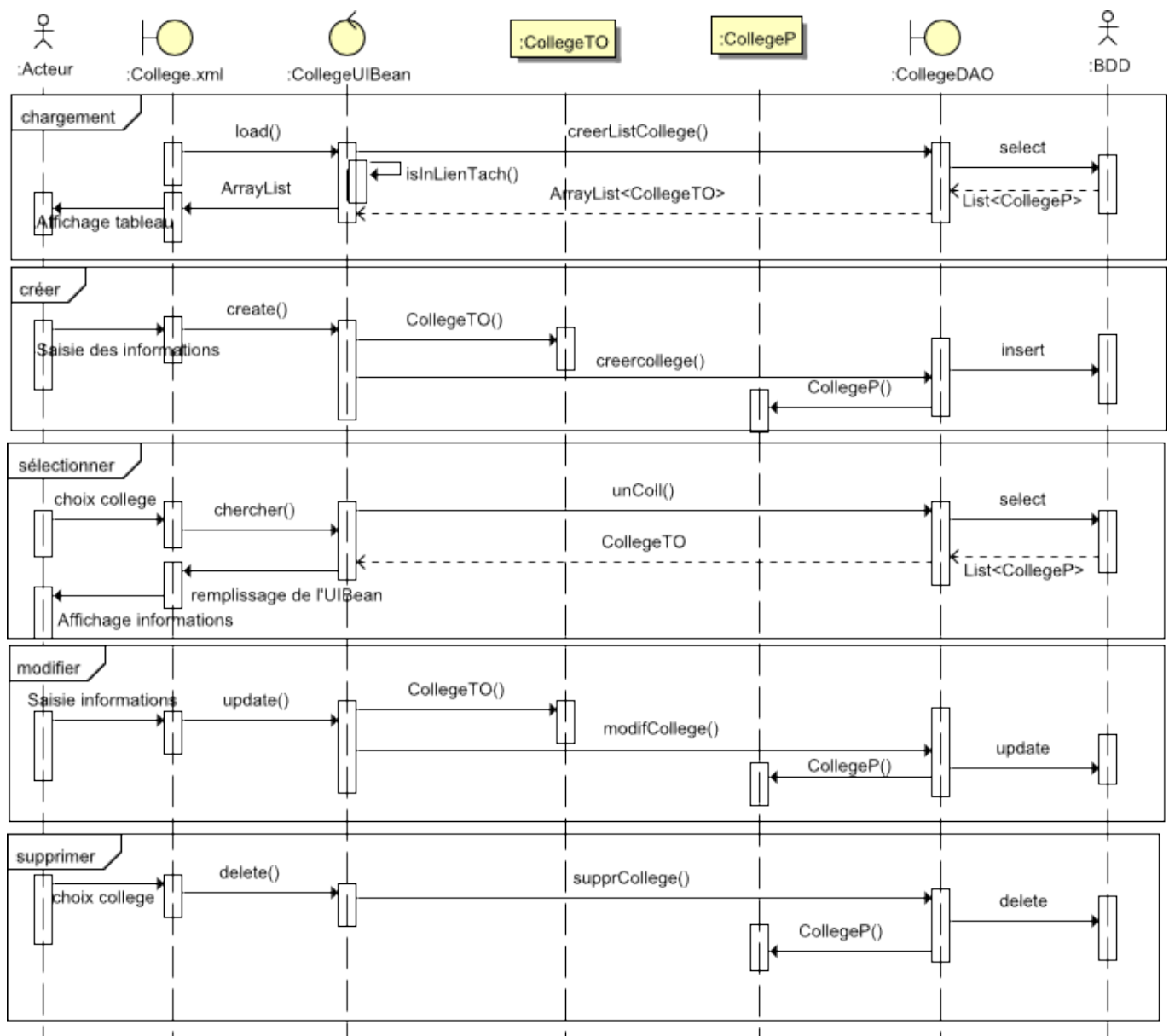
1) Description de l'architecture de l'application

- Modèle présentation : c'est à dire que la vue et le contrôleur sont fusionnés.
- Les objets transitent de l'IHM à la base de données par le biais de quatre couches :
 - Les UIBeans
 - Les TransferObject (TO)
 - Les Persistents
 - Les DAO

Les UIBeans sont en relations avec les fichiers xml. Ce sont eux qui récupèrent les informations saisies par les utilisateurs et qui se chargent de les transformer en TO et les envoient aux DAO. Ces derniers transforment les TO en Persistents pour qu'ils soient enregistrés dans la base de données. Ils sont en charge du traitement. Un UIBean est composé de variables, de méthodes de traitement et des getters/setters des variables.

A l'inverse, pour l'affichage des informations, les DAO se chargent de récupérer les informations de la base de données par le biais des Persistents, puis les transforment en TO et les renvoient à l'UIBean qui se chargera de les envoyer au fichier xml pour l'affichage.

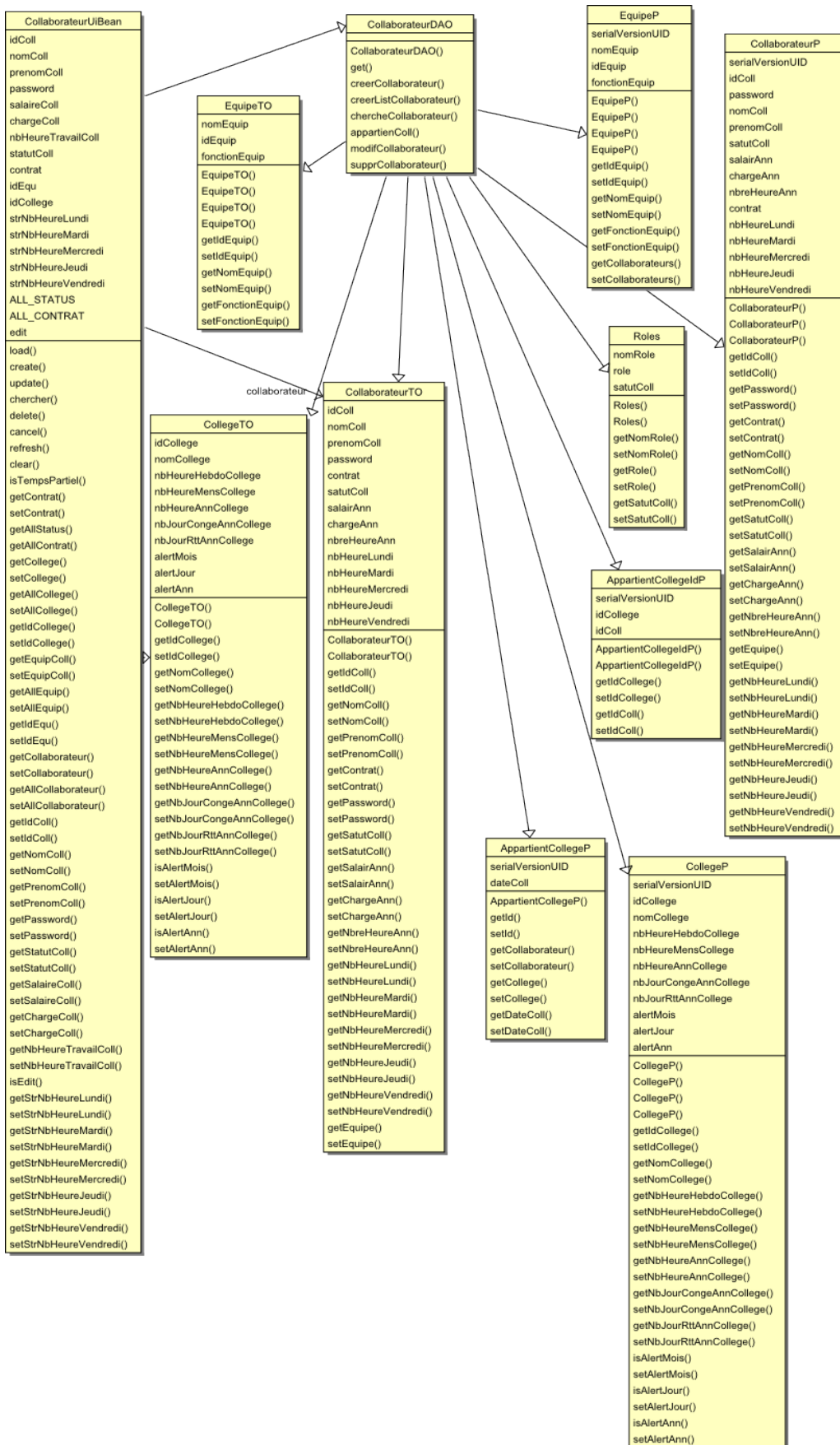
Exemple d'un diagramme de séquence détaillée pour la gestion des collèges

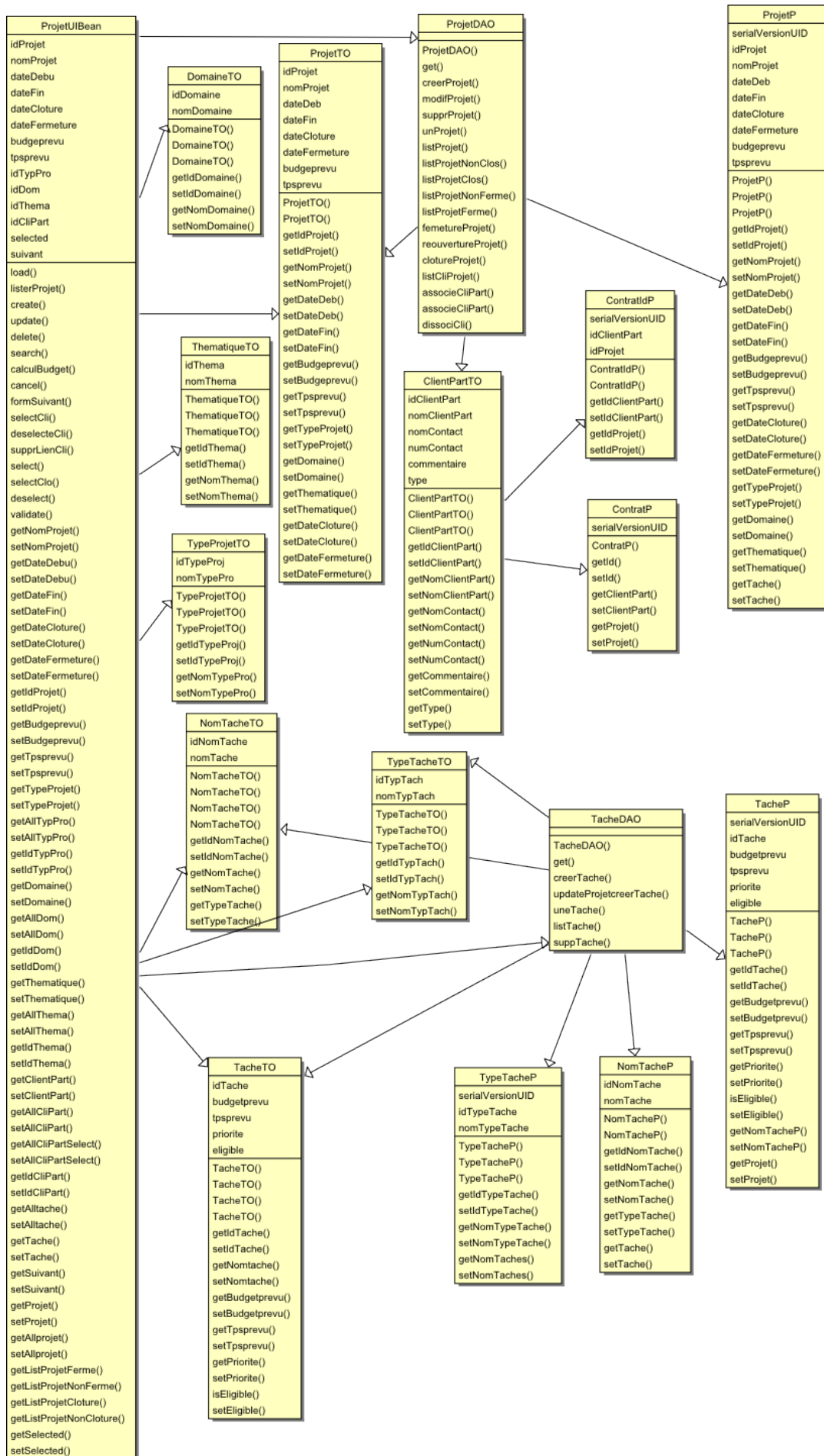


2) Diagrammes de classes

Exemple de deux diagrammes de classes :

Gestion Utilisateur

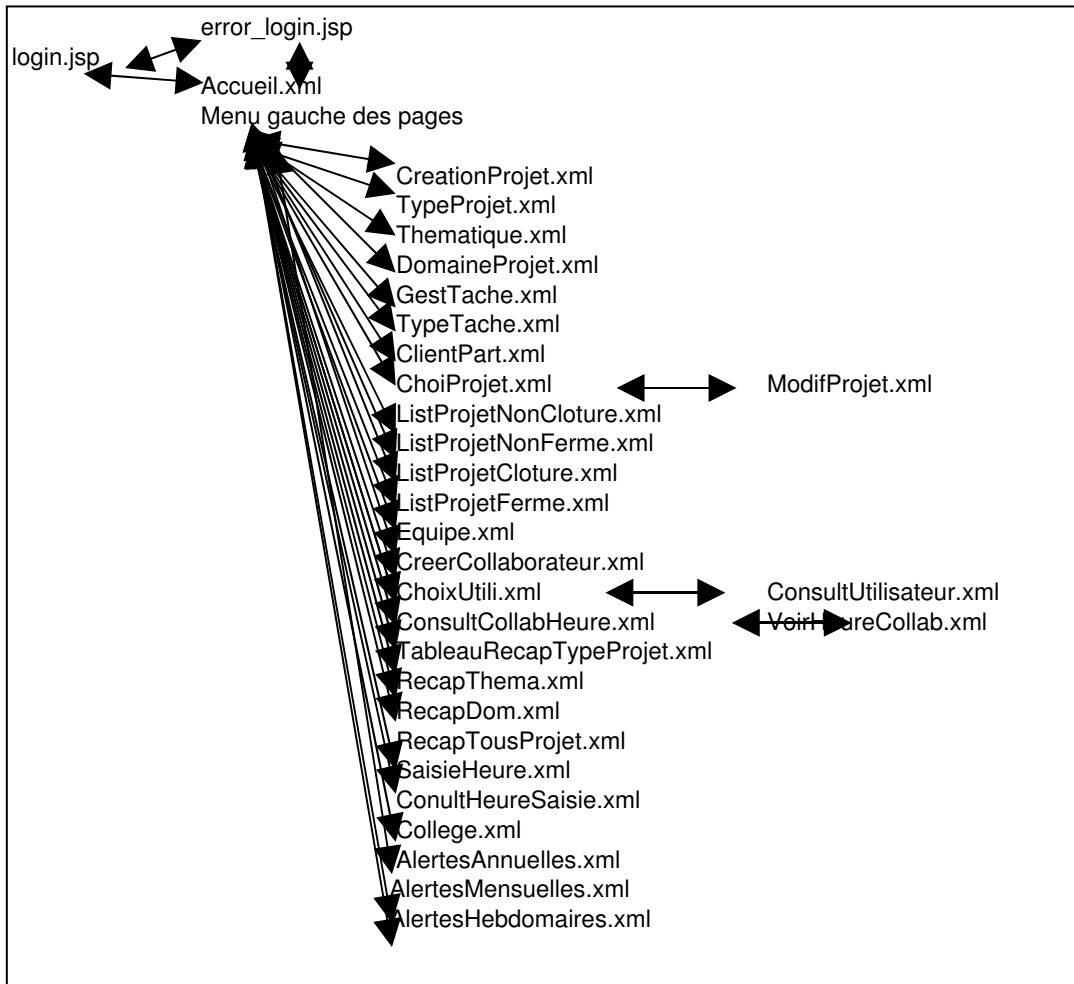




3) Enchaînement d'écrans

L'application est basée sur son menu à gauche, chaque page est accessible via ce menu.

Schéma montrant toutes les pages et leurs liens dans le menu



Captures d'écran de la page d'accueil pour les différents profils

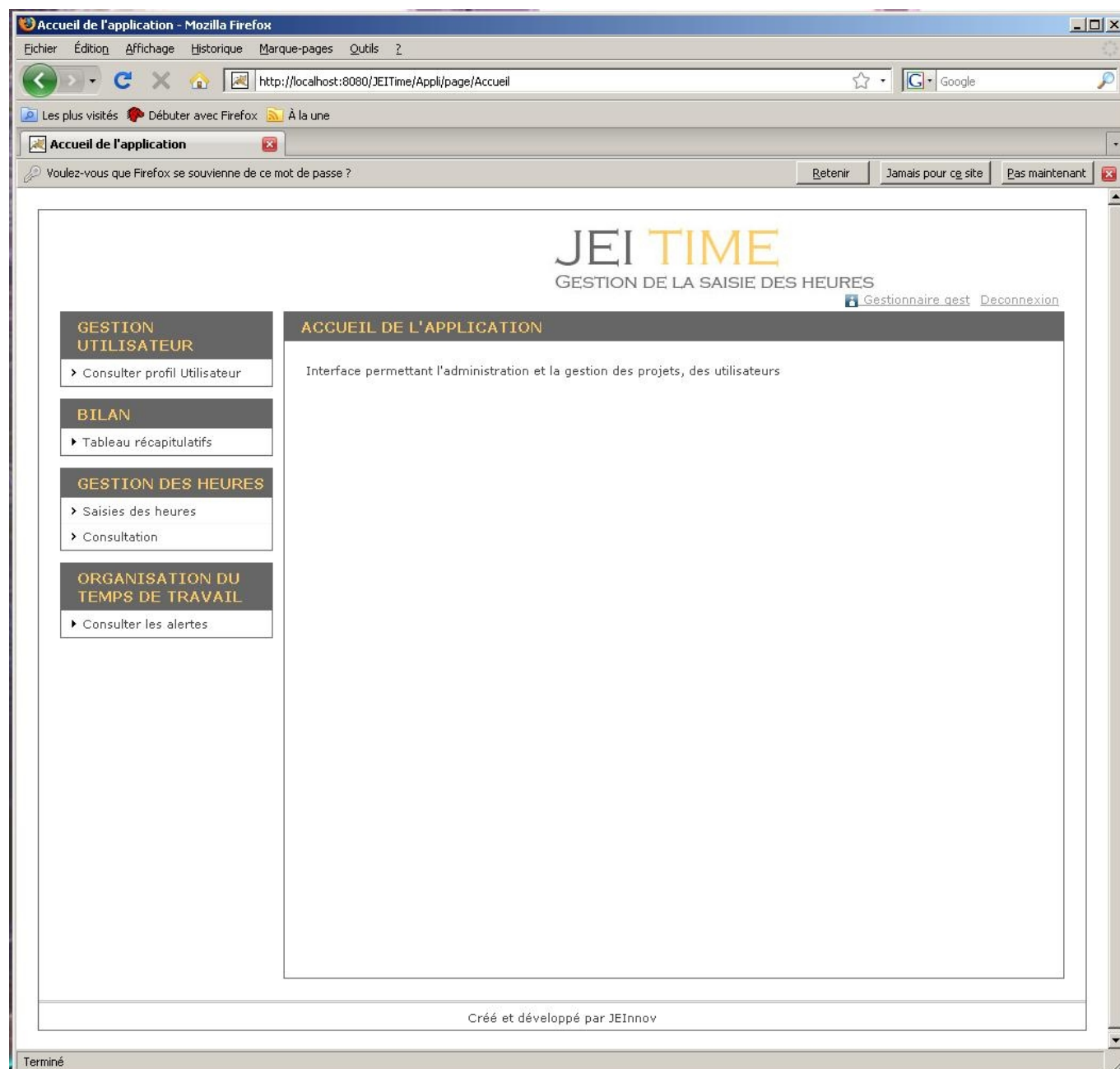
Interface administrateur :



Interface chef de projet :



Interface Gestionnaire Financier :



Interface utilisateur :



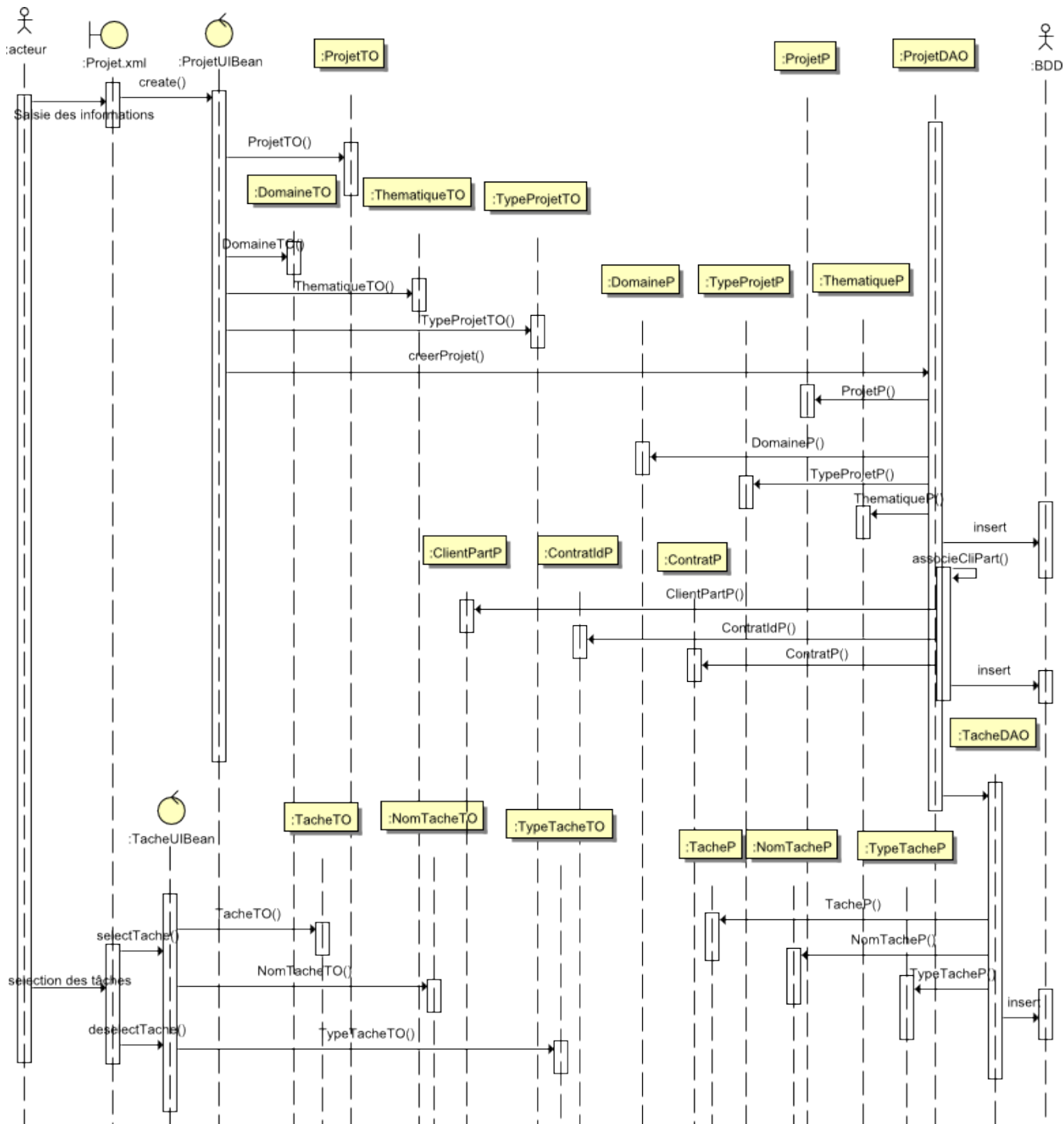
Annexes 1

Ensemble des Diagrammes de séquence détaillée

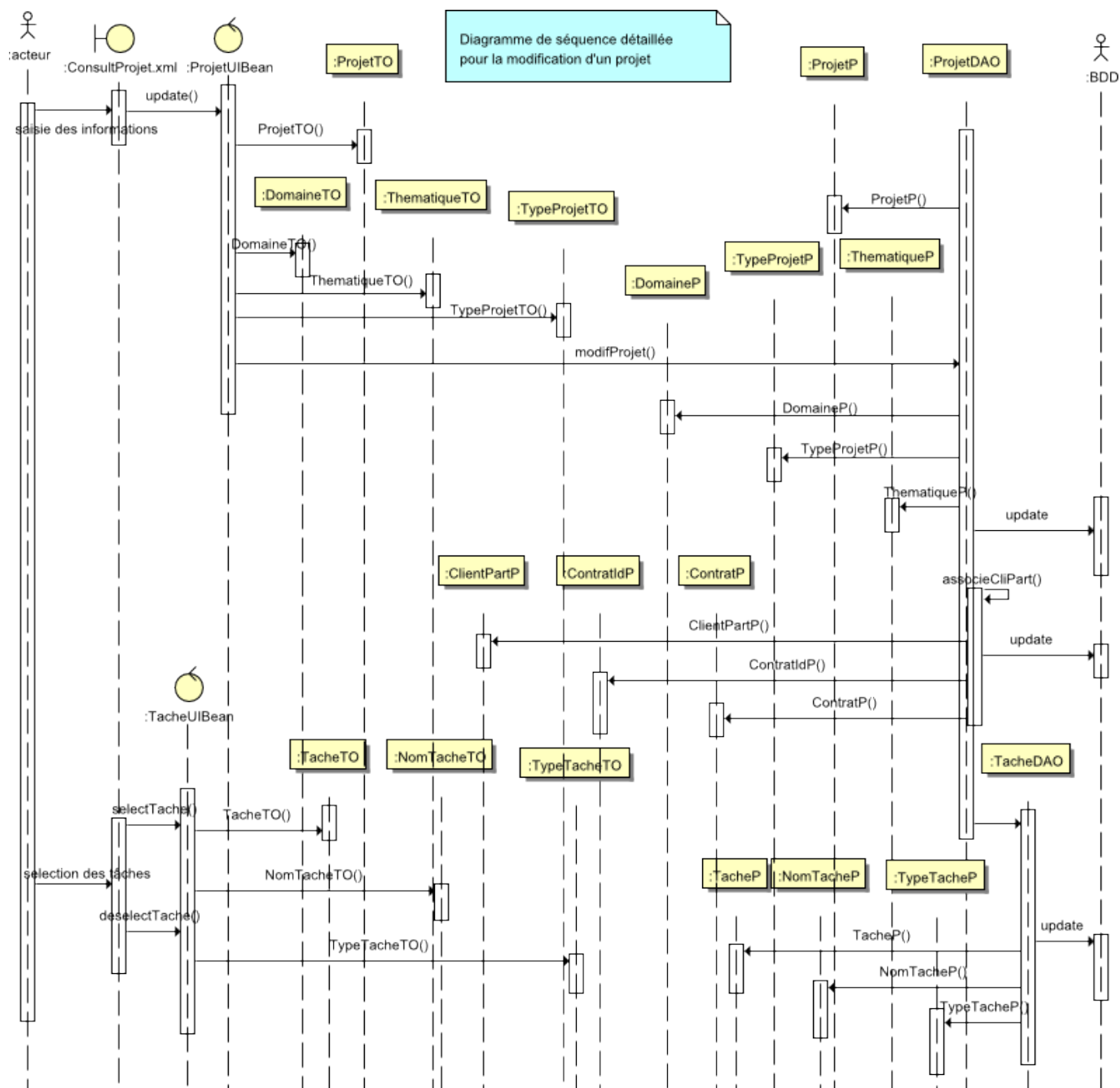
Les diagrammes de séquence détaillée sont classés selon les cas d'utilisation : gestion projet, gestion utilisateur, bilans, gestion des heures et organisation du temps de travail.

Gestion Projet

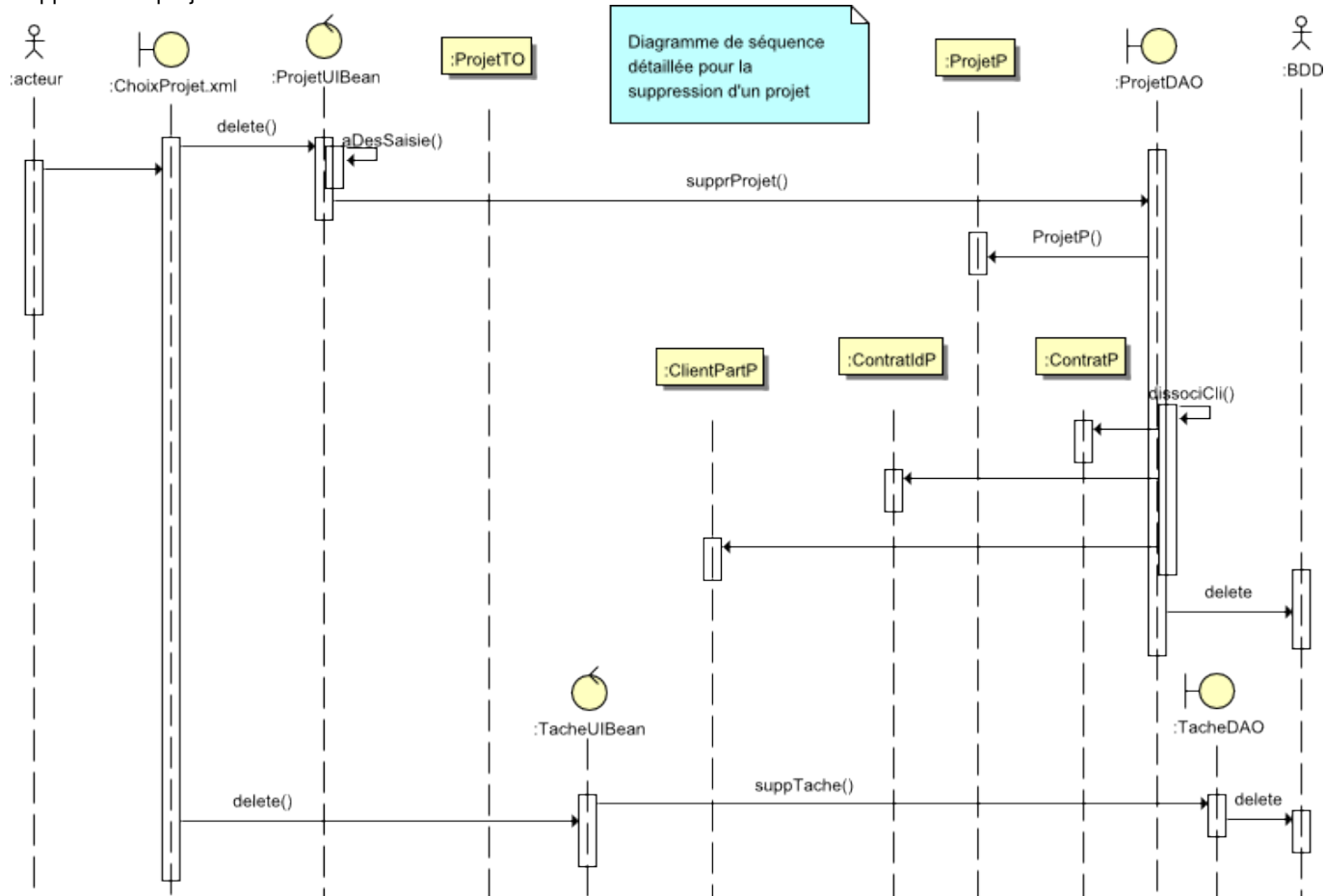
Création projet :



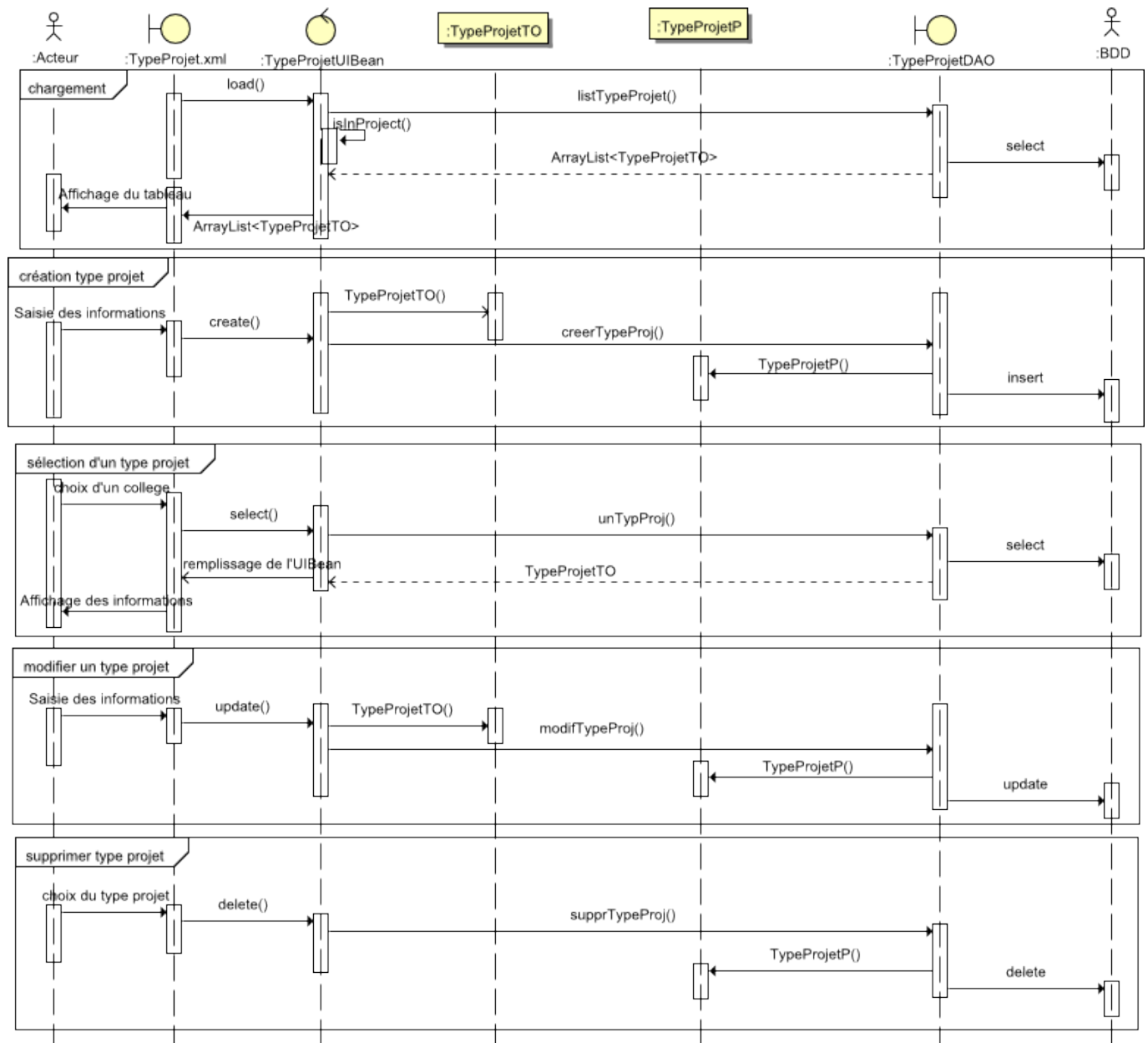
Modifier un projet



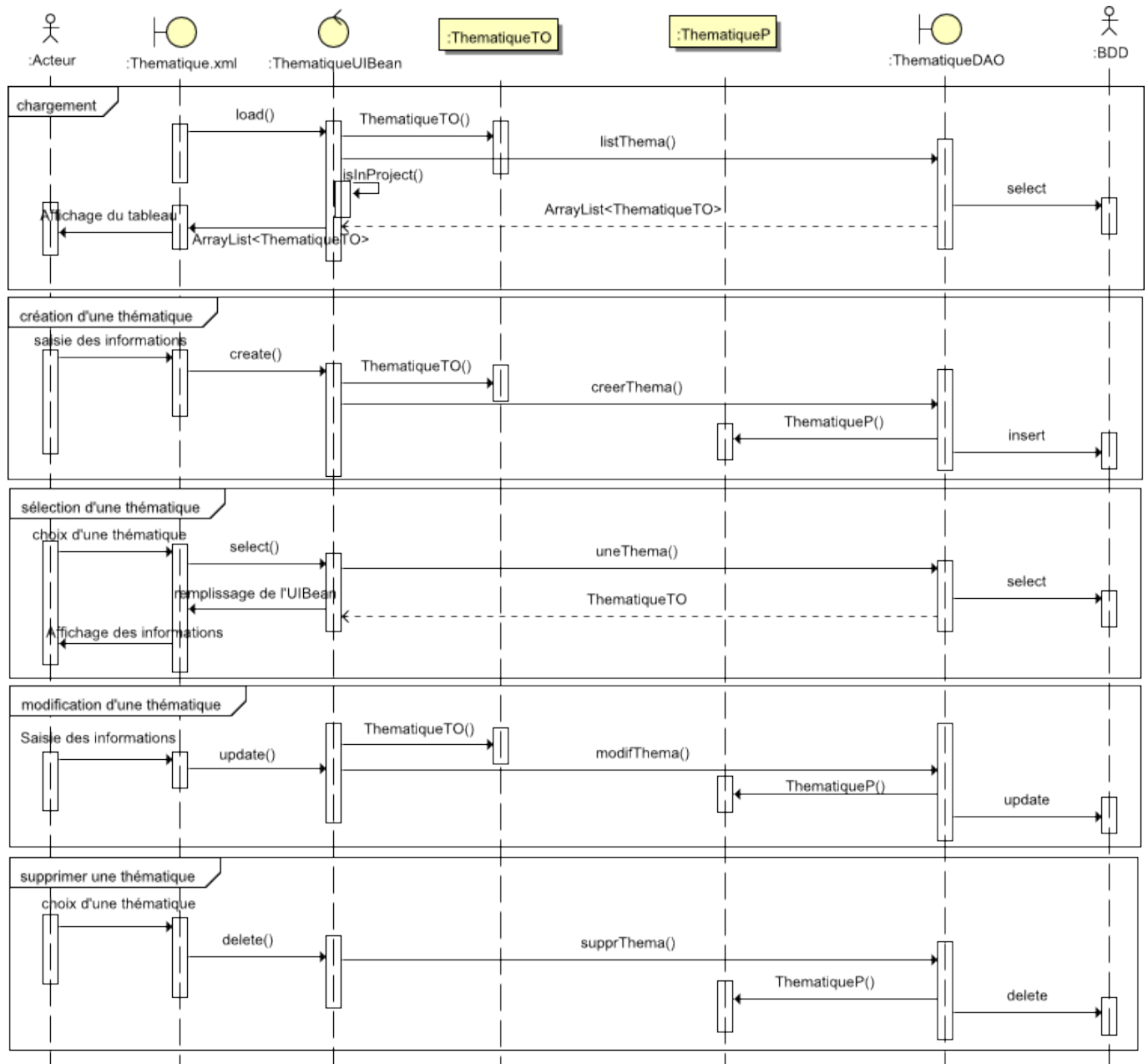
Supprimer un projet



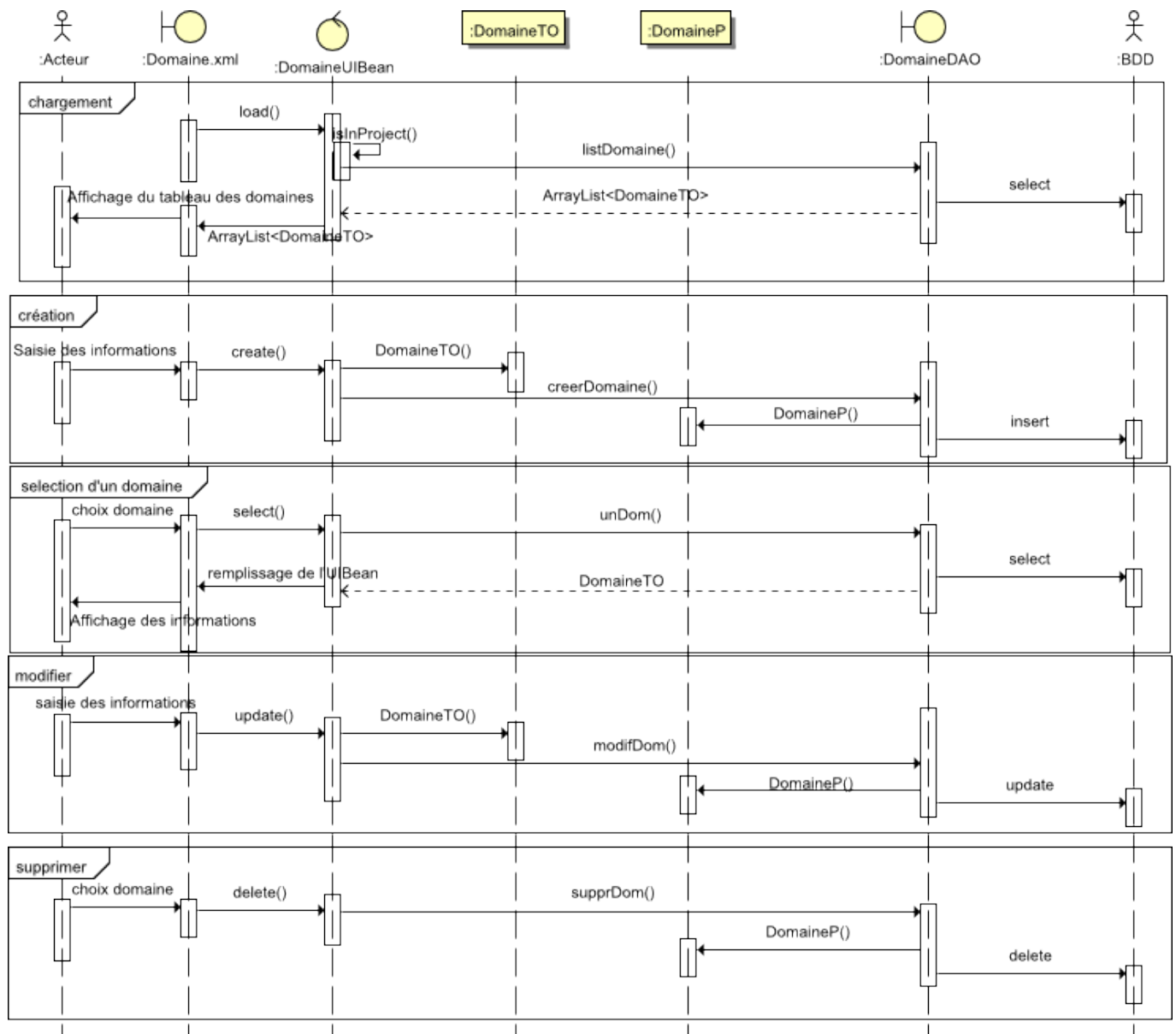
Gestion des Types de Projet :



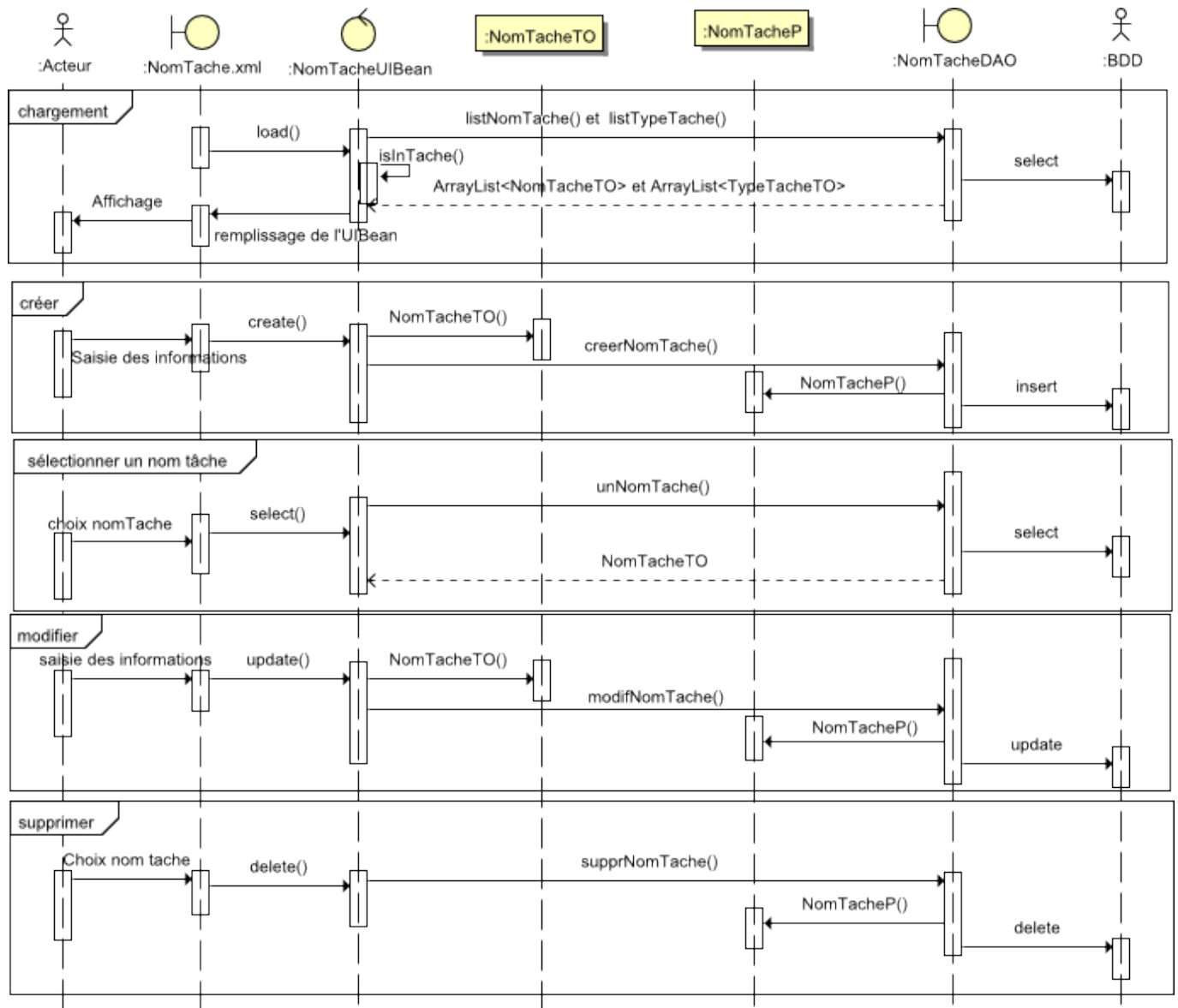
Gestion des thématiques :



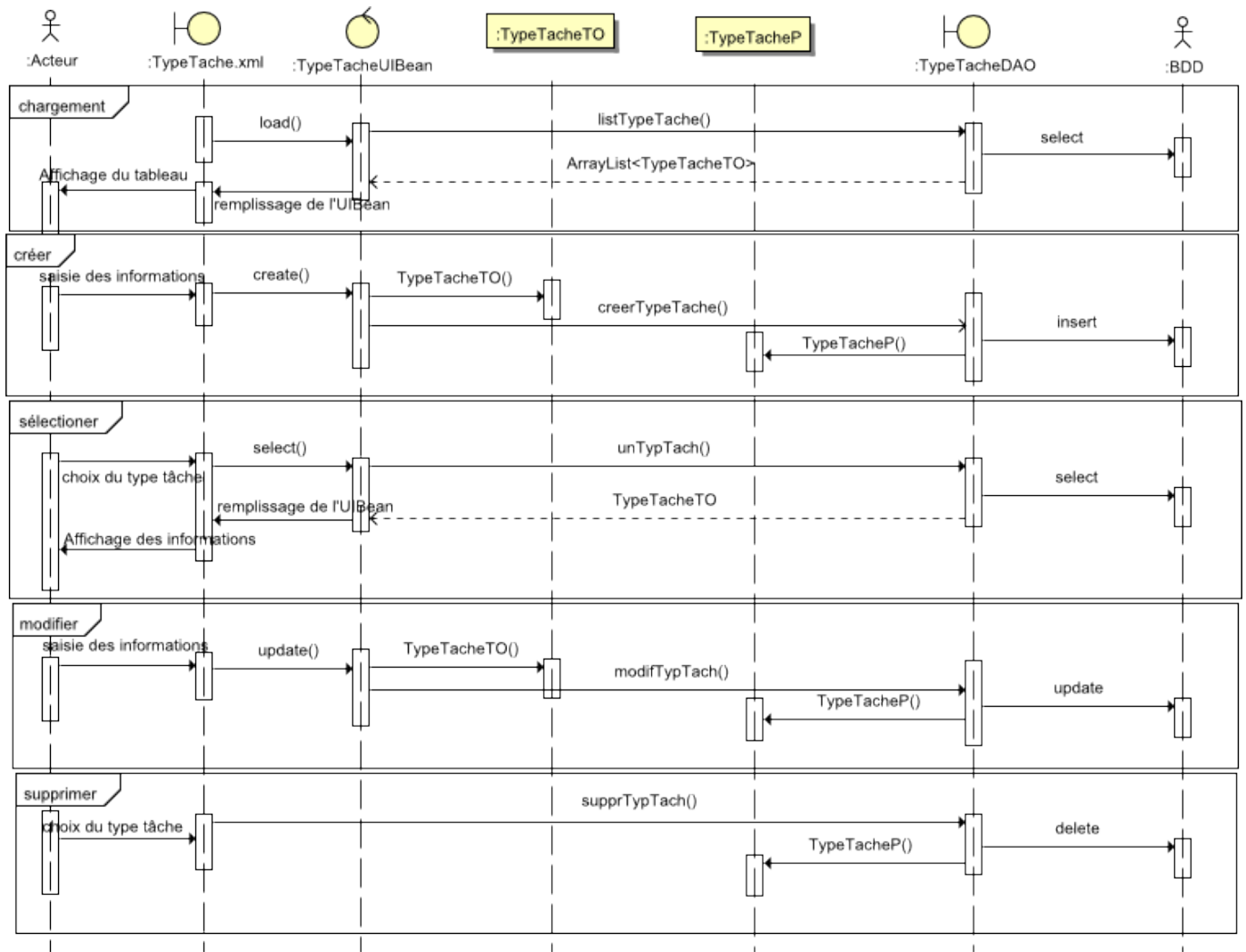
Gestion des domaines :



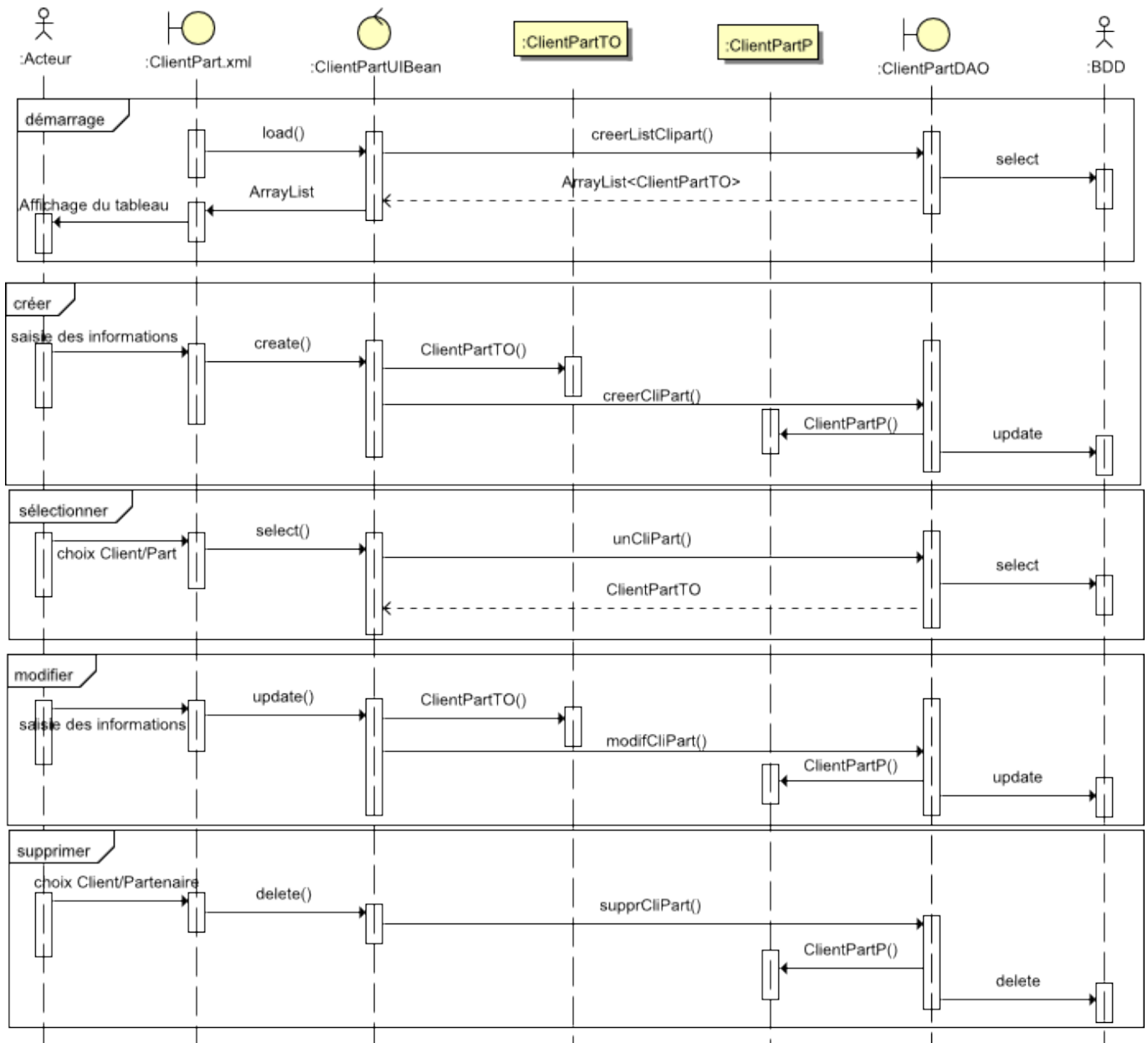
Gestion des tâches :



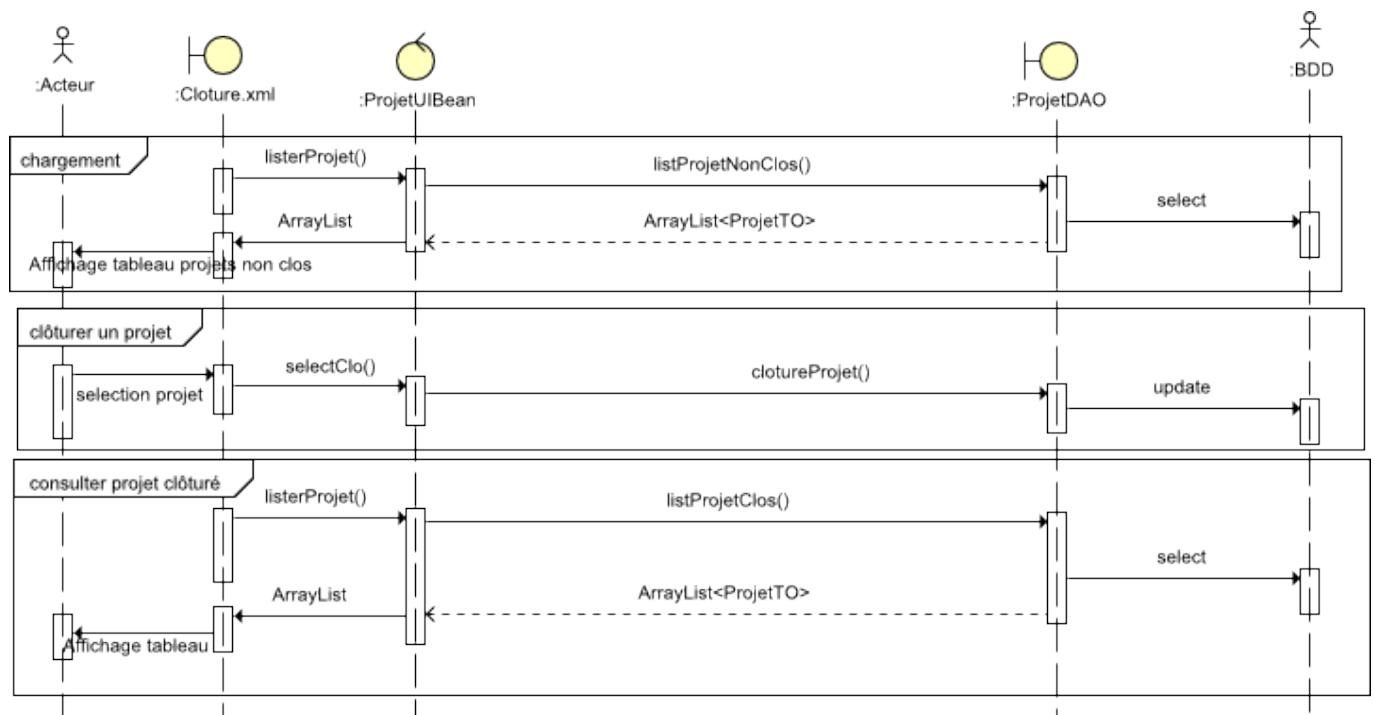
Gestion des Types de tâche



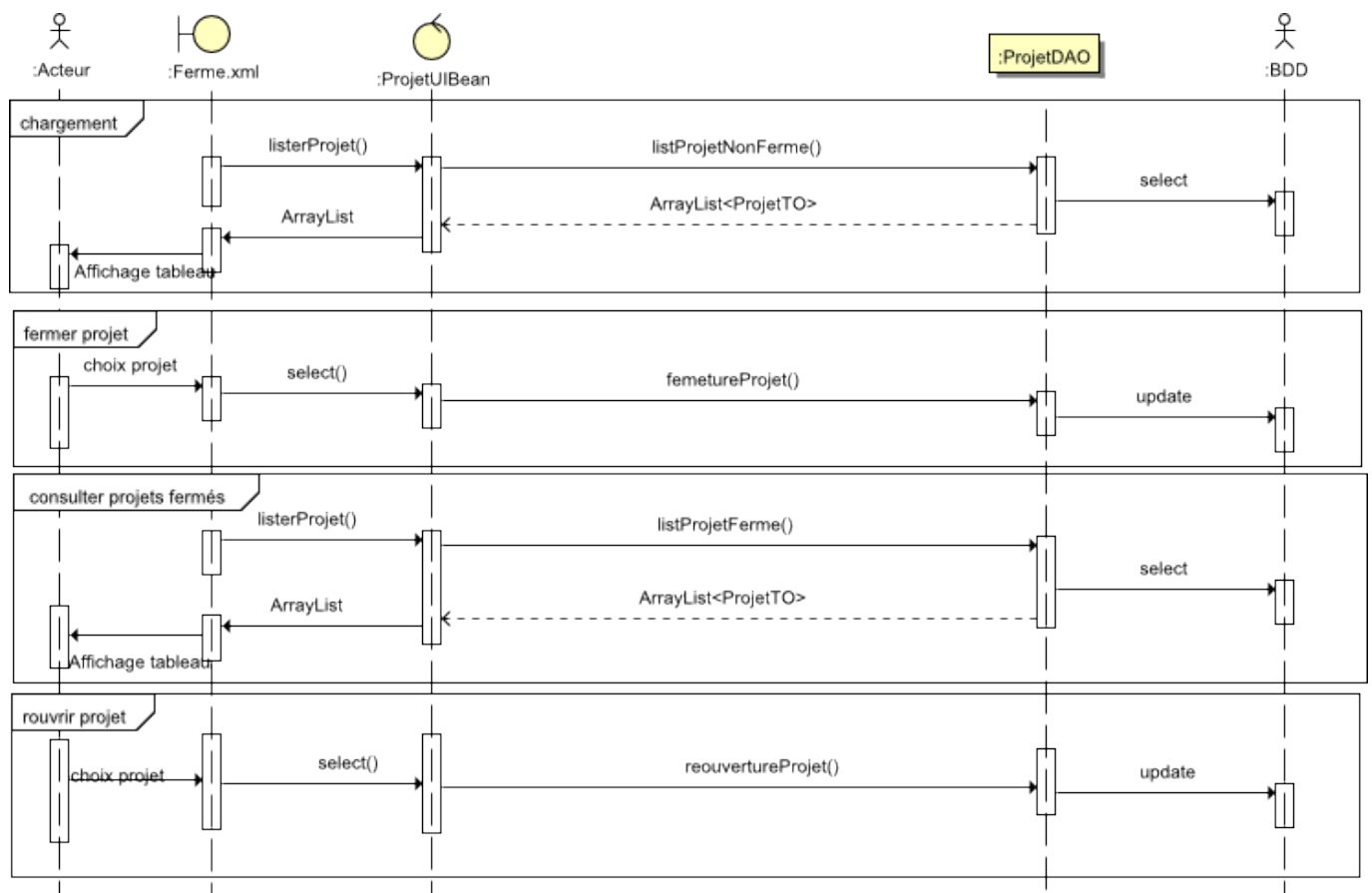
Gestion des Clients Partenaires



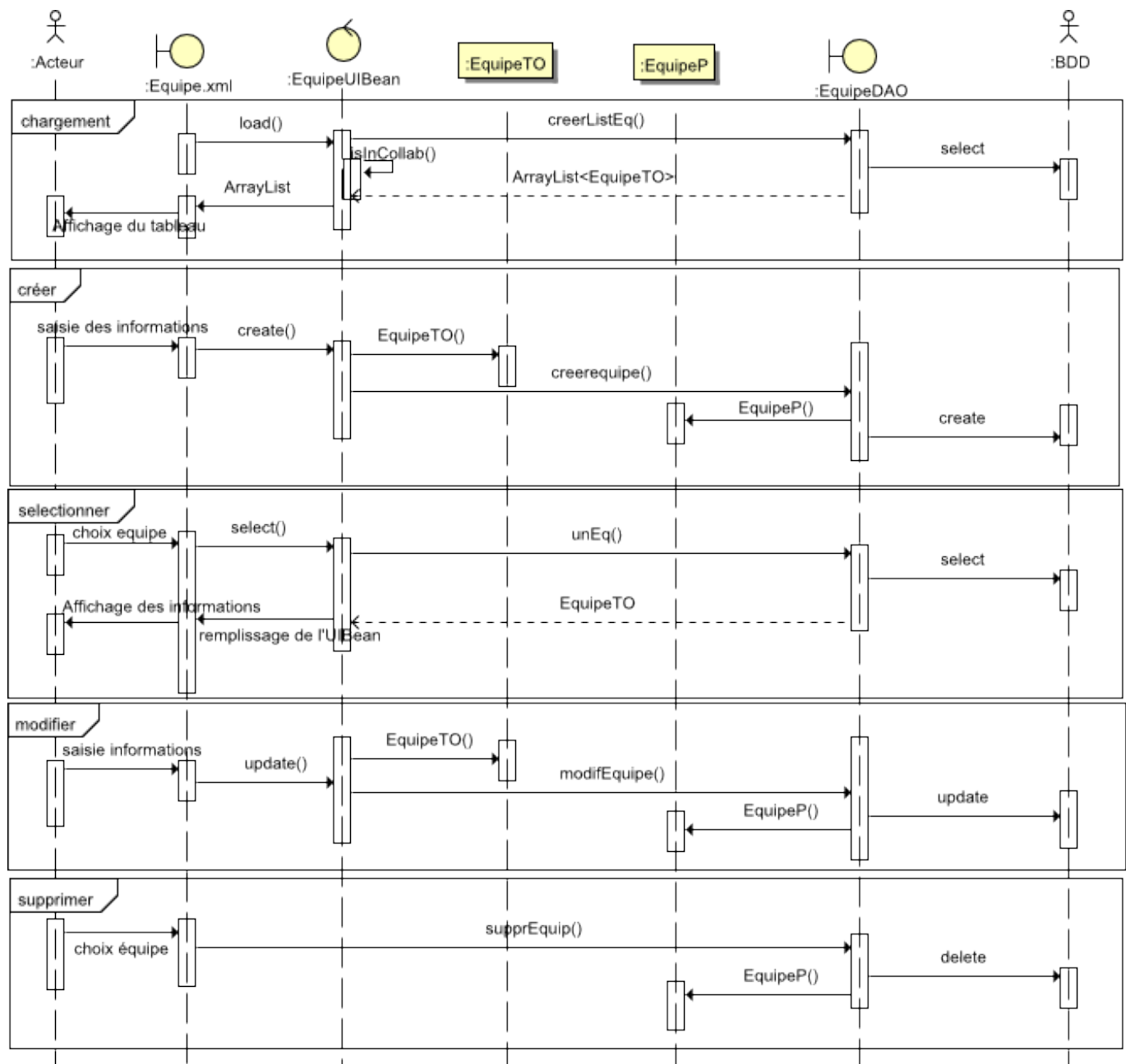
Clôture Projet



Fermeture Projet :

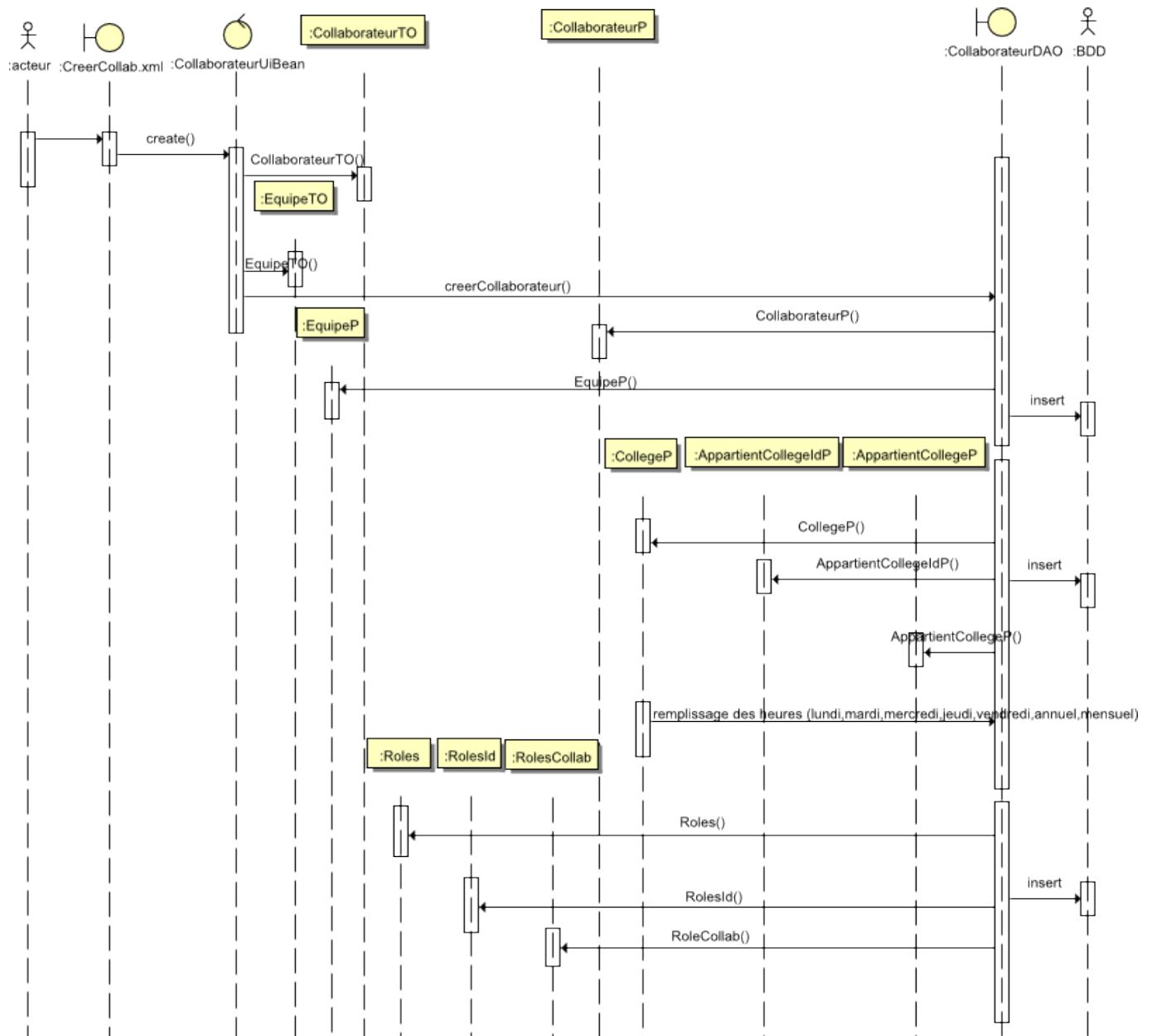


Gestion des équipes :

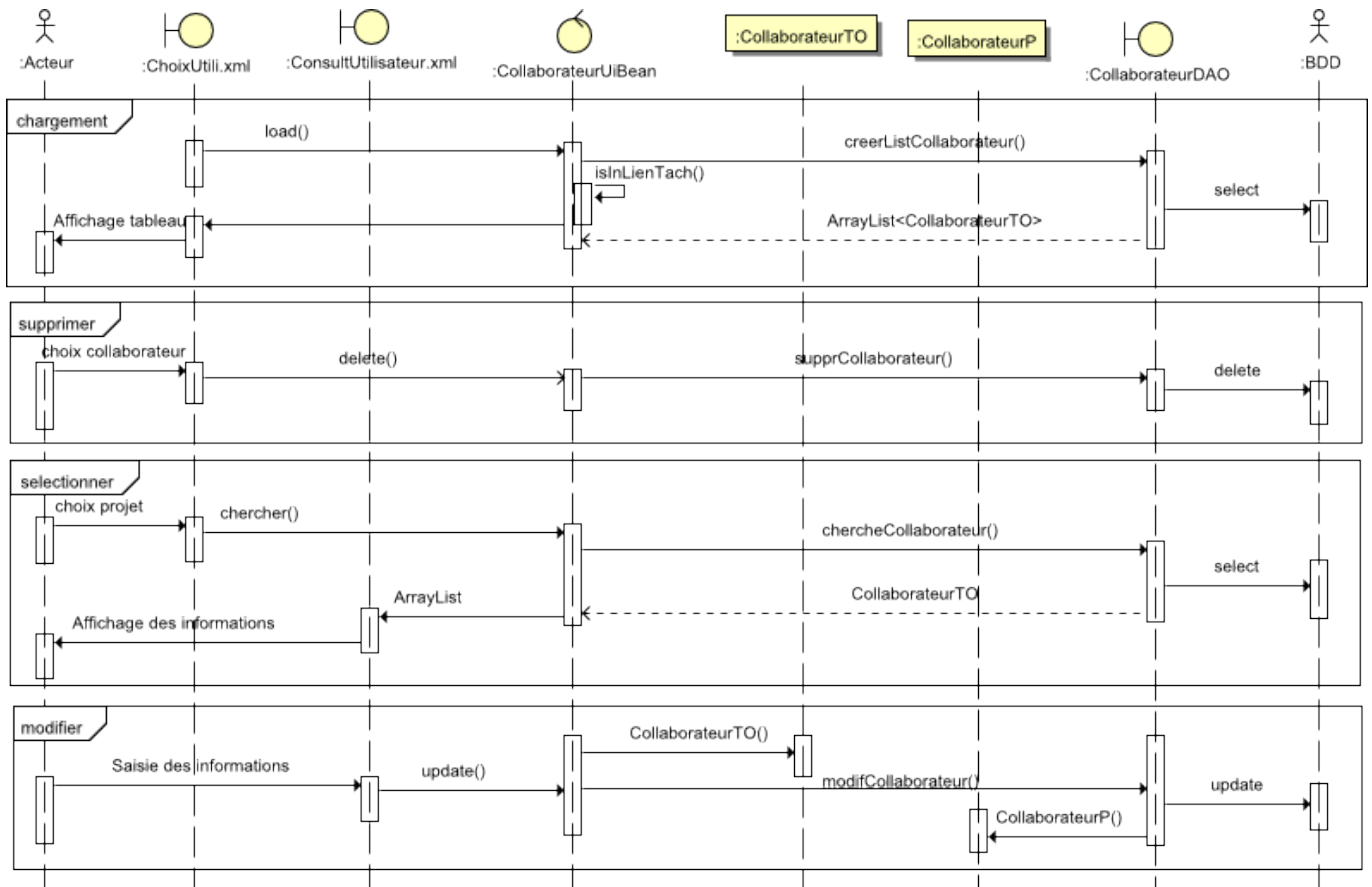


Gestion des utilisateurs

Création utilisateur :



Consulter, modifier ou supprimer un utilisateur :



Bilans

Tableau récapitulatif par types de projet Ici vous pouvez remarquer la présence de RecapTypeHibernate qui n'est pas un objet servant au mapping d'hibernate, c'est à dire qu'il ne correspond pas à une table de la Bdd. Cet objet sert à transformer les informations provenant de la base de données. Il sont ensuite rassemblés pour être transformés en RecapTO.

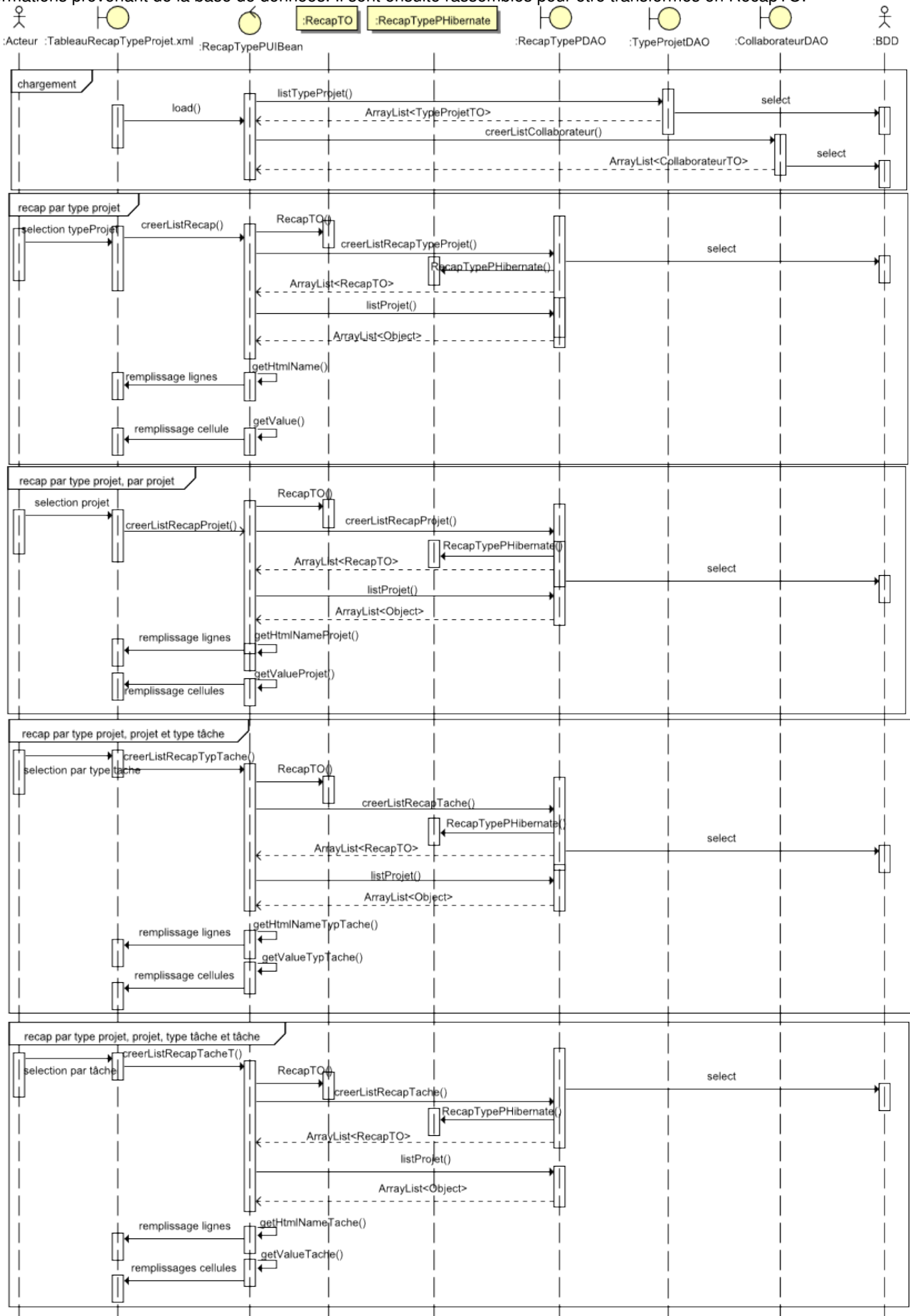


Tableau récapitulatif par thématique :

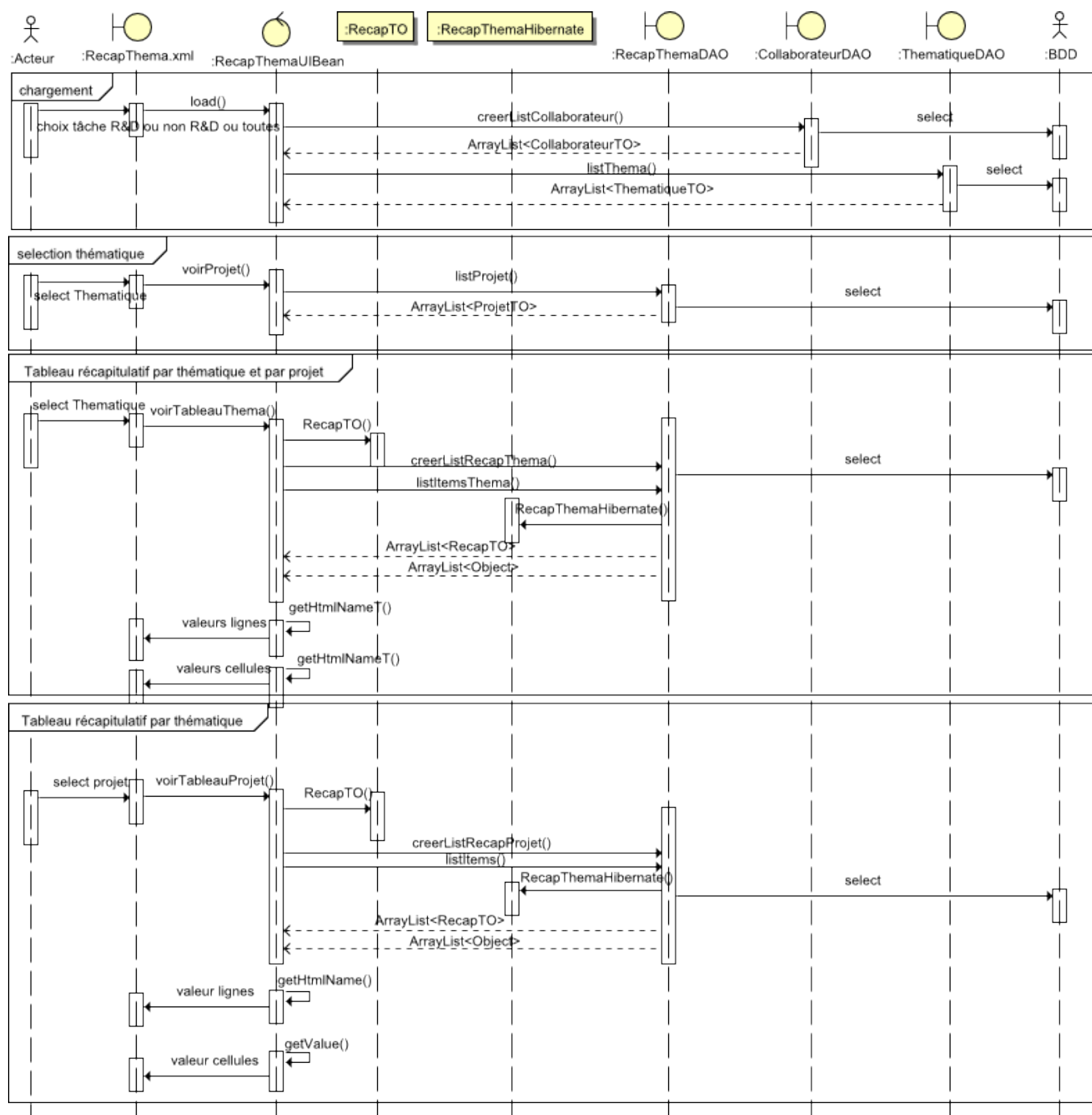


Tableau récapitulatif par Domaines :

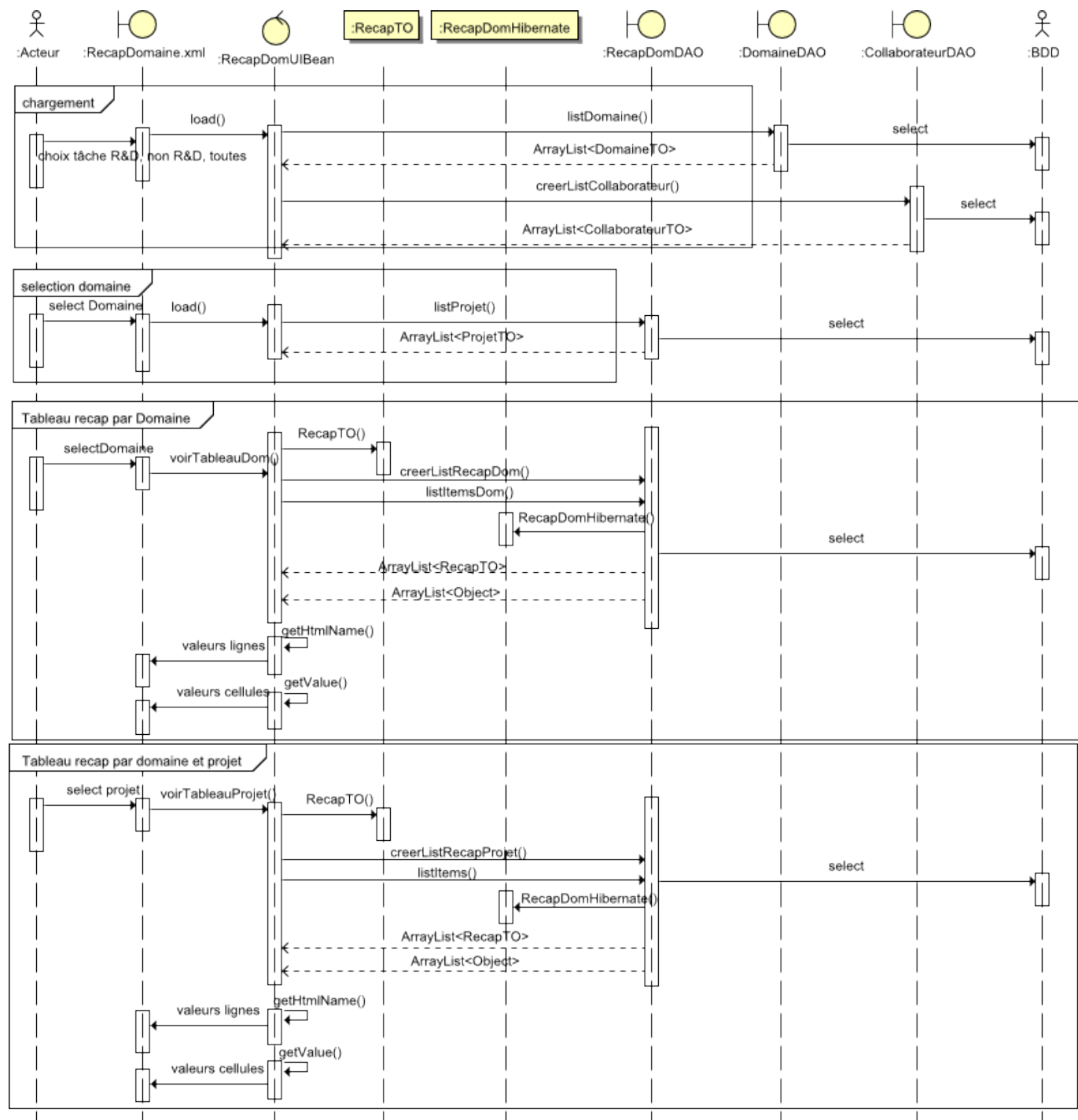
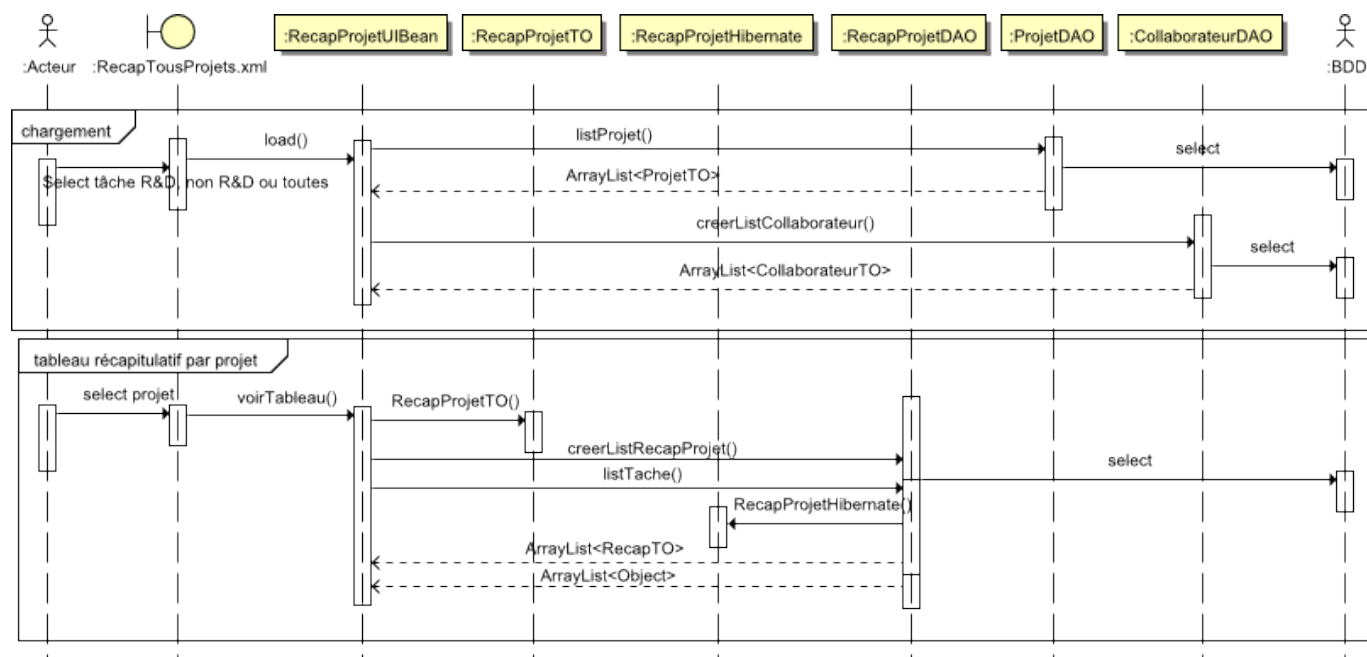
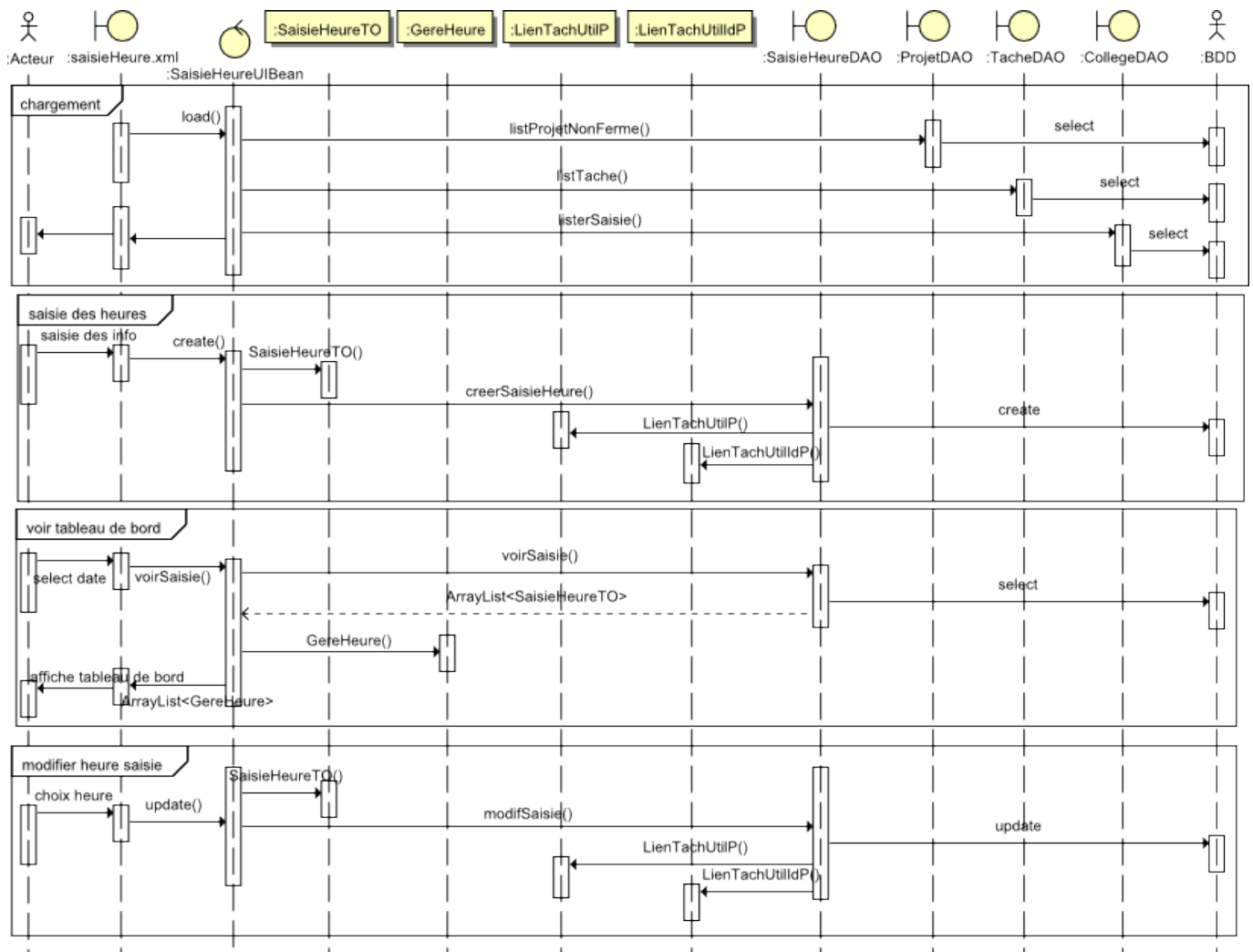


Tableau récapitulatif par projet (uniquement) :

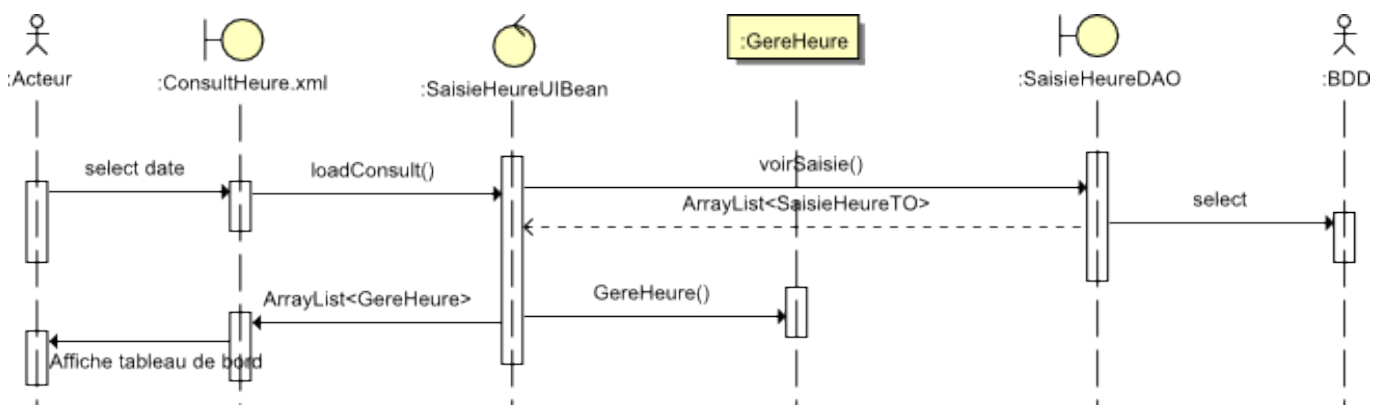


Gestion des heures

Saisie des heures :

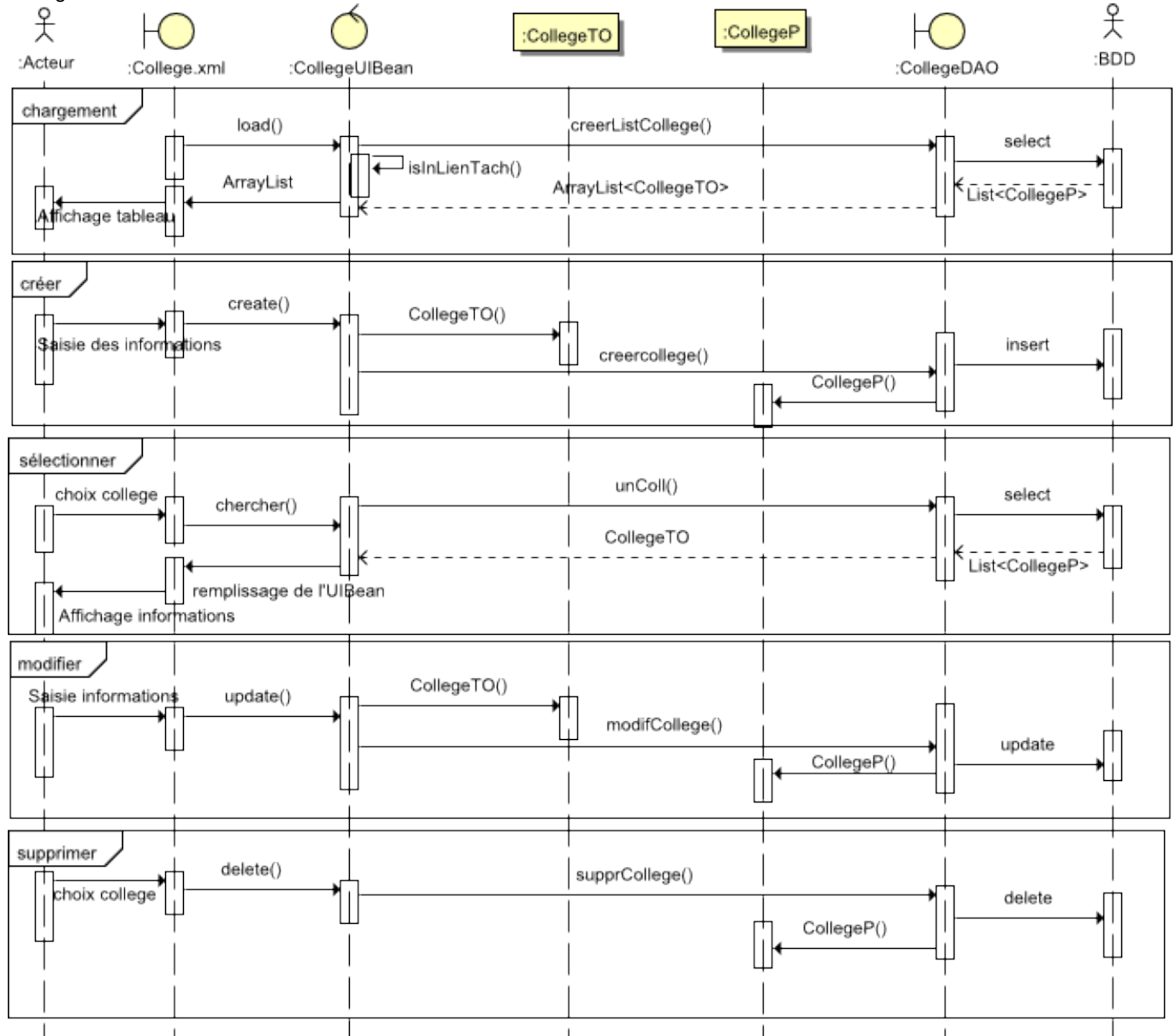


Consultation tableau de bord :



Organisation du temps de travail :

Collège :



Alertes : Ici vous pouvez remarquer la présence de AlertesHibernate, cet objet sert à transformer les informations provenant de la Bdd.

