

## Support de cours en système d'informations, méthodes avancées : Chapitre 3 : Capture des besoins fonctionnels

### 1. Introduction

Par besoins fonctionnels on désigne l'ensemble des besoins liés au métier et domaine traités par le système, par opposition aux besoins techniques, que nous verrons dans le chapitre suivant. Aussi, il faut se détacher à cette étape des besoins en interaction homme-machine (IHM) qu'il est toutefois possible d'intégrer dans d'autres étapes. Les étapes de capture des besoins fonctionnels sont décrites dans ce qui suit.

### 2. Identification des cas d'utilisation

Il s'agit dans cette étape d'identifier l'ensemble des cas d'utilisation à partir du diagramme de contexte dynamique de l'étape d'analyse préliminaire. Un cas d'utilisation regroupe un ensemble cohérent de messages (peut être réduit à un seul message) émis vers et/ou reçu du système d'information (voir le chapitre précédent sur l'identification des messages).

L'erreur à éviter est de descendre trop bas dans l'identification des cas, c'est-à-dire de définir un cas d'utilisation comme une seule transaction (dans le sens de manipulation informatique : ajout, modification, suppression). Il doit plutôt décrire une intention de l'acteur vis-à-vis du système en termes de changement d'état global et de bénéfice métier. Le système et / ou l'acteur avant et après l'exécution doit changer d'état ou bénéficier de l'exécution du cas.

**Exemple :** gestion de la bibliothèque. Les cas d'utilisation

- Gestion des adhérents (personnes);
- Gestion des adhésions (abonnement dans la bibliothèque);
- Gestion des prêts;
- Gestion des livres (indépendamment des prêts);
- Elaboration de statistiques de suivi des prêts (nombre de prêts, ...);
- Elaboration des statistiques de suivi de la bibliothèque (nombres d'adhérents, de prêts, ...);
- Gestion de consultations internes des livres.

### 3. Identification des acteurs principaux et les acteurs secondaires

Un acteur principal est obligatoire pour un cas d'utilisation (au moins un). Il représente l'acteur concerné par l'intention fonctionnelle. Les acteurs secondaires sont des acteurs qui ne sont pas concernés directement par le cas, mais qui peuvent être sollicités pour la réalisation du cas.

**Exemple :** l'agent administratif peut consulter le chef de service en cas de problème de traitement des informations personnelles d'un employé.

**Exemple pratique :** nous reprenons l'exemple du projet de gestion de *Kitabi*. Le tableau suivant liste les cas d'utilisations ainsi que les acteurs et messages émis et reçus.

Cas d'utilisation	Acteurs	Envoie / réception d'information
Gestion des adhérents	Agent de prêts	<i>Envoie</i> : informations détaillées sur les adhérents <i>Reçoit</i> : Liste des adhérents
Gestion des adhésions	Agent de prêts	<i>Envoie</i> : Les informations d'adhésion (ex. date) et sur les renouvellements <i>Reçoit</i> : la liste des adhésions (surtout pour renouvellement)
Gestion des prêts	Agent de prêts	<i>Envoie</i> : informations sur les prêts et restitutions <i>Reçoit</i> : disponibilité des livres, adhérents sanctionnés
	Adhérent	<i>Envoie</i> : réservation de livres <i>Reçoit</i> : disponibilité des livres, livres réservés
	Consultant	<i>Reçoit</i> : disponibilité des livres
Gestion des livres	Bibliothécaire	<i>Envoie</i> : information détaillée sur les livres <i>Reçoit</i> : liste des livres
Elaboration de statistiques de suivi des prêt	Bibliothécaire	<i>Reçoit</i> : statistiques de suivi des prêts
Elaboration des statistiques de suivi de la bibliothèque	Directeur	<i>Reçoit</i> : statistiques de suivi de la bibliothèque
Gestion de consultations	Agent de prêt	<i>Envoie</i> : informations sur la consultation (livres, consultant) <i>Reçoit</i> : livres en cours de consultations (en particulier les coordonnées du consultant)

*Exemple de liste de cas d'utilisation*

## 4. Description des cas d'utilisation

Durant cette étape, chaque cas d'utilisation sera décrit par l'intention (but) suivi de l'acteur dans l'exécution du cas et les actions élémentaires qu'il peut effectuer. La liste des actions peut ne pas être exhaustive.

**Exemple :** Description textuelle du cas d'utilisation *Gestion des Adhérents*

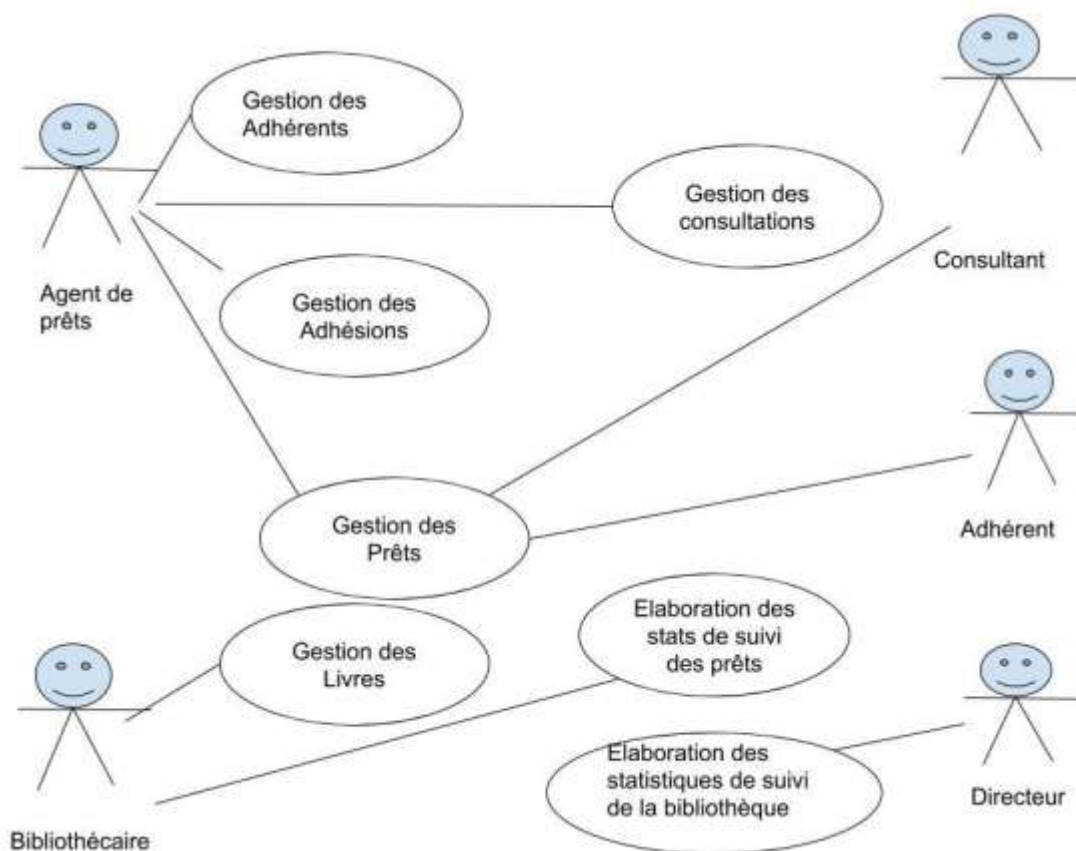
<u>Gestion des adhérents</u>
<p><b><u>But</u></b> : maintenir les informations sur un adhérent à jour</p> <p><b><u>Actions</u></b> : ajouter, supprimer, mettre à jour les informations personnelles d'un adhérent indépendamment des adhésions.</p>

*Exemple de fiche descriptive d'un cas d'utilisation*

## 5. Elaboration du diagramme des cas d'utilisations

Dans cette étape, on va élaborer le diagramme des cas d'utilisation. On représente les cas par une ellipse et chaque acteur par le stickman.

**Exemple :** Diagramme des cas d'utilisation de la gestion de la bibliothèque



*Diagramme des cas d'utilisation du cas Kitabi*

## 6. Structuration des cas d'utilisation

Chaque cas d'utilisation peut faire l'objet d'une structuration avec une fiche descriptive. Le contenu de la fiche est laissé à l'analyste. Un exemple de fiche descriptive est le suivant.

Sommaire d'identification
---------------------------

**Titre du cas d'utilisation :** *le nom du cas d'utilisation*

**But :** *l'intention ou valeur ajoutée attendue de l'exécution du cas* **Résumé :** *un bref résumé du cas*

**Résumé :** *brève description du cas*

**Acteurs :** *l'acteur principal et les éventuels acteurs secondaires*

**Date de création :** *date de création du cas* **Date de mise à jour :** *date de la dernière MAJ*

**Version:** *version actuelle* **Responsable :** *créateur du cas ou responsable*

### Description des enchaînements

**Préconditions :** *ce que doit être réalisé avant l'exécution du cas*

**Enchaînements :**

- Début du cas : *quand est-ce que commence le cas ?*
- enchaînements 1 : nom de l'enchaînement
  - actions à effectuer
  - exceptions possibles pour les actions enchaînement 2
- Fin du cas : *quand est-ce que se termine le cas ?*

**Exceptions :**

- Exception 1 : *cas exceptionnel dans la réalisation d'une action*
- Exception 2
- ...

**Postconditions :** *ce que doit être réalisé à la fin de l'exécution du cas*

**Exemple :** Cas Kitabi

### Sommaire d'identification

**Titre du cas d'utilisation :** gestion des adhérents

**But :** maintenir la fiche des adhérents à jour

**Résumé :** ajouter un adhérent, supprimer un employé, modifier les informations sur un adhérent

**Acteurs :** agent de prêt

**Date de création :** 17/11/2011 **Date de mise à jour :** 17/11/2011

**Version:** 1.0 **Responsable :** B. Ahmed

**Préconditions :** Agent de prêt authentifié **Enchaînements :** le cas commence lorsque l'agent de prêt dispose des informations sur un adhérent ou lorsqu'un adhérent ne renouvelle pas pour une longue période.

**Enchaînement 1 :** Ajouter un nouvel adhérent -l'agent saisit les informations sur l'adhérent  
[exception 1 : l'adhérent avec les mêmes informations existe]

**Enchaînement 2 :** Modifier les informations sur un adhérent

- l'agent cherche un adhérent puis modifie les informations Enchaînement 3 : supprimer un adhérent
- l'agent cherche l'adhérent à supprimer puis supprime toutes les informations s'y relatant [exception 2 : adhésion en cours ou délai d'archivage non atteint]

ce cas se termine lorsque l'adhérent est ajouté, ses informations mises à jour ou si la suppression est réussie.

**Traitement des exceptions :**

[exception 1 : empêcher l'ajout]

[exception 2 : empêcher la suppression] Postcondition : adhérent ajouté, ses infos à jour ou il est supprimé.

## 7. Compléter la description des cas d'utilisation

Bien que la description textuelle des cas d'utilisation soit indispensable, il est toujours utile de la compléter par une description graphique sous la forme de diagrammes UML. Les différents diagrammes qu'on peut utiliser à ce niveau sont les suivants :

- **Diagramme d'activités :** c'est le diagramme le plus recommandé. Il permet de montrer les enchaînements d'un cas et aussi les enchaînements parallèles.
- **Diagramme d'états-transition :** il est plus complexe à comprendre.

- **Diagramme de séquences** : il s'adapte à des scénarios particuliers. Comme le diagramme d'activités, ce diagramme permet de mieux illustrer les enchaînements.
- **Diagramme de collaboration** : son pouvoir d'expression par rapport aux cas n'est pas aussi élevé que celui des diagrammes de séquence et d'activité.

#### *Organisation des cas d'utilisation*

A cette étape, on identifie les éventuelles relations entre les cas d'utilisation (inclusion, extension, généralisation/spécialisation).

### **7.1. Identification des relations d'inclusion**

Les relations d'inclusion sont identifiées par factorisation des traitements communs à plusieurs cas. Un exemple de cela est l'authentification requise pour chaque acteur avant le début de toute utilisation du système.

### **7.2. Identification des relations d'extension**

Les cas d'utilisation définis comme extensions à d'autre cas regroupent des traitements optionnels ou répondant à des conditions spécifiques. Un exemple de cela est l'extension de l'ajout d'une commande par l'ajout de produits.

### **7.3. Identification des relations de généralisation/spécialisation**

Ce type de relation est identifié lors de l'existence de traitements spécifiques ou modifiés d'un cas ou de plusieurs cas par rapport à un traitement normal.

## **8. Identification des classes candidates**

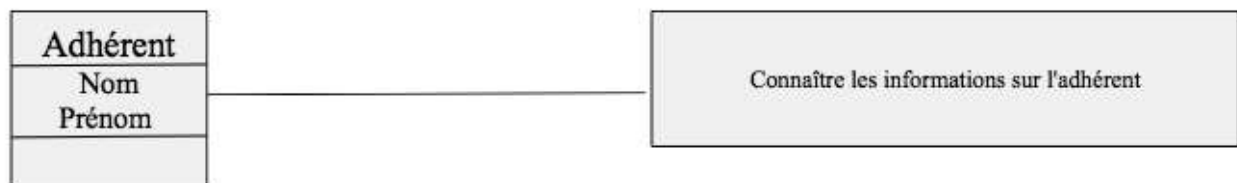
Cette étape consiste à identifier la liste préliminaire des classes qui permettent de répondre aux exigences statiques (attributs) et dynamiques (opérations) de chaque cas d'utilisation. L'identification des classes est intuitive mais guidée par le contenu et description des cas d'utilisation. A ce niveau là, il n'est pas obligatoire de définir les attributs et opérations de chaque classe. Cependant, on peut décrire chacune des classes par une note contenant la responsabilité qu'elle joue.

Une responsabilité d'une classe est sa raison d'être. On l'exprime par le rôle joué par la classe avec un niveau de détail élevé par rapport aux attributs et opérations.

**Exemple de diagramme de classes candidates** : On prend le cas « gestion des adhésions »



### Responsabilité des classes



Une responsabilité peut contenir dans sa description les attributs de la classe, les opérations, mais aussi les associations. Cependant, lorsque deux responsabilités de deux classes donnent lieu à deux associations identiques, il faut représenter une seule association entre les deux classes.

Le diagramme de classes issu de chaque cas d'utilisation est appelé « diagramme de classes participantes ». A ce niveau là, on ne doit pas chercher l'ensemble exhaustif des classes, car d'autres classes peuvent réapparaître plus tard.

## 9. Validation et consolidation

Au terme de la capture des besoins fonctionnels, il convient, avant de passer à l'étape parallèle (capture des besoins techniques) de valider les résultats de cette étape avec les utilisateurs.

La validation peut être guidée par un tableau comme suit

	Besoin 1	Besoin 2	...	Besoin <i>m</i>
Cas d'ut. 1	X			X
...		X		X
Cas d'ut. <i>n</i>	X			

Naturellement, l'analyse de ce tableau doit permettre de s'assurer que toutes les exigences du

système sont prises en charge avec le cas d'utilisation. Faute de quoi, il faudra revenir sur les étapes précédentes autant de fois que nécessaire.

## **10. Conclusion**

Ce chapitre a été consacré à la première étape réelle du processus 2TUP qui est l'étude préliminaire. Cette étape permet de traduire les besoins fonctionnels recensés dans l'étude préliminaire sous la forme de cas d'utilisation. C'est durant cette étape qu'on fait le passage être une expression en langue naturelle de ce que les utilisateurs veulent fonctionnellement et la manière de répondre à ces besoins. L'étape suivante est similaire à la capture des besoins fonctionnelle quoiqu'il soit possible de ne pas l'entamer qu'après avoir terminé les étapes de la branche gauche. Néanmoins, il est préférable de la présenter juste après la capture des besoins fonctionnels.