

Formation : Conception et mise en œuvre d'un logiciel

Etude de cas : Gestion de formations

Table des matières

1- Objectifs	2
2- Prérequis	2
3- Etude de cas : Gestion des formations.....	3
4- Etape 1 : Modélisation métier (business modeling)	3
5- Etape 2 : Définition des besoins du système informatique.....	6
6- Etape 3 : Analyse du domaine (partie statique)	9
a. Analyse linguistique de l'énoncé.....	10
a. Découpage du modèle métier statique en package.....	12
7- Etape 4 : Analyse du domaine (partie dynamique)	14
b. Diagramme de séquence	14
8- Etape 5 : Définition des itérations.....	15
9- Etape 6 : Définition de l'architecture système	16
a. Architecture en couches préliminaire.....	17
10- Etape 7 : Définition des opération système (itération 1)	18
11- Etape 8 : Diagrammes d'interactions (itération 1)	18
a. Diagrammes d'interaction des opérations système.....	18
b. Diagrammes d'interaction des opérations système (diagramme de communication).....	19
c. Diagramme de séquence de l'initialisation de la formation et création de son contenu	20
12- Etape 9 : Diagramme de classes de conception.....	20
13- Etape 10 : Retour sur l'architecture	21
14- Etape 11 : Passage au code objet.....	21

1- Objectifs

- ❖ L'objectif de cette étude de cas est de vous montrer :
 - Les diagrammes UML utilisés pour la modélisation métier.
 - Comment se servir de cette modélisation métier pour mieux définir les besoins informatiques.
 - Comment l'analyse linguistique permet d'aider à la modélisation métier.
 - Comment décrire une architecture en couches avec UML.
 - Comment utiliser les diagrammes de communication et de séquence pour décrire les interactions entre objets informatique et répartir les opérations.
 - Comment répercuter les décisions d'affectation des responsabilités aux objets dans les diagrammes de classe.
 - Comment traduire les diagrammes UML de conception détaillé en code java.
- ❖ L'étude de cas sera réalisée en respectant les étapes suivantes :
 - Etape 1 : Modélisation métier (business modeling).
 - Etape 2 : Définition des besoins du système informatique.
 - Etape 3 : Analyse du domaine (partie statique).
 - Etape 4 : Analyse du domaine (partie dynamique).
 - Etape 5 : Définition des itérations.
 - Etape 6 : Définition de l'architecture système.
 - Etape 7 : Définition des opération système (itération 1).
 - Etape 8 : Diagrammes d'interactions (itération 1).
 - Etape 9 : Diagramme de classes de conception.
 - Etape 10 : Retour sur l'architecture.
 - Etape 11 : Passage au code objet.

2- Prérequis

- Notions de base dans l'UML, notamment les diagrammes suivants :
 - Diagramme de cas d'utilisation ;
 - Diagramme d'activité ;
 - Diagramme de communication ;
 - Diagramme de séquence ;
 - Diagramme de classe.
- L'outil **Eclipse Papyrus**. Vous pouvez utiliser un autre outil. Par exemple :
 - Lucidchart (<https://www.lucidchart.com/pages/>)

- StarUML (<http://staruml.io/>)
- Umbrello (<https://umbrello.kde.org/>)
- UML Designer tool (<http://www.uml designer.org/>)
- Altova(<https://www.altova.com/umodel>)
- Umple (<https://cruise.eecs.uottawa.ca/umple/>)
- Visual Paradigm (<https://www.visual-paradigm.com/>)

3- Etude de cas : Gestion des formations

❖ L'énoncé :

Dans le cadre de l'amélioration qu'elle veut apporter à son SI, une entreprise souhaite modéliser, dans un premier temps, le processus de formation de ses employés afin que quelques-unes de leurs tâches soient informatisées :

1. Le processus de formation est initialisé lorsque **le responsable de formation** reçoit une demande de formation **d'un employé**. Cette demande est instruite par le responsable de formation qui la qualifie et transmet son accord ou son désaccord à l'intéressé.
2. En cas d'accord, le responsable recherche dans le catalogue des formations agréées, un stage qui correspond à la demande. Il informe l'employé du contenu de la formation et lui propose une liste des prochaines sessions. Lorsque l'employé a fait son choix, le responsable formation inscrit le participant à la session auprès de **l'organisme de formation** concerné.
3. En cas d'empêchement, l'employé doit informer le responsable de formation au plus tôt pour annuler l'inscription ou la demande.
4. A la fin de la formation, l'employé doit remettre au responsable de formation une appréciation sur le stage qu'il a effectué, ainsi qu'un document justifiant de sa présence.
5. Le responsable de formation contrôle par la suite, la facture que l'organisme de formation lui a envoyée avant de la transmettre **au comptable achats**.

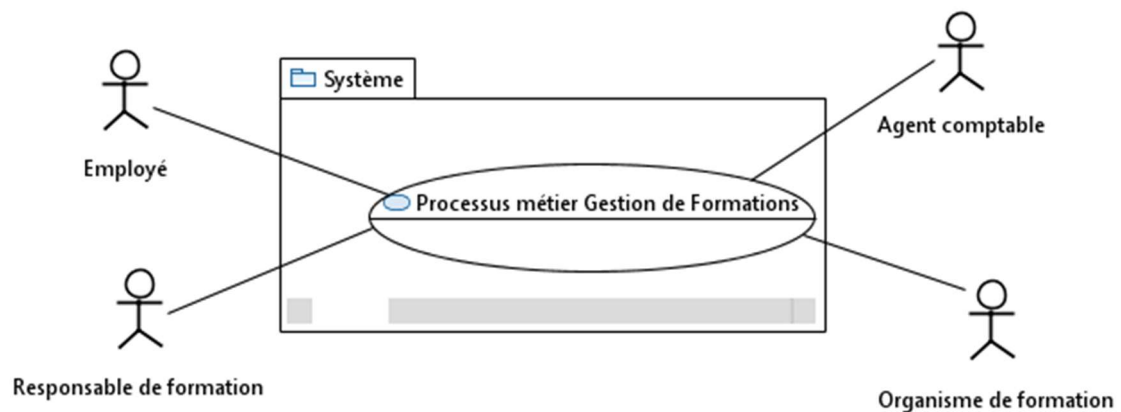
4- Etape 1 : Modélisation métier (business modeling)

Etape 1.1 : Modélisation d'un processus métier

Modéliser le processus de formation et ses acteurs

- ✓ Quel est le processus métier ?
- ✓ Quels sont les acteurs internes à l'entreprise ?
- ✓ Quels sont les acteurs externes à l'entreprise ?
- ✓ Faire le schéma de cas d'utilisation.

Solution :



Etape 1.2 : diagramme d'activité pour modéliser un processus métier

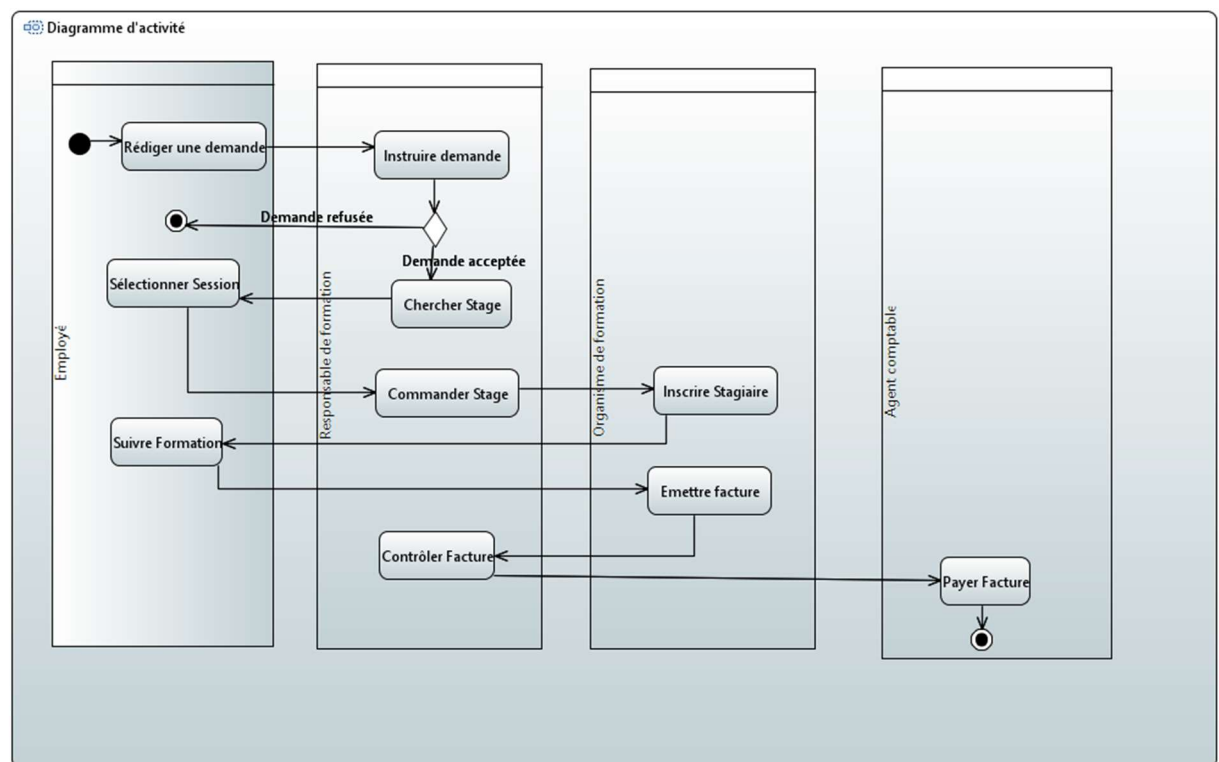
Le processus de formation comporte un ensemble d'activités ordonnées dans le temps et affectées à des acteurs identifiés précédemment.

Travail demandé :

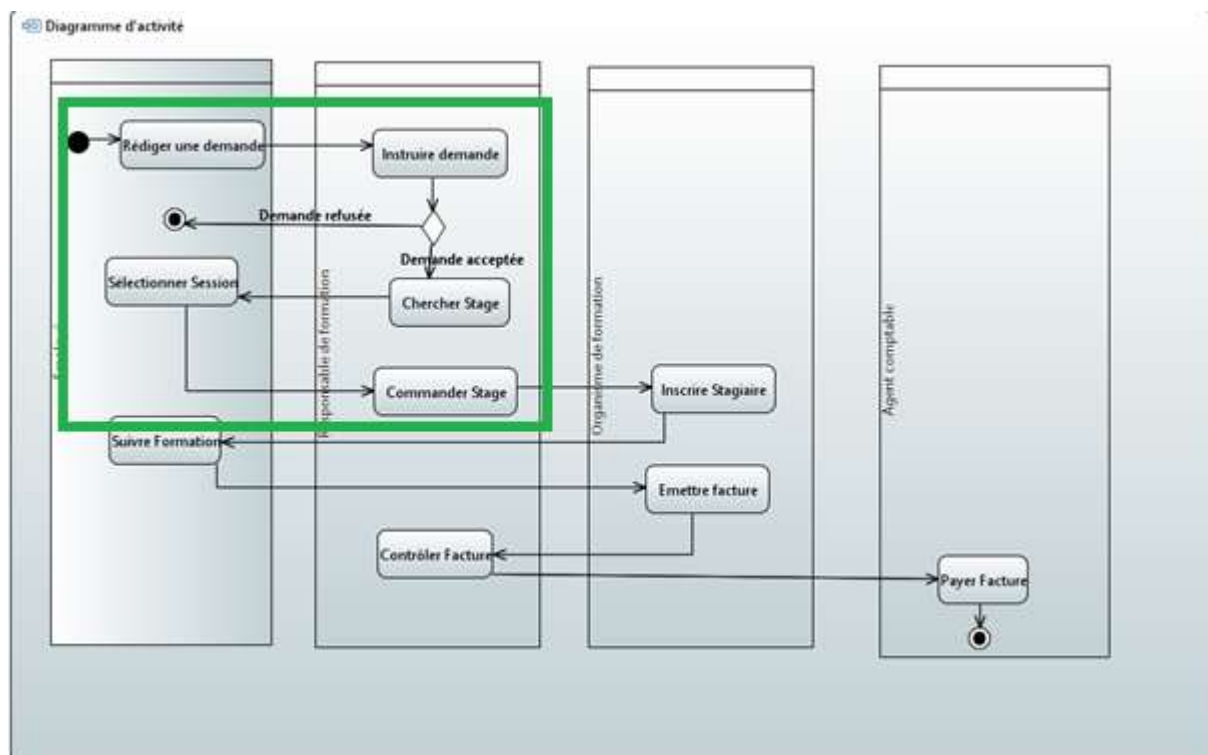
Modéliser le processus de formation avec un diagramme d'acteur (diagramme d'activité)

- ✓ Utiliser les couloirs (ou ***swimlanes***) pour affecter les responsabilités aux acteurs
- ✓ Compléter le diagramme, en y ajoutant la création et le changement d'état des entités métier, suite à la réalisation des activités.
- ✓ Le diagramme ainsi obtenu fait le pont entre les trois axes de modélisation : fonctionnel, dynamique et statique.

Solution :



✓ Activité du processus de formation à informatiser :



5- Etape 2 : Définition des besoins du système informatique

- ✓ **Le cahier des charges fonctionnel** du système informatique est déduit à partir du diagramme d'activité.
- ✓ En effet, la définition des tâches à informatiser, est réalisée par sélection de certaines activités du modèle métier.
- ✓ Le système doit permettre d'initialiser une demande de formation et de suivre cette demande jusqu'à l'inscription effective d'un employé. Il doit donc automatiser les activités métier suivantes :
 - Rédiger une demande (employé)
 - Instruire une demande (responsable formation)
 - Chercher un stage (responsable formation)
 - Sélectionner une session (employé)
 - Commander un stage (responsable formation)

a. Etape 2.1 : Diagramme de cas d'utilisation

Travail demandé :

Élaborer le diagramme de cas d'utilisation du système informatique de gestion des demandes de formation en tenant compte :

- De la liste des activités métier recensées précédemment.
- Du fait qu'un employé peut annuler une demande ou une inscription à une session.
- Du fait que le système gère un catalogue de formation auquel les employés peuvent accéder en lecture et le responsable de formation en écriture.

Solution

Demande une formation : L'employé peut consulter le catalogue et sélectionner un thème, ou une formation, ou même une session particulière. La demande est automatiquement enregistrée par le système et transmise au responsable de formation par e-mail. Si l'employé n'a pas choisi une session, mais simplement une formation ou un thème, le responsable formation va consulter le catalogue et sélectionner les sessions qui paraissent correspondre le mieux à la demande. Cette sélection sera transmise par e-mail à l'employé, qui pourra ainsi faire une nouvelle demande plus précise.

Commander un stage : une fois qu'un employé a demandé une formation, et que celle-ci est acceptée, le responsable de formation va utiliser le système pour envoyer automatiquement par fax ou e-mail une demande d'inscription sous forme de bon de commande à l'organisme de formation.

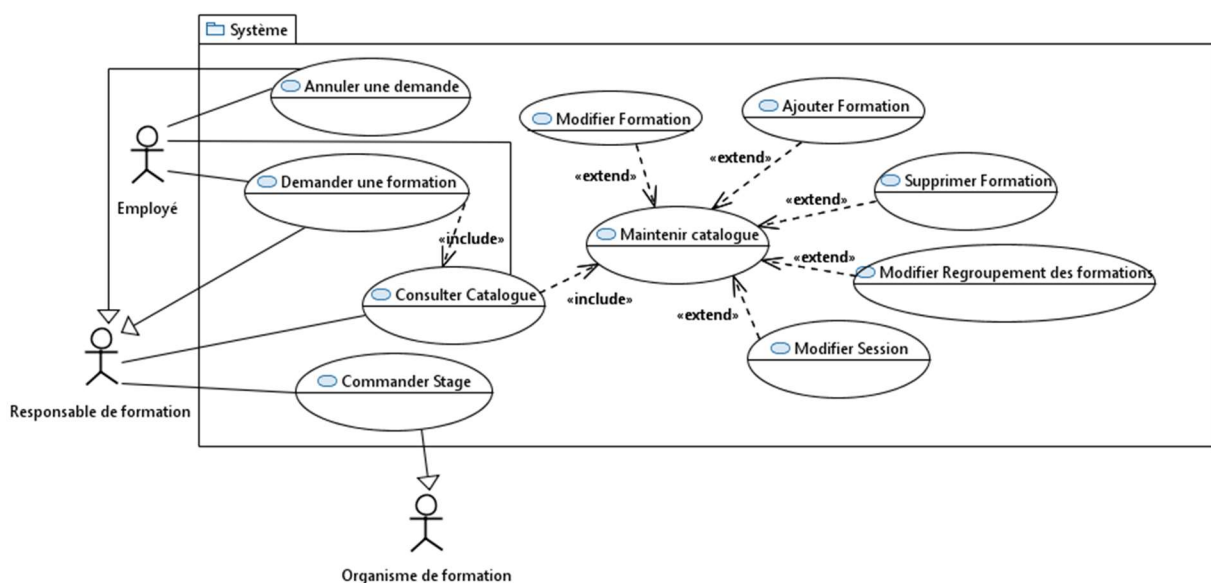
Annuler une demande : l'employé peut consulter l'état de ses demandes de formation en cours et les annuler individuellement. Le responsable de formation est automatiquement averti par e-mail.

Maintenir le catalogue : le responsable de formation peut introduire une nouvelle formation dans le catalogue, modifier une formation existante ou supprimer une formation qu'un organisme a abandonné. Il peut également modifier les regroupements de formations qui ont été faits par thèmes. Il a aussi la possibilité de mettre à jour les dates et lieux des sessions.

Consulter le catalogue : pour demander une formation et pour maintenir le catalogue, le système doit proposer une fonctionnalité de base de consultation du catalogue. Cette fonctionnalité peut donc être factorisée dans un nouveau cas d'utilisation.

Enfin, pour ne pas surcharger le modèle, nous ne représenterons pas le processus d'identification de l'employé ou du responsable formation.

Solution : Diagramme de cas d'utilisation du système gestion des demandes de formation



b. Etape 2.2 : Description essentielle d'un cas d'utilisation

Travail demandé :

- ✓ Rédiger une description essentielle du cas d'utilisation « *Ajouter Formation au catalogue* »
- ✓ Respecter le format suivant :

Sommaire d'identification	
Titre	
Acteurs	
Date de création	
Date de mise à jour	
Version	
Responsable	

Description des scénarii	
Pré conditions	
Poste conditions	
Scénario nominal : séparation des actions des acteurs et du système en deux colonnes.	
Enchaînement alternatif	
Exception	
Règles de gestion	

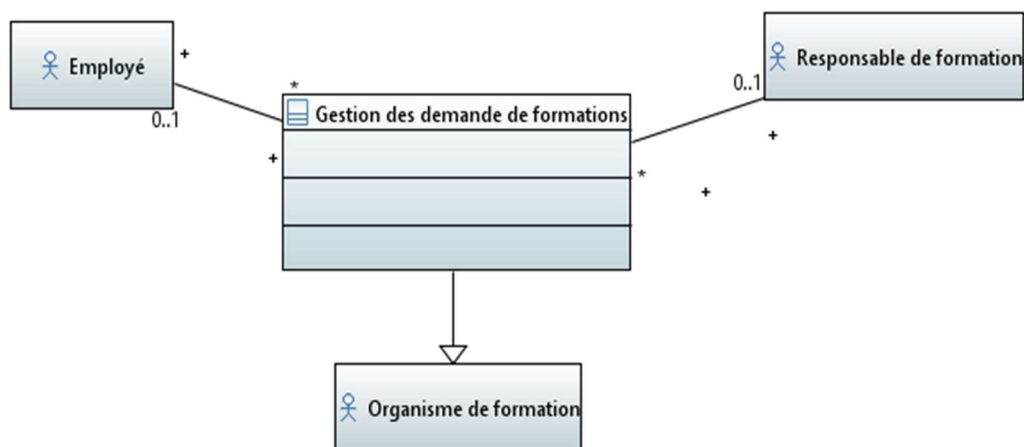
Solution :

Sommaire d'identification	
Titre	Ajouter Formation au catalogue
Acteurs	Responsable de formation
Date de création	02/01/2020
Date de mise à jour	20/01/2020
Version	2.0
Responsable	M. Foulane Mohamed
Description des scénarii	
Pré conditions	L'utilisateur doit être authentifié
Poste conditions	La formation est bien enregistrée au niveau du catalogue.
Scénario nominal	<p>1. L'utilisateur clique sur le menu « Maintenir Catalogue ».</p> <p>2. Le Système affiche la page pour maintenir le catalogue.</p> <p>3. L'utilisateur clique sur le menu « Ajouter une formation ».</p> <p>4. Le système affiche la page contenant un formulaire pour saisir les données de la formation (Description de la formation, un résumé, les objectifs principaux, le thème de la formation).</p> <p>5. L'utilisateur saisit la description de la formation, le résumé de la formation, les objectifs et choisit le thème et clique sur le bouton valider.</p> <p>6. Le système ajoute la formation selon les règles de gestion RG1 et RG2.</p>
Enchaînement alternatif	Rien à signaler.
Exception	<p>5a. Si la formation existe déjà, le système affiche un message d'erreur : « La formation dont la description est ... existe déjà ».</p> <p>5b. Si le thème est vide, le système affiche un message d'erreur : « Veuillez renseigner le thème de cette formation ».</p>
Règles de gestion	<p>RG1 : la description de la formation doit être unique.</p> <p>RG2 : Le thème ne doit pas être nul.</p>

c. Etape 2.3 : Diagramme de contexte statique

Travail demandé :

- ✓ Elaborer le diagramme de contexte statique du système
- ✓ Tenir compte des remarques suivante :
 - Le système de gestion de demande est fondamentalement multiutilisateur (intranet)
 - Le responsable formation est le seul utilisateur en modification à un moment donné.
 - Les organismes de formation n'ont pas accès au système : ils ne font que recevoir des commandes.



6- Etape 3 : Analyse du domaine (partie statique)

L'analyse lexicale de la description fonctionnelle du système est très indiquée pour préparer une première version du diagramme de classe.

L'énoncé doit néanmoins être repris d'une manière plus simple :

1. Le processus de formation est initialisé lorsque le responsable de formation reçoit une demande de formation d'un employé.
2. Cette demande est instruite par le responsable de formation qui la qualifie et transmet son accord ou son désaccord à l'intéressé.
3. En cas d'accord, le responsable recherche dans le catalogue des formations agréées, un stage qui correspond à la demande.
4. Il informe l'employé du contenu de la formation et lui propose une liste des prochaines sessions.
5. Lorsque l'employé a fait son choix, le responsable formation inscrit le participant à la session auprès de l'organisme de formation concerné.

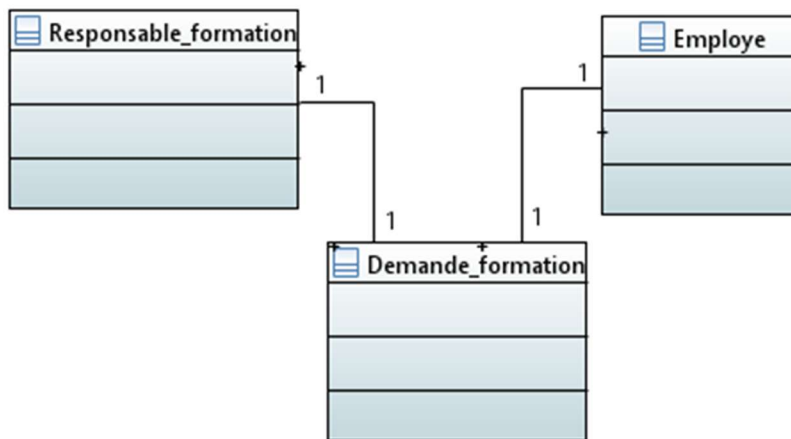
6. Le responsable formation contrôle par la suite, la facture que l'organisme de formation lui a envoyée avant de la transmettre au comptable achats.

a. Analyse linguistique de l'énoncé

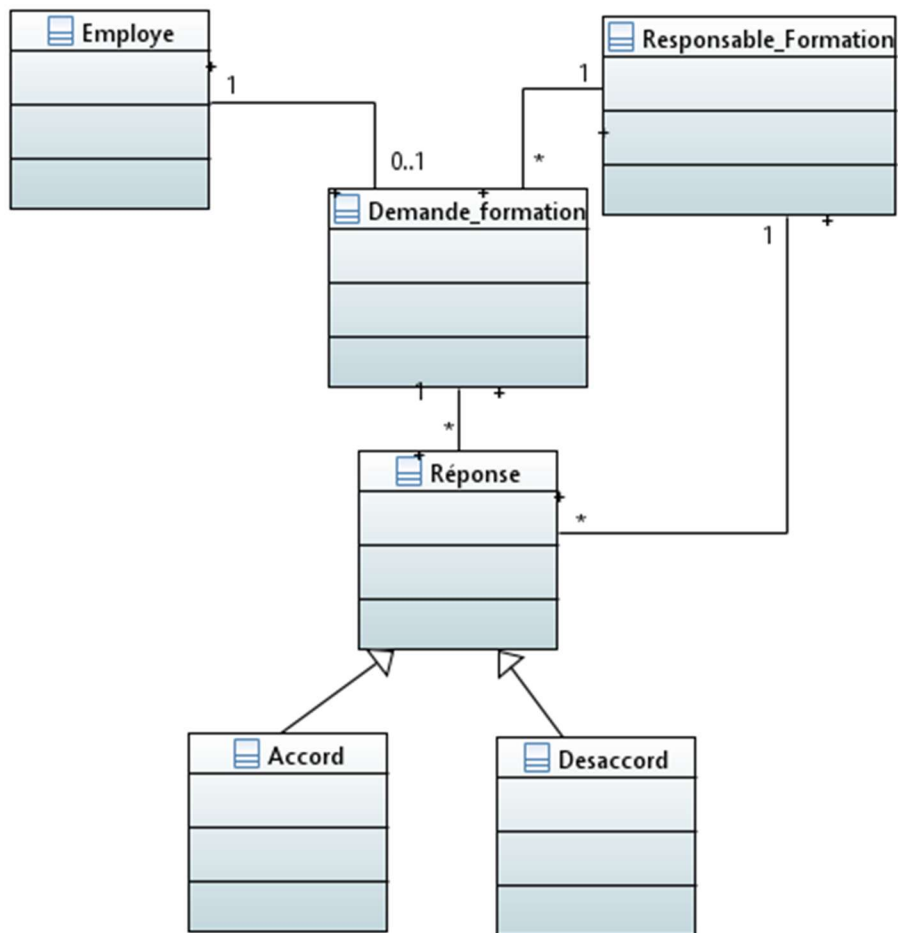
Travail demandé :

- ✓ Modéliser, sous forme de diagramme de classes, les phrases de 1 à 6 de l'énoncé.

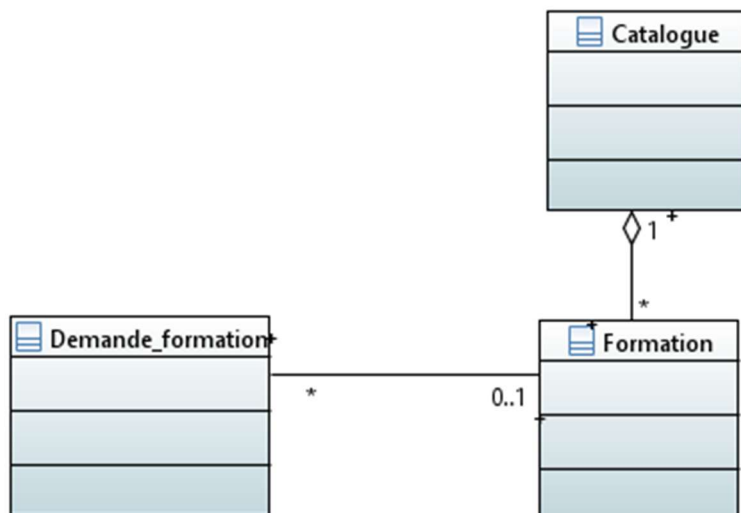
Solution : Modélisation statique de la phrase 1



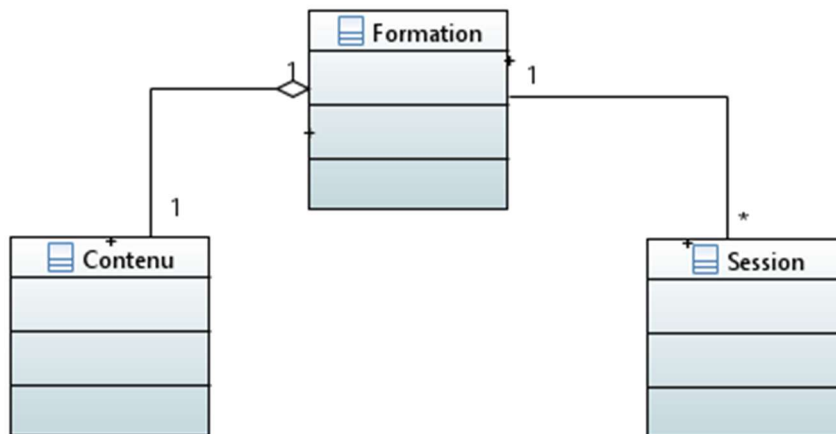
Solution : Modélisation statique de la phrase 2



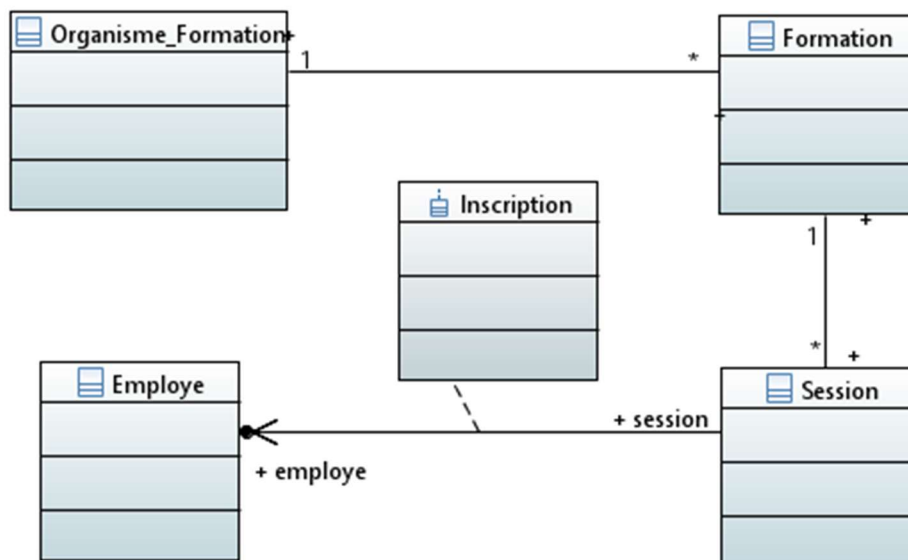
Solution : Modélisation statique de la phrase 3



Solution : Modélisation statique de la phrase 4



Solution : Modélisation statique de la phrase 5

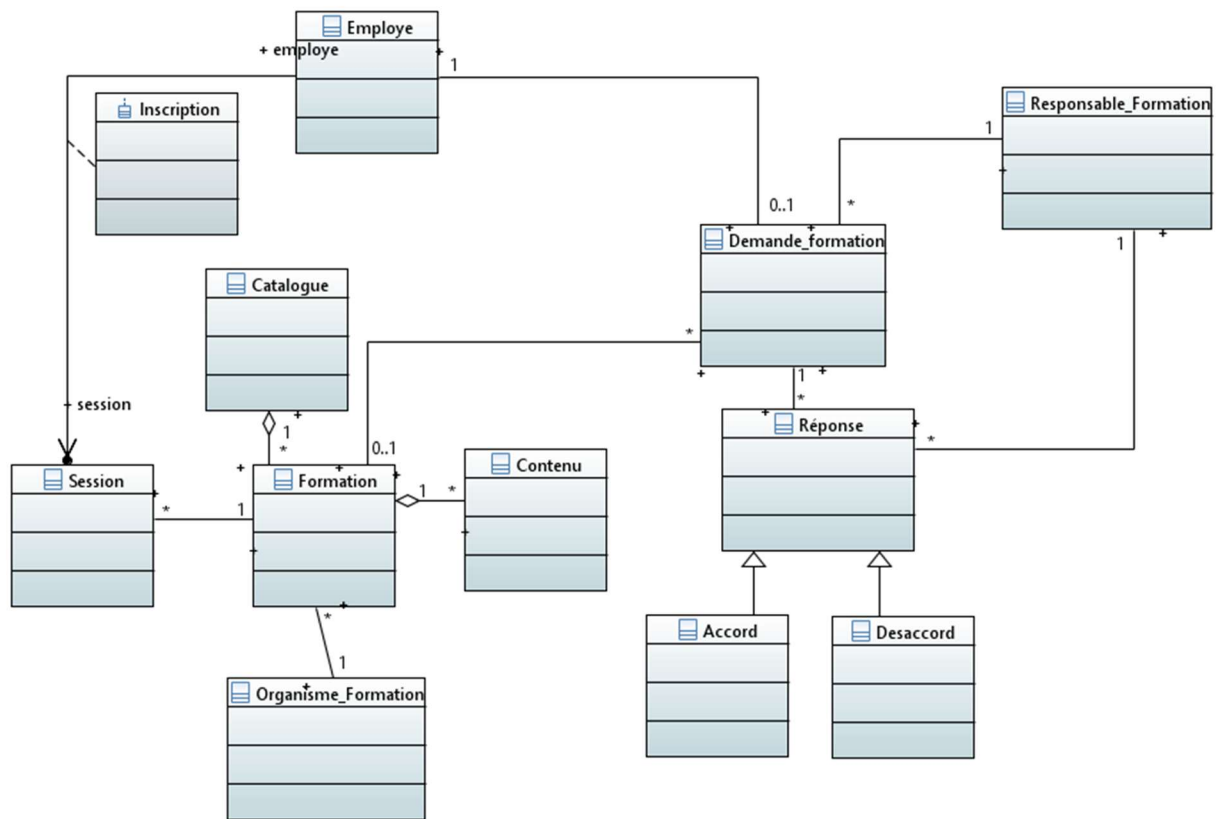


a. Découpage du modèle métier statique en package

Travail demandé :

- ✓ Rassembler tous les fragments précédents sur un même diagramme de classes.
- ✓ Découper ce diagramme en packages.
- ✓ Elaborer le diagramme de packages.

Solution : Assemblage des fragments précédents sur un même diagramme de classes



Solution : Découpage du diagramme en packages

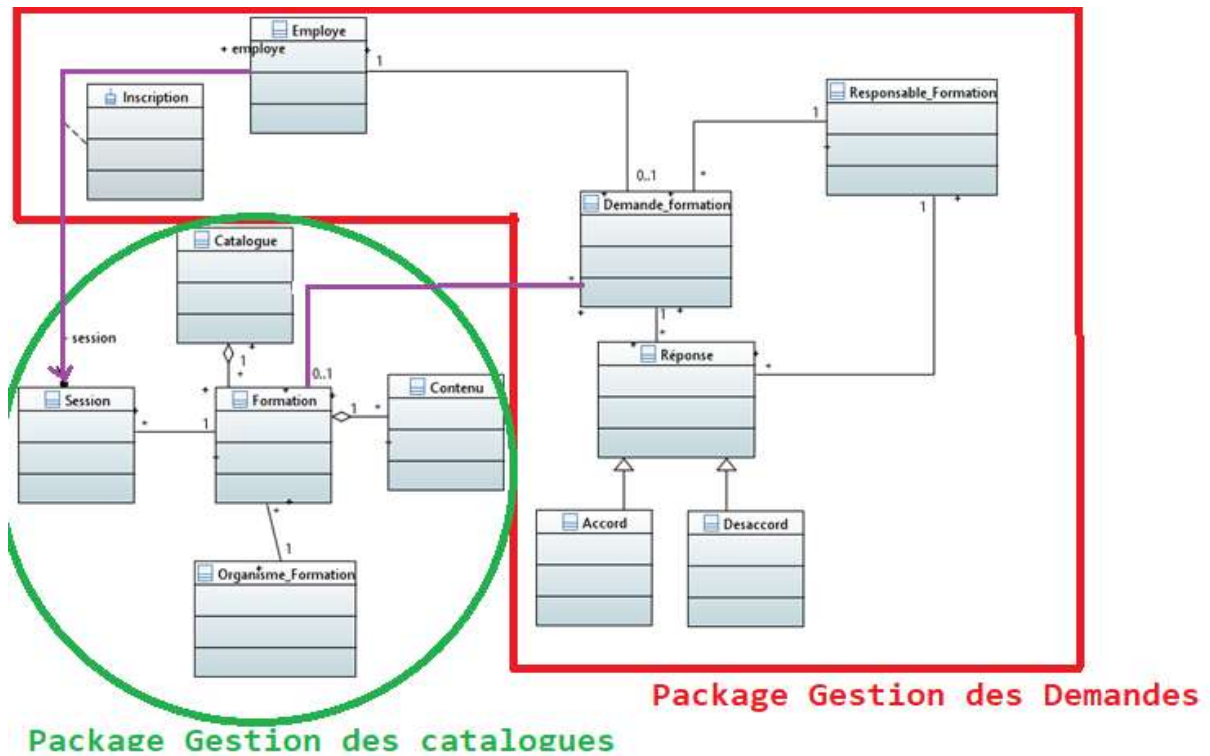


Diagramme de packages :



7- Etape 4 : Analyse du domaine (partie dynamique)

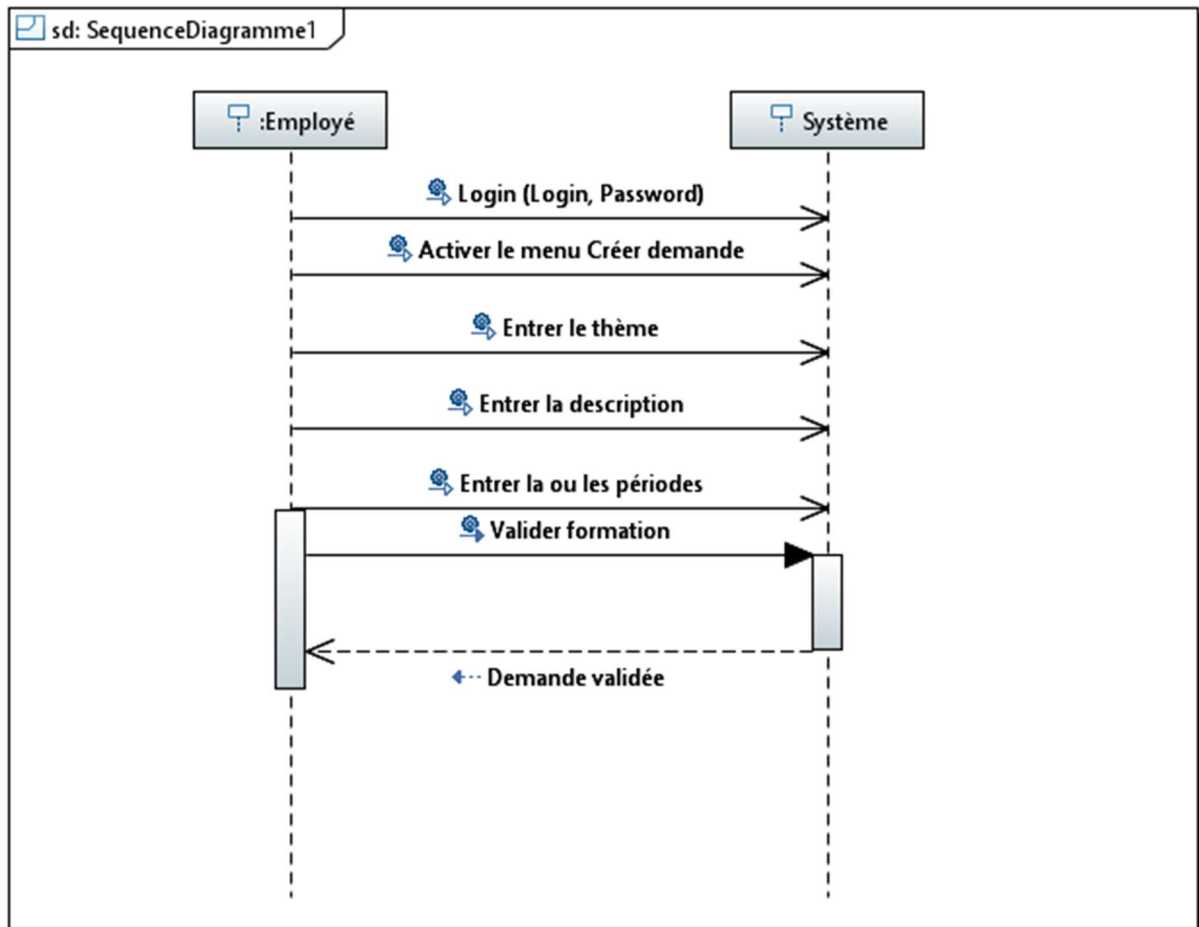
b. Diagramme de séquence

Travail demandé :

Faire le diagramme de séquence système (détaillé) du cas d'utilisation « demander une formation » en tenant compte des faits suivant :

- ✓ La demande est initiée par l'employé, via la création d'une demande de formation
- ✓ Lors de la création de cette demande, l'employé doit effectuer plusieurs sélections (thèmes, périodes, ...) avant de procéder à sa validation.
- ✓ Le système effectue une validation des données saisies.
- ✓ Plusieurs messages sont de types asynchrone.

Solution :



8- Etape 5 : Définition des itérations

Objectif :

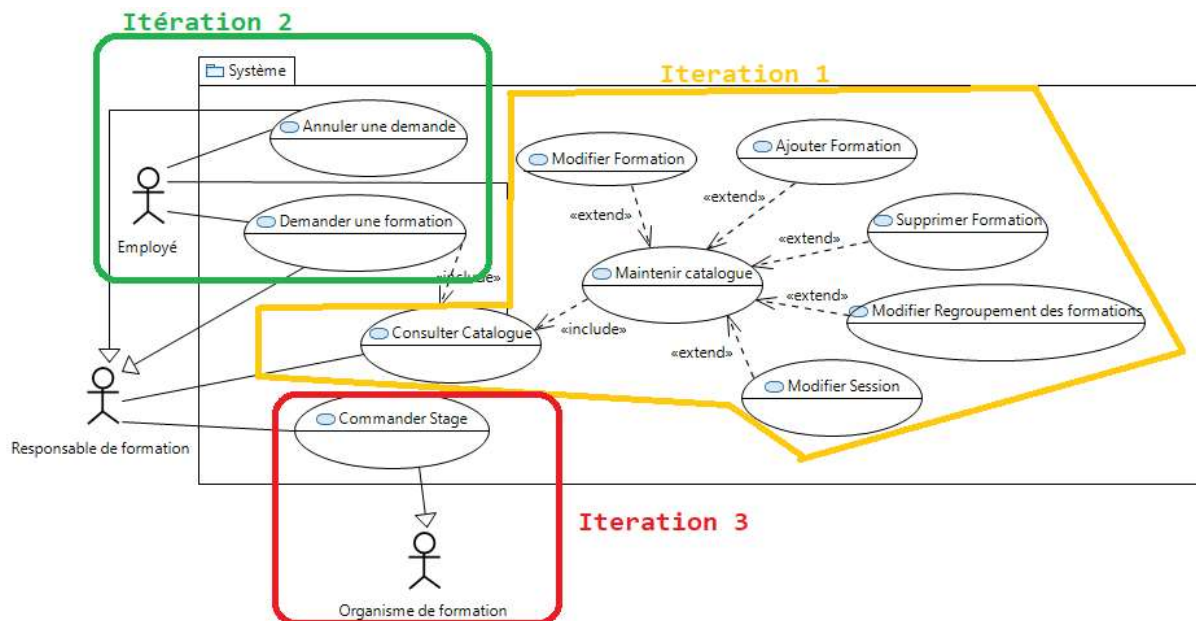
Proposer une découpe du projet, guidée par les cas d'utilisation, en plusieurs itérations à partir du travail d'analyse précédent.

Travail à faire : Découper le projet en 3 itérations.

Ordre de priorité conseillé :

- ✓ Cas d'utilisation concernant les packages métier duquel peuvent dépendre les autres packages.
- ✓ Cas d'utilisation principaux du système.
- ✓ Les autres cas d'utilisation

Solution :



9- Etape 6 : Définition de l'architecture système

- ✓ Le principal objectif de l'architecture à trois niveaux (3-tiers) est d'isoler l'application (la logique métier), du client (Navigateur/ IHM) et de la base de données. Le souci premier est donc de répondre au critère d'évolutivité : **pouvoir modifier une couche sans avoir à retoucher une autre couche. C'est le principe de couplage faible.**
- ✓ L'application est divisée en 3 couches applicatives : couche présentation, couche métier et couche DAO (Data Access Object).
- ✓ La couche présentation contient les classes dites « **contrôleurs** » qui gèrent l'interfaçage entre les IHM et la couche métier.
- ✓ Les « **contrôleurs** » connaissent alors l'interface des objets de la couche métier et joue le rôle de Façade vis-à-vis des IHM.
- ✓ La couche métier dite également couche Service est facultative. Cette couche est obligatoire lorsque l'application doit gérer les règles de gestion métier.
- ✓ La couche DAO (Data Access Object) gère l'accès aux base de données, la génération de rapports, etc..
- ✓ Les « **contrôleurs** » doivent connaître uniquement l'interface de la couche métier. Ces contrôleurs ne doivent en aucun cas connaître les implémentations de la couche métier. Ici, il faut absolument implémenter le Design Pattern IOC (Inversion Of Control).

- ✓ Par rapport aux entités (les classes persistantes ou bien les Business Object), ces dernières ne doivent pas être utilisées dans la couche présentation. L'objectif, si on modifie une table (classe BO) on ne doit pas impacter les pages HTML, XHTML, etc. Ici, Il s'agit du Design Pattern dit Value Object.

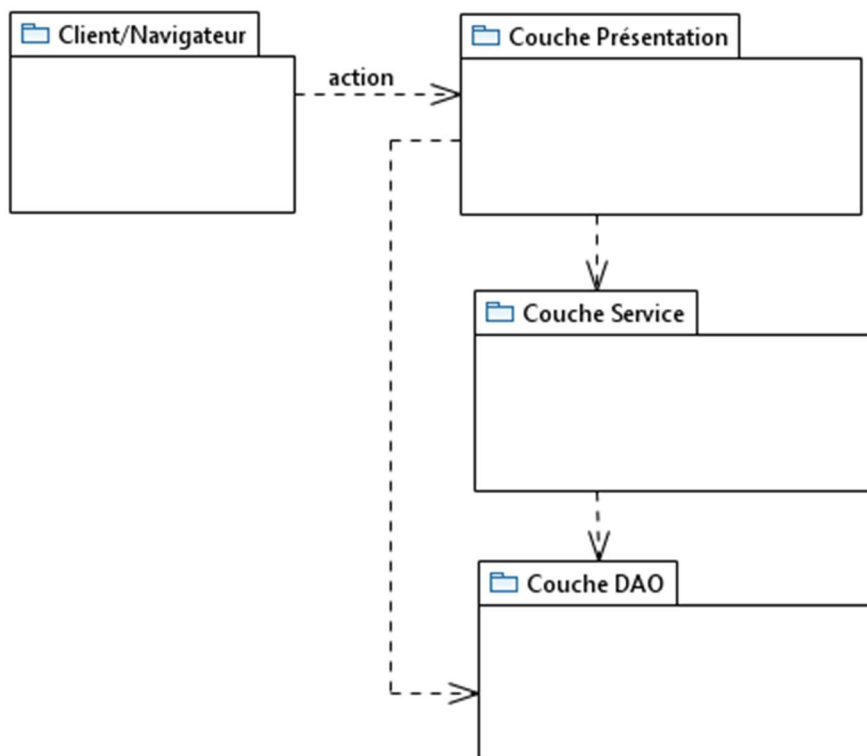
a. Architecture en couches préliminaire

Travail demandé

Proposer un diagramme d'architecture préliminaire du projet en tenant compte :

- ✓ Des recommandations précédentes
- ✓ Du fait que la couche présentation (les contrôleurs) devra être détaillée lors de la conception.
- ✓ Du fait que la couche métier regroupe les packages identifiés lors de l'étape d'analyse statique du domaine.
- ✓ La couche DAO gère l'accès aux bases de données
- ✓ Vous pouvez ajouter une couche pour la conversion des objets VO vers les objets BO (en général cette couche est appelée Domaine).

Solution :



10- Etape 7 : Définition des opération système (itération 1)

- L'itération 1 correspond aux cas d'utilisation « consulter catalogue » et « Maintenir catalogue ».
- Une opération système est une unité de traitement qui est réalisée par le système dans le cadre d'un cas d'utilisation et qui peut être réutilisée dans un autre.
- Les opérations système pour le cas d'utilisation « Maintenir catalogue » sont :
 - ✓ CreerFormation(), ModifierFormation() ;
 - ✓ CreerOrgaFormation(), ModifierOrgaFormation() ;
 - ✓ CreerTheme(), ModifierTheme() ;
 - ✓ CreerSession(), ModifierSession() ;
 - ✓ En considérant que l'action de modification inclut toujours la suppression.

Travail demandé : Rédiger le contrat de l'opération système CreerFormation().

Solution :

1. AfficherPageCreerFormation()
2. ConsulterThemes()
3. InitialiserFormation()
4. CreerContenu()
5. CreerSessions()
6. AjouterFormation()

11- Etape 8 : Diagrammes d'interactions (itération 1)

Le contrat d'opération constitue le dernier livrable en matière d'analyse. En effet, si ce dernier décrit ce que fait une opération en termes de changements d'état, il ne doit pas encore décrire comment il y procède.

C'est justement le travail du concepteur de choisir comment les objets logiciels vont interagir pour réaliser telle ou telle opération.

a. Diagrammes d'interaction des opérations système

Travail demandé :

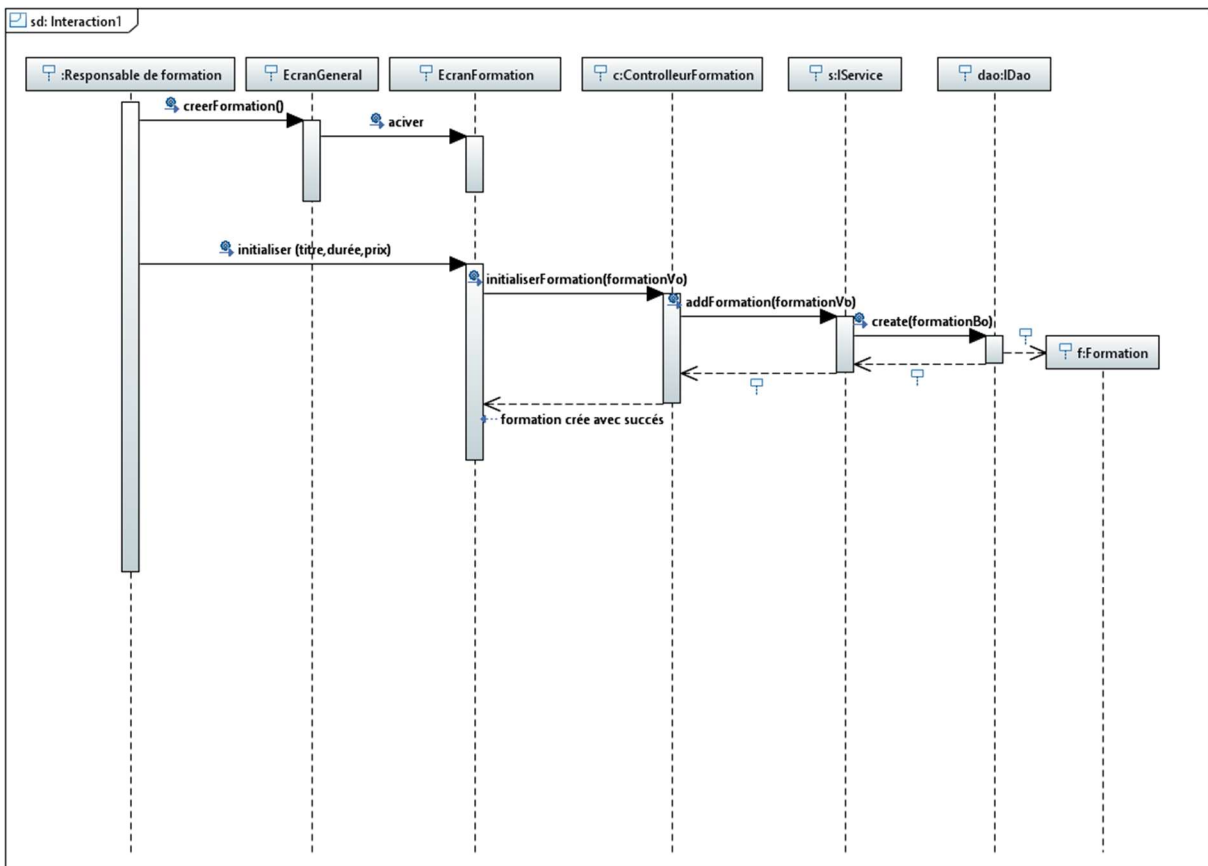
Réaliser un diagramme de séquence ou un diagramme de communication qui montre la réalisation de l'opération système CreerFormation().

Démarche :

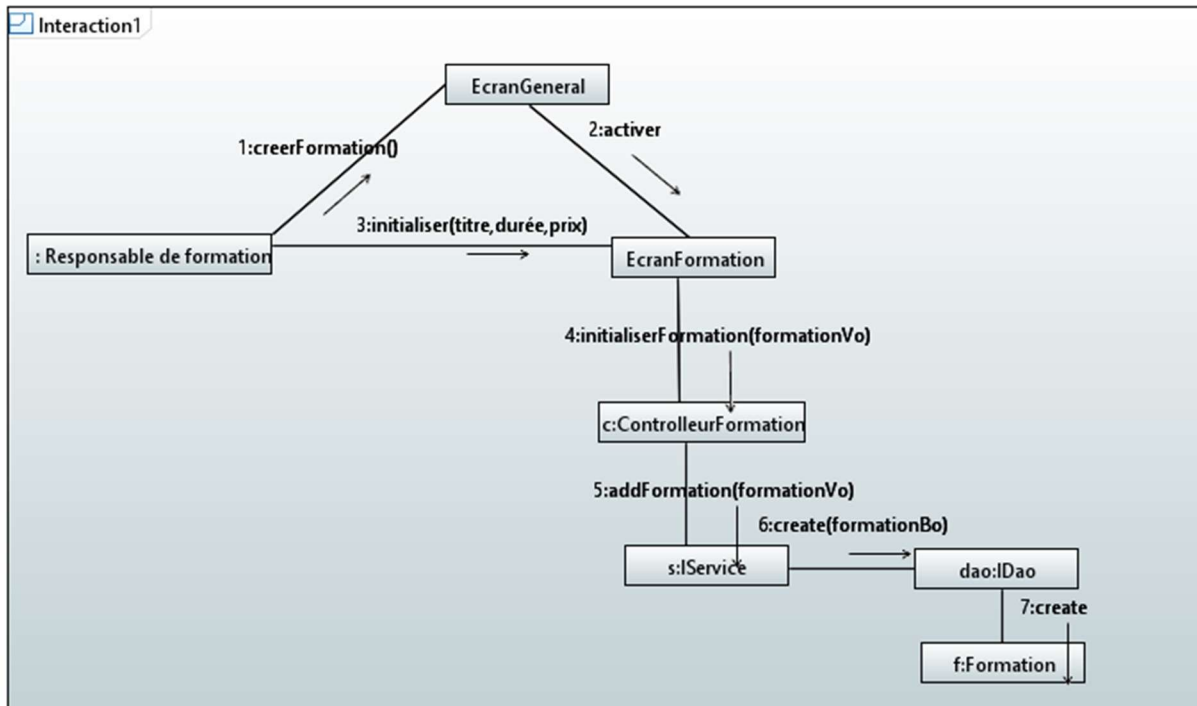
- ✓ Vous pouvez utiliser les stéréotypes (dans Rational Rose, ces stéréotypes sont appelés stéréotypes de **Jacobson**) pour constituer le diagramme de séquence ou de communication.
- ✓ Reprendre les post-conditions répertoriées lors de l'étape précédente.

- ✓ Ne pas oublier que les post-conditions ne représentent que le nouvel état du système à la fin de l'exécution de l'opération système.
- ✓ Privilégiez la représentation basé diagramme de communication, puisqu'au fur et à mesure de la conception, le diagramme de séquence devient de moins en moins lisible.
- ✓ Utiliser la notation multi-objet.

Solution :



b. Diagrammes d'interaction des opérations système (diagramme de communication)



c. Diagramme de séquence de l'initialisation de la formation et création de son contenu

- ❖ Schématiser le cas où on peut accéder à l'écran de création du contenu à partir de l'écran de création de formation ?

12- Etape 9 : Diagramme de classes de conception

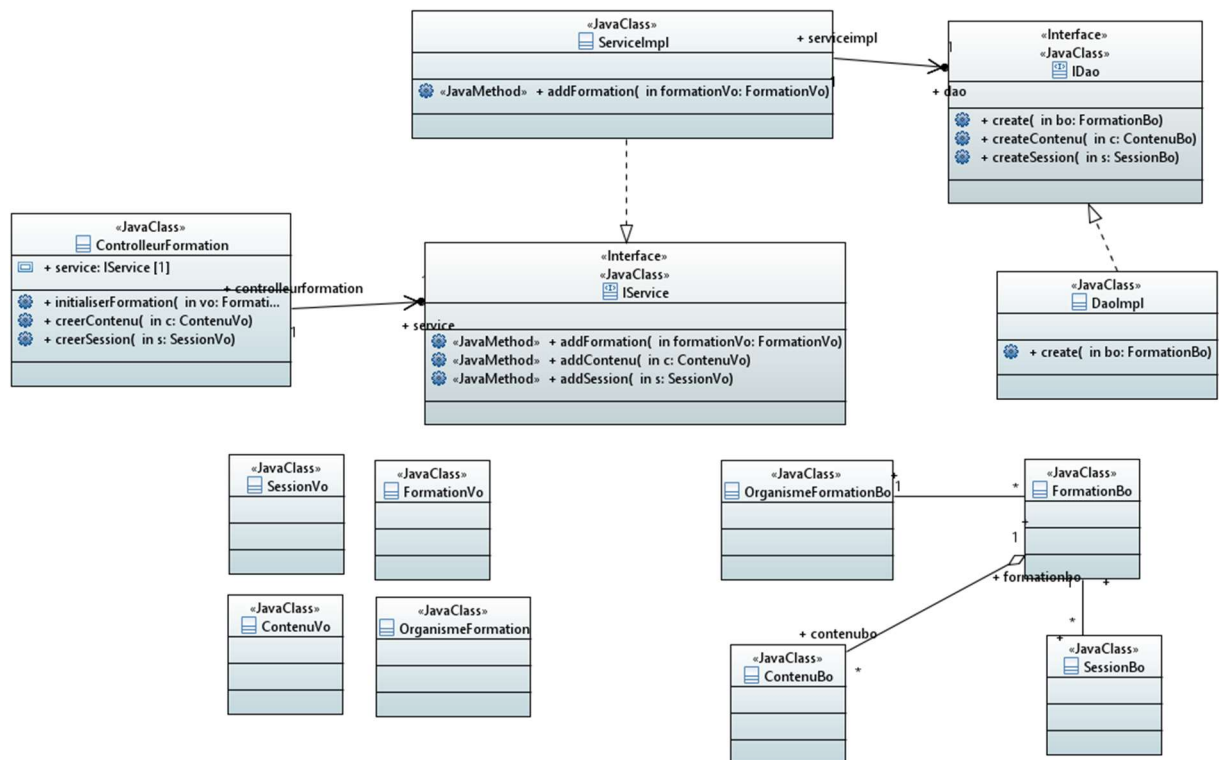
Chaque opération système va donner lieu maintenant à une étude dynamique sous forme d'un diagramme de communication, comme cela a été le cas pour l'opération `CreerFormation()`.

Ces diagrammes vont permettre d'élaborer des diagrammes de classes de conception, et ce en ajoutant principalement les informations suivantes aux classes issues du modèles d'analyse :

- ✓ Les opérations : un message ne peut être reçu par un objet que si sa classe a déclaré l'opération publique correspondante.
- ✓ La navigabilité des associations ou des dépendances entre classes, suivant que les liens entre objets sont durables ou temporaires, et en fonction de la circulation des messages.

Faire le diagramme de classes de conception pour `CreerFormation()`

Solution :



13- Etape 10 : Retour sur l'architecture

Reprendre l'architecture en couche du système, et monter toutes les classes identifiées à l'intérieur des packages correspondants.

14- Etape 11 : Passage au code objet

Les modèle de conception réalisés permettent de produire, d'une manière aisée, du code objet tel que java ou C # :

- ✓ Les diagrammes de lasses permettent de décrire le squelette du code (les déclarations)
 - ✓ La classe UML devient une classe Java (idem en C#) ;
 - ✓ Les attributs et les méthodes UML, deviennent des attributs et des méthodes Java (Idem en C#) ;
 - ✓ Les rôles navigables produisent des variables d'instances ;
 - ✓ Avec les diagrammes de communication (collaboration), il est facile d'écrire le corps des méthodes, en particulier la séquence d'appels de méthodes sur les objets qui communiquent.
- ❖ Ecrire le code java correspondant à la classe DemandeFormation
 - ❖ Ecrire le code java correspondant à la classe ControleFormation
 - ❖ Utiliser Papyrus pour générer le code source.