

Génie Logiciel 2

Chapitre 2 : Etude préliminaire et spécification des besoins

Dr. Sahar SMAALI

sahar.smaali@univ-constantine2.dz

Etudiants concernés

Faculté/Institut	Département	Niveau	Spécialité
NTIC	TLSI	Licence 3	GL

Etude préliminaire

Etude préliminaire

- **But** : éviter de développer un logiciel non adéquat.
- **Résultat**: ensemble de documents décrivant les aspects pertinents de l'environnement du futur système, son rôle et sa future utilisation.
- **Activité essentielle** au début du processus de développement. Elle est menée avec les études de faisabilité et la planification.

- Analyse du domaine
- Définition de la problématique et du cadre du projet
- Recensement des besoins fonctionnels et non-fonctionnels

Etude préliminaire

1. Analyse du domaine

- **Domaines :**

- Domaines vastes → Réservations de compagnies aériennes, diagnostic médical et analyse financière.
- Domaines étroits → Fabrication de peinture ou réunions de planification

- **Réunir toute l'information du domaine disponible pour**

- Comprendre le problème
- Prendre les bonnes décisions lors de l'analyse des besoins et des autres étapes du processus de génie logiciel.
- Diffuser aux différents membres de l' équipe

Etude préliminaire

1. Analyse du domaine

- **Document de l'analyse du domaine**

- **Glossaire**

- Signification de tous les termes utilisés dans le domaine qui ne font pas partie du langage courant ou qui ont une signification particulière.

- **Connaissances générales du domaine**

- Principes scientifiques, les techniques d'analyse et le fonctionnement de toute technologie.

- **Etude de l'existant concurrents**

- Systèmes disponibles pour aider les utilisateurs et les clients, y compris ceux déjà utilisés.

Etude préliminaire

1. Analyse du domaine

- **Avantages**

- Communication plus efficace avec les clients
- Meilleure compréhension des problématiques des clients
- Assurer que les solutions adoptées résolvent les problèmes du clients.

Etude préliminaire

1. Analyse du domaine- Exemple

● **Système de gestions des dossiers médicaux des patients**

● **Glossaire :**

- Expliquer les termes et abréviations utilisés dans les dossiers de l'hôpital par les médecins et les infirmières

● **Connaissance générale du domaine**

- fournir l'organigramme de l'hôpital ainsi que des statistiques sur les patients, les soins et les services offerts ainsi que le staff médical et administratif

● **Systèmes concurrents**

- Analyser le système ancien de l'hôpital et étudier la possibilité de l'améliorer. Analyser les systèmes existants sur le marché

- **Problématique**

- Difficulté à laquelle les utilisateurs ou les clients sont confrontés
- Opportunité qui se traduira par un avantage, tel qu'une productivité ou des ventes améliorées.

- **Comprendre le problème avant de commencer à créer la spécification des besoins**

- Ne pas résoudre le mauvais problème

- **Problématique courte et succincte**

- une ou deux phrases

2. Définition de la problématique et du cadre du projet

- **Exemple**

- **Problématique**

- Comment automatiser et faciliter la gestion des dossiers de l'hôpital?

- On ne s'intéresse dans ce cours uniquement à la gestion des dossier médicaux des patients hospitalisés.

- **Objectifs du projet :**

- Éviter toute perte d'informations des patients
- Gagner du temps
- Faciliter la gestion des chambres des services

Etude préliminaire

3. Recensement des besoins fonctionnels et non-fonctionnels

Définition

Un besoin est une **caractéristique** que le système doit avoir ou une **contrainte** qu'il doit satisfaire pour être **accepté par le client ou les utilisateurs**.

Besoins Fonctionnels

- ☐ Que doit faire le système?
- ☐ Le system du point de vue selon son future utilisateur?
- ☐ **Quoi?**

Besoins non fonctionnels

- ☐ Contraintes
- ☐ Exigences techniques ou choix
- ☐ **Comment?**

3. Recensement des besoins fonctionnels et non-fonctionnels

● **Les sources servant au recensement des besoins :**

- Utilisateurs directs
- Personnes ayant une relation avec le système
- Autres systèmes avec lequel va interagir le système
- Matériel sur lequel sera déployé le système
- Contraintes juridiques et administratives
- Contraintes techniques

3. Recensement des besoins fonctionnels et non-fonctionnels

- **Les sources servant au recensement des besoins :**
 - Utilisateurs directs
 - Personnes ayant une relation avec le système
 - Autres systèmes avec lequel va interagir le système
 - Matériel sur lequel sera déployé le système
 - Contraintes juridiques et administratives
 - Contraintes techniques
- **Plusieurs moyens utilisés pour recenser les besoins :**
 - Analyse des systèmes et documents actuels
 - Interviews
 - Questionnaires
 - Ateliers

Etude préliminaire

Exemple- Système de gestion des dossiers des patients

- **Besoins fonctionnels**

- Admission d'un patient à un service
- Création des dossiers des patients
- Enregistrement des comptes rendu des patients

- **Besoins non-fonctionnels**

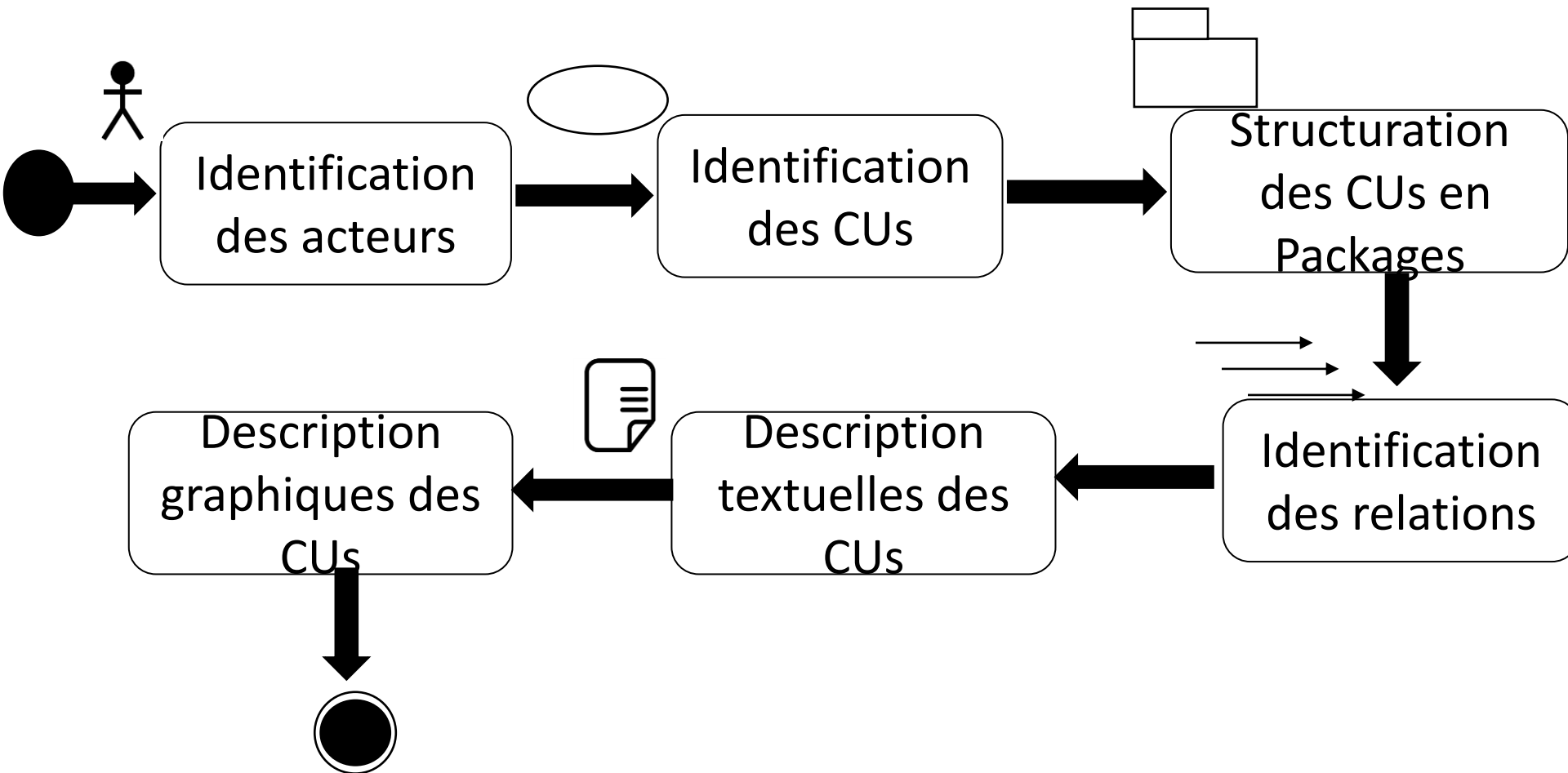
- Authentifications des secrétaires et des infirmiers
- Création des comptes utilisateurs
- Le dossier médical du patient doit contenir aussi les hospitalisations effectuées dans d'autres hôpitaux.

Spécification des besoins (Modèle de cas d'utilisation)

Modèle de cas d'utilisation

- Un système est destiné à être utilisé par des utilisateurs (humains ou machines)
- Le point de vue des utilisateurs est important pour comprendre :
 - Le fonctionnement du système
 - Les services rendus par le système
- DCU Permet de décrire d'un point de vue utilisateur, le système étudié
- Il n'y a pas une démarche spécifique pour la réalisation du DCU car une telle démarche n'a pas été standardisée

• Démarche de construction du modèle des cas d'utilisations



Modèle de cas d'utilisation

1. Identification des acteurs

- **Acteur** : représente un rôle joué par des entités externes (utilisateurs humains, dispositifs matériels ou autres systèmes) qui interagissent directement avec le système étudié.
- Il ne faut pas confondre la notion d'acteur avec la notion d'utilisateur.
 - **Acteur**: décrit le rôle qu'un utilisateur joue par rapport au système.
 - **Utilisateur**: Une personne utilisant le système
- **2 types d'acteurs** :
 - **Acteur principal** : pour lesquels l'objectif du cas d'utilisation est essentiel .
 - **Acteur secondaire** : qui interagissent avec le cas d'utilisation mais dont l'objectif n'est pas essentiel.

Modèle de cas d'utilisation

1. Identification des acteurs

- **Exemple- Système de gestion des dossiers des patients**

Acteur	Type	Rôle
Secrétaire	Principal	représente le personnel administratif responsable sur la gestion des chambres et de lits dans l'hôpital
Infirmière	Principal	représente le personnel responsable sur la gestion des dossiers médicaux des patients.

Modèle de cas d'utilisation

2. Identification des cas d'utilisation

- **Cas d'utilisation :**

- représente une **fonctionnalité** fournie par le système spécifié par une **séquence d'actions**.
- abstractions du **dialogue** entre les acteurs et le système.
- décrit des interactions potentielles sans entrer dans les détails.

Modèle de cas d'utilisation

2. Identification des cas d'utilisation

● Exemple- Système de gestion des dossiers des patients

Acteur	Cas d'utilisation
Secrétaire	<ul style="list-style-type: none">• Admettre un patient• Créer le dossier médical d'un patient• Mettre à jour le dossier médical d'un patient• S'authentifier
Infirmière	<ul style="list-style-type: none">• Saisir compte rendu d'une consultation d'un patient.• Enregistrer les soins offerts à un patient.• S'authentifier

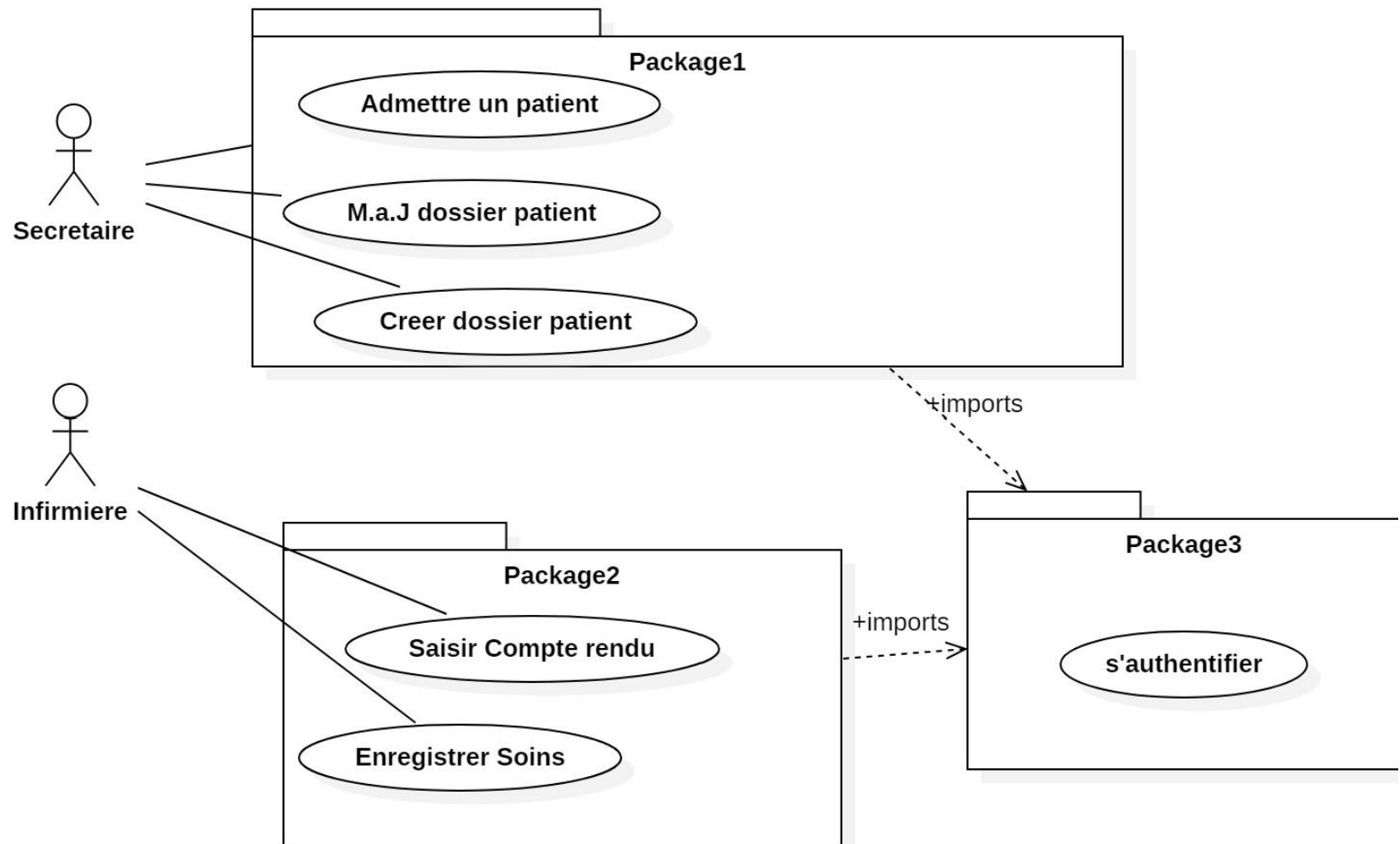
3. Structuration des cas d'utilisation en packages

- Le DCU d'un système peut très vite devenir **encombrant et difficile à décortiquer**.
- Regrouper les CU par **acteur** ou par **domaine fonctionnel**.
- **Le diagramme de paquetage (package) permet de :**
 - d'avoir des ensembles cohérents de CU
 - Découper un problème en sous-problèmes (sous-systèmes)
 - Réutiliser les unités logiques et cohérentes

Modèle de cas d'utilisation

3. Structuration des cas d'utilisations en packages

● Exemple- Système de gestion des dossiers des patients



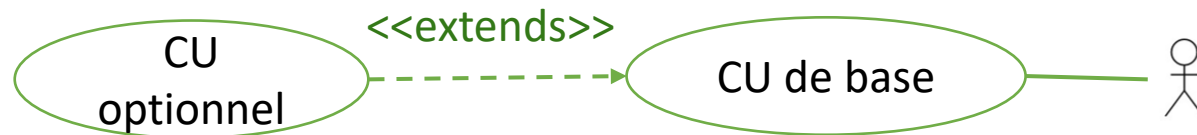
Modèle de cas d'utilisation

3. Identification des relations

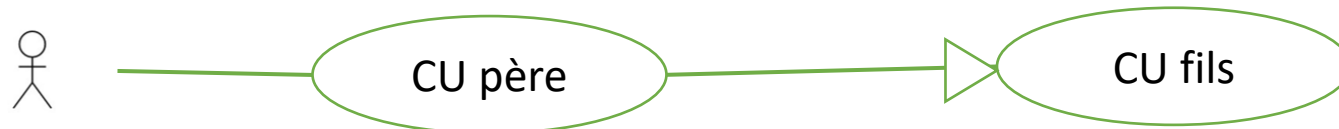
- UML propose trois types de relations entre les cas d'utilisation
- **Inclusion** permet d'inclure un CU dans un autre de façon obligatoire



- **Extension** permet à incorporer un CU dans un autre de façon optionnelle



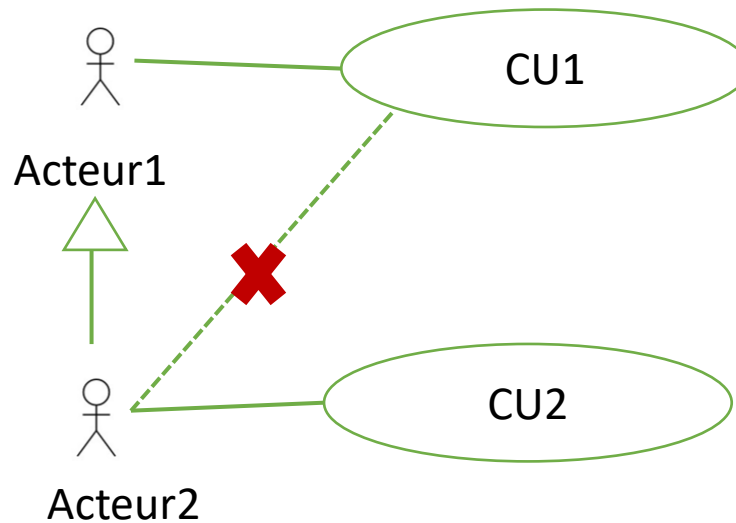
- **Généralisation** permet d'exprimer un comportement commun entre plusieurs cas d'utilisation.



Modèle de cas d'utilisation

3. Identification des relations

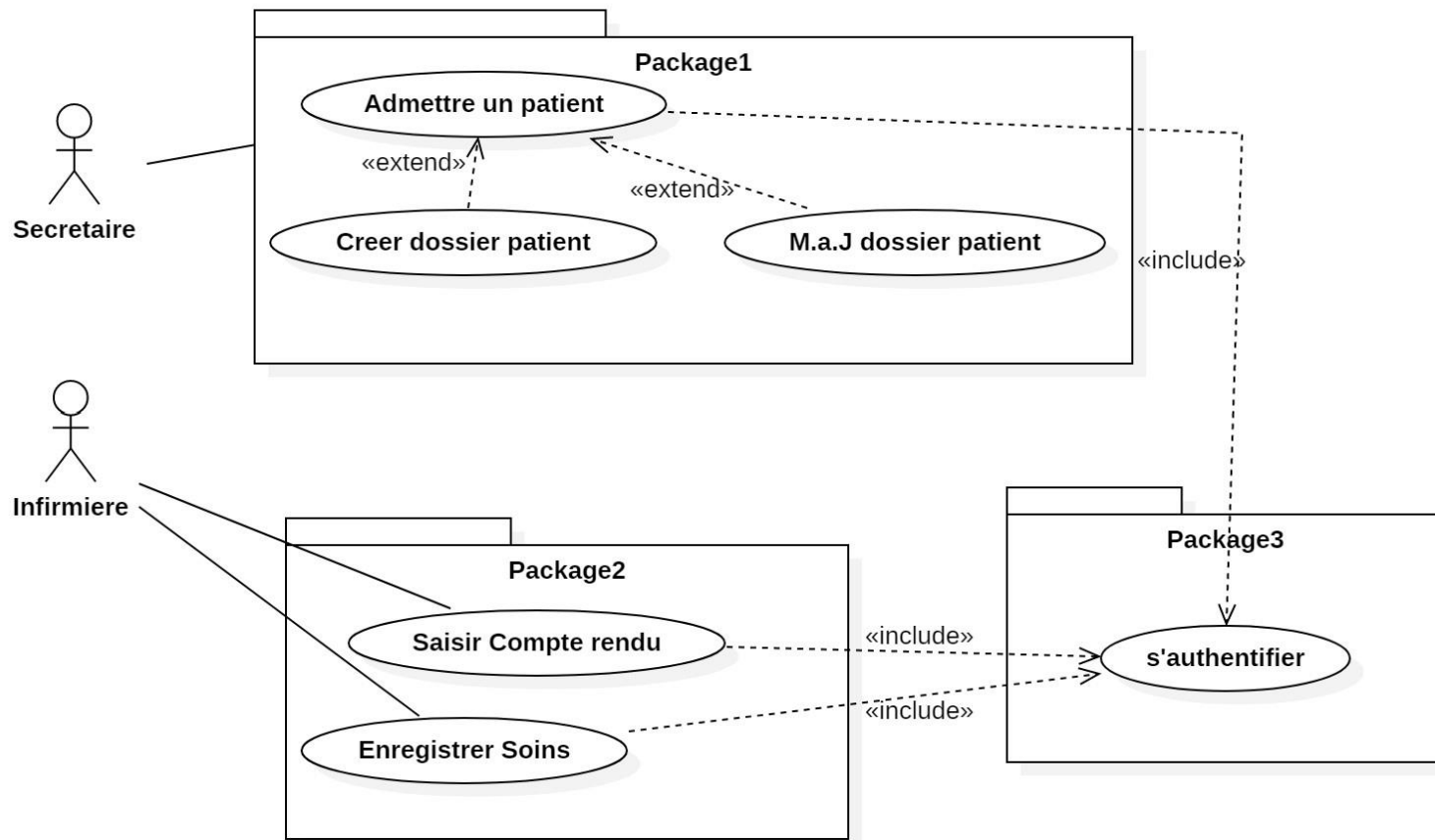
- La seule relation qui peut exister entre les acteurs est la relation de généralisation.



Modèle de cas d'utilisation

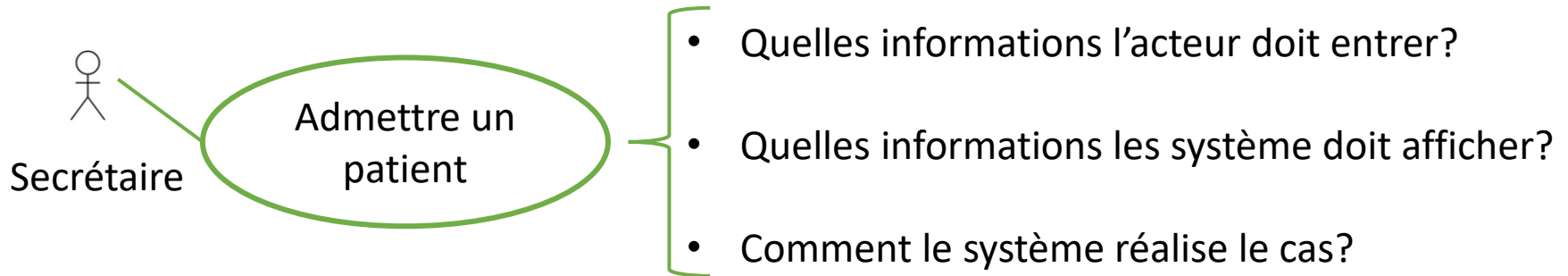
3. Identification des relations

● Exemple- Système de gestion des dossiers des patients



Modèle de cas d'utilisation

4. Descriptions textuelles des Cas d'utilisation

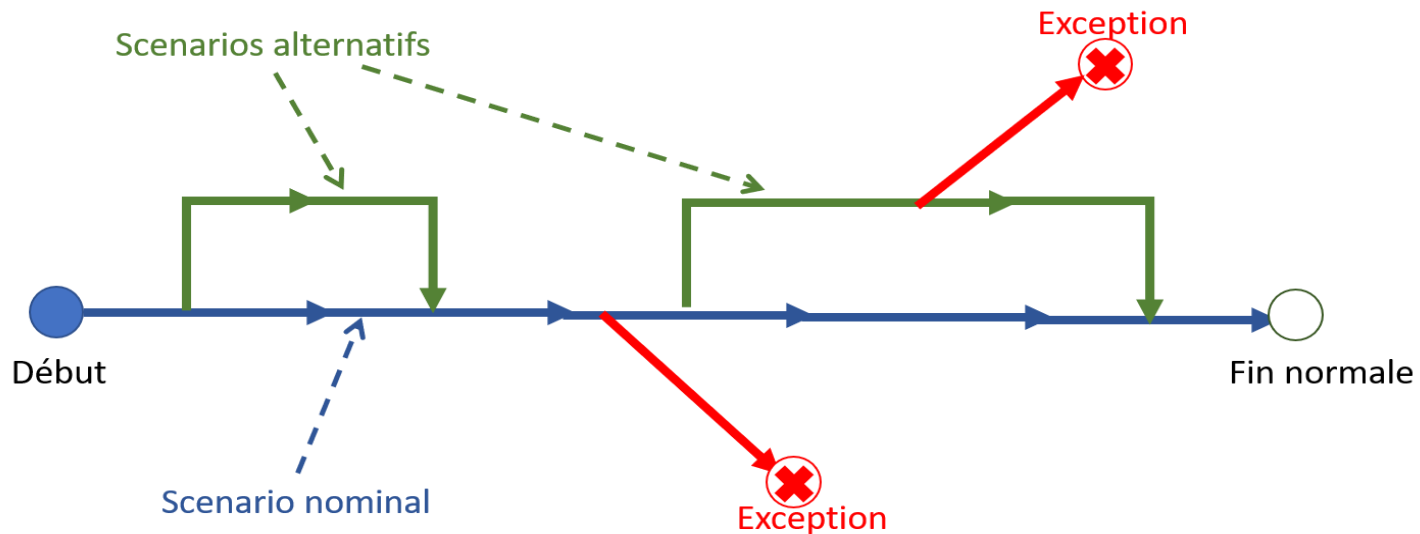


- Une **description textuelle** du CU est nécessaire pour **comprendre** la chronologie **d'actions exécutées** par les acteurs et le système pour réaliser un cas donné.
- La description textuelle est une forme **informelle** pour la compréhension d'un CU.
- C'est une sorte de **documentation** des CU qui permet de **communiquer** facilement et précisément avec les **utilisateurs**.

Modèle de cas d'utilisation

4. Descriptions textuelles des Cas d'utilisation

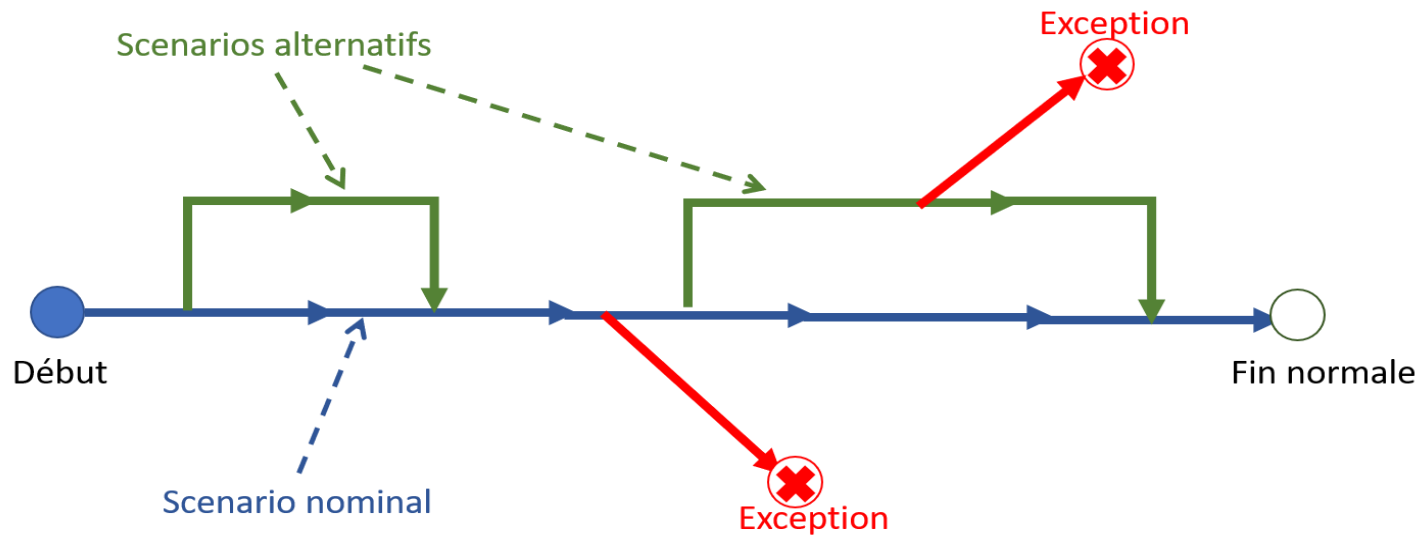
- Un **scénario** est une suite spécifique d'interactions entre les acteurs et le système.
 - Scénario principal : **nominal**
 - Scénario secondaire: c'est soit un cas **alternatif** , soit un cas **d'exception**



Modèle de cas d'utilisation

4. Descriptions textuelles des Cas d'utilisation

- Chaque scénario est composé d'étapes qui peuvent être 3 sortes:
 - Un message d'un acteur au système
 - Une validation ou un changement d'état du système
 - Un message du système vers un acteur



Modèle de cas d'utilisation

4. Description textuelle d'un CU

- ❑ **Généralement, la description textuelle comprend les éléments suivants:**
 - **Identification du cas** : indique le nom du cas et les acteurs impliqués (principaux et secondaires) dans sa réalisation, et résume son objectif.
 - **Préconditions**: Il s'agit des conditions obligatoires pour le déclenchement du cas.
 - **Postconditions**: modifications apportées au système à la fin du cas.
 - **Scénario nominal** : décrit le déroulement idéal des actions, où tout va pour le mieux sans échec.
 - **Scénarios alternatifs**: sont des variantes du scénario normale.
 - **Scénarios d'exception**: décrivent ce qui se passe lors d'une erreur.

• Exemple de description textuelle d'un CU

Nom du cas	Admettre un patient
Type	Principal
Acteurs	Secrétaire
Objectif	Permet à la secrétaire d'attribuer une chambre à un patient.
Pré-condition	La secrétaire doit être authentifiée
Scénario nominal	<ol style="list-style-type: none">1) La secrétaire clique sur le bouton « Admettre un patient »2) Le système lui affiche un formulaire contenant les renseignements sur l'hospitalisation du patient (dates entrée et sortie, service, etc.).3) La secrétaire remplit le formulaire.4) Le système valide le formulaire.5) Le système affiche une liste des chambres libres dans ce service.6) La secrétaire choisit une chambre.7) Le système lui demande si elle veut attribuer la chambre a un patient existant ou de créer un nouveau patient.8) La secrétaire choisit d'attribuer la chambre un patient existant.9) Le système fait appel au cas "mettre à jour un patient".10) Le système enregistre les informations de l'hospitalisation du patient.

• Exemple de description textuelle d'un CU

Scénarios alternatifs	<p>A1. Le système ne trouve pas le patient. L'enchaînement démarre après le point 1 de la séquence nominale :</p> <ul style="list-style-type: none">2) Le système indique que le patient n'existe pas dans la BD.3) Appel au cas "créer un dossier médical".4) La séquence nominale reprend au point 8. <p>A2. Le système ne valide pas le formulaire. L'enchaînement démarre après le point 5 de la séquence nominale :</p> <ul style="list-style-type: none">6) Le système demande au secrétaire de remplir le formulaire.7) La séquence nominale reprend au point 7.
Scénarios d'exception	<p>E1. Le système ne trouve pas une chambre libre. L'enchaînement démarre après le point 2 de la séquence nominale : Le système annule l'opération.</p>
Post-condition	<p>Une chambre est attribuée au patient.</p>

5. Description graphique des cas d'utilisation

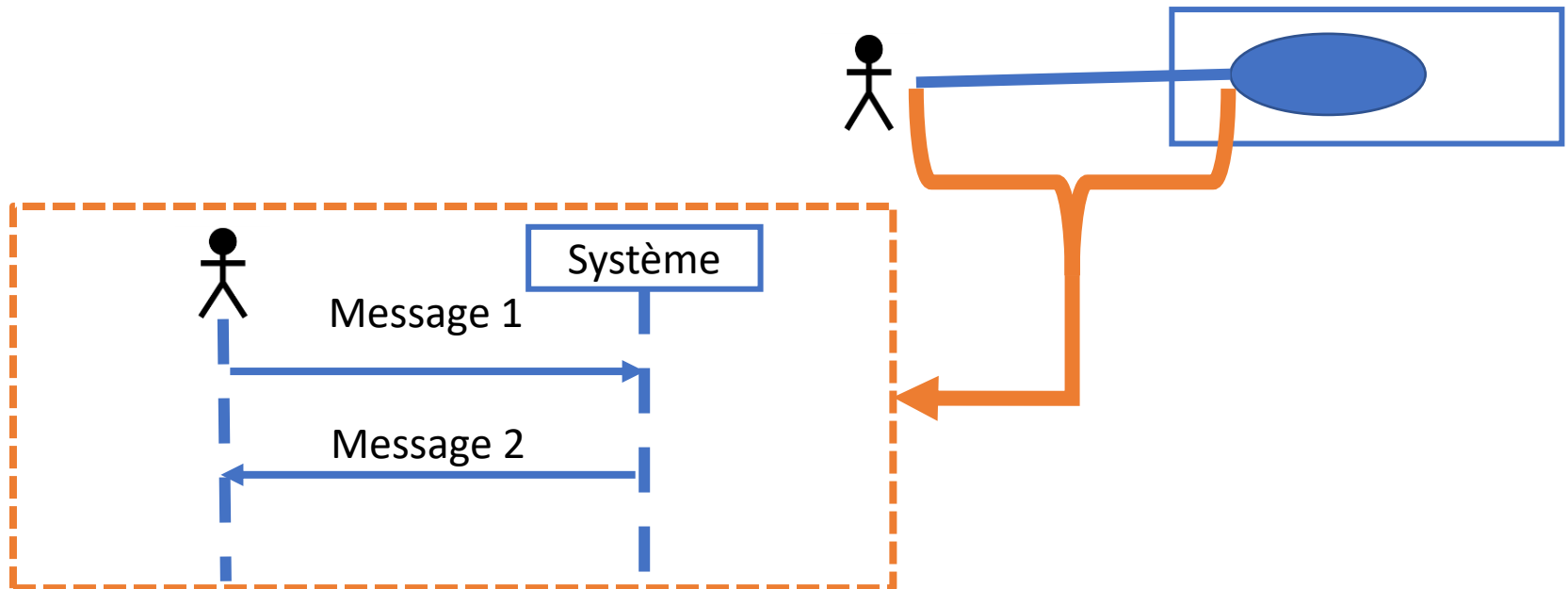
- Compléter la description textuelle par un ou plusieurs **diagrammes dynamiques**, qui apporteront un niveau supérieur de **formalisation**.
 - Diagrammes d'activités
 - *Diagrammes de séquence*
 - Diagrammes de communication

Modèle de cas d'utilisation

5. Descriptions graphiques des cas d'utilisation

● Diagramme de séquence Système

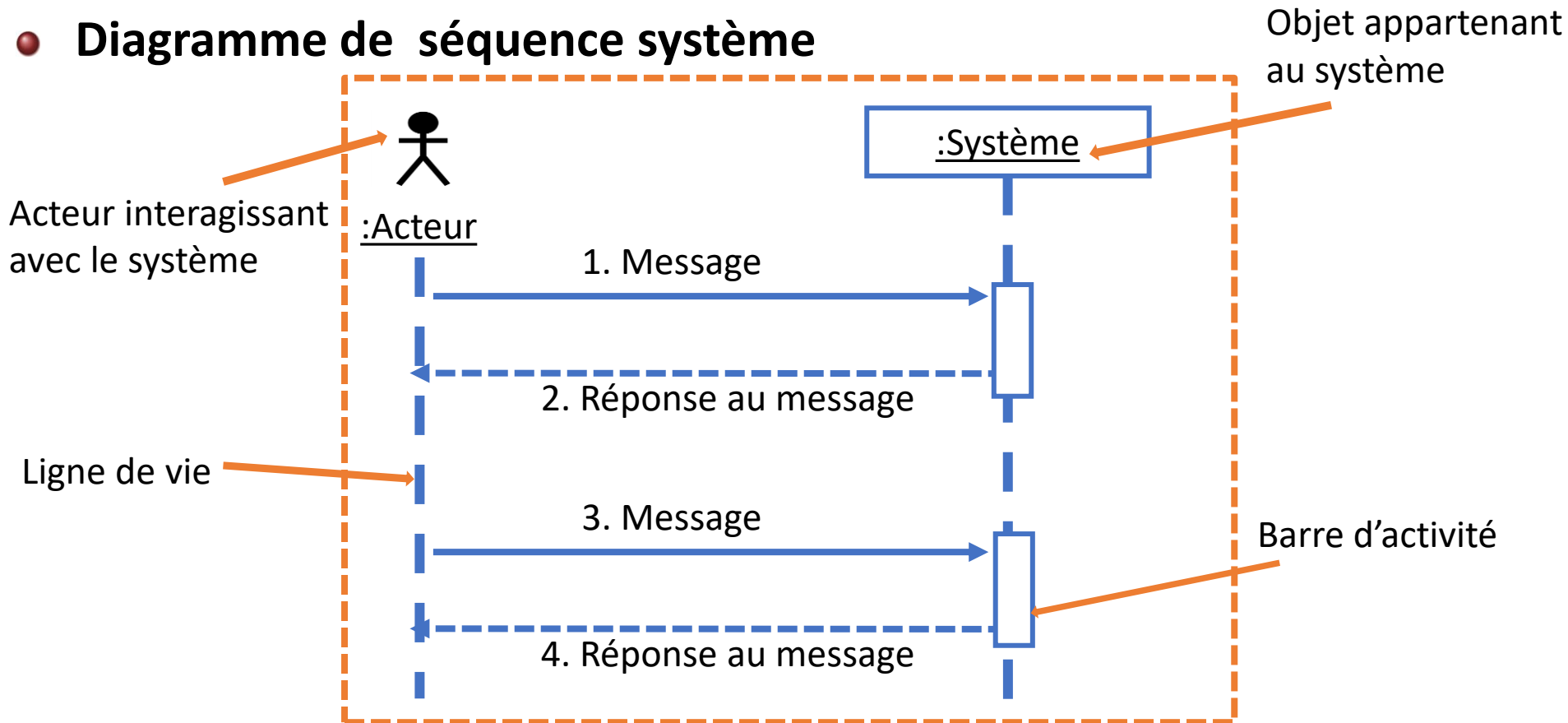
- Décrit le déroulement de la séquence d'actions d'un scénario donné d'un CU.
- Considère que le système est une **boîte noire**.
- Décrit **les messages échangés** avec le **système** (vu comme une seule entité monolithique) selon un ordre **chronologique**.



Modèle de cas d'utilisation

5. Descriptions graphiques des cas d'utilisation

● Diagramme de séquence système



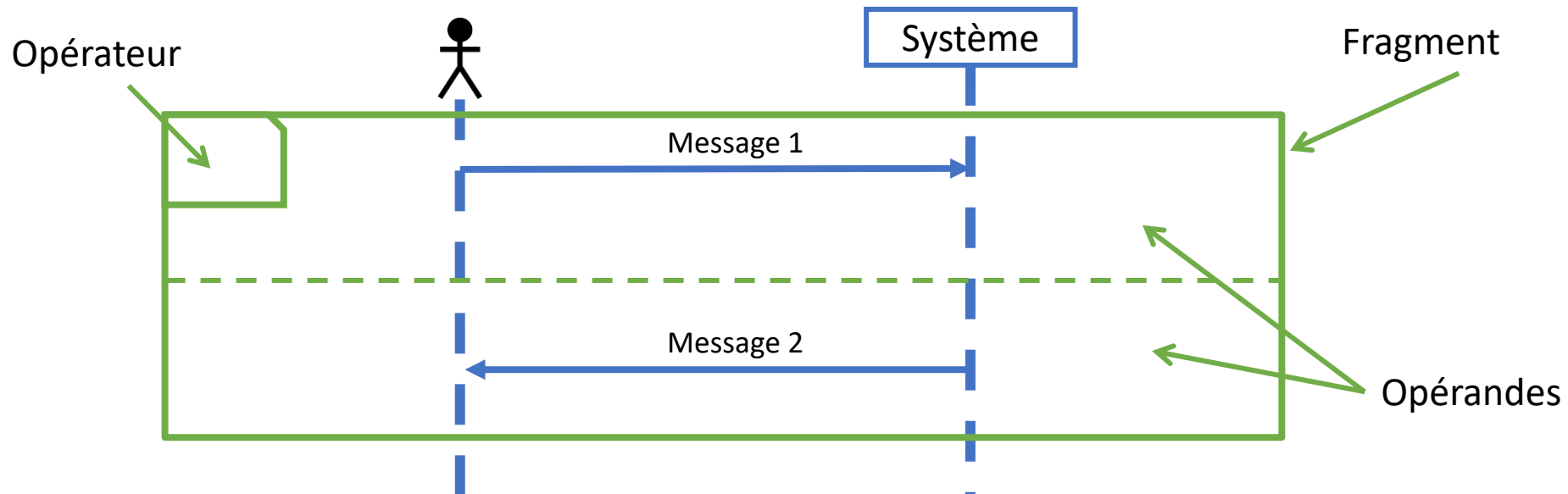
Le Nom de l'objet ou du Rôle est souligné pour indiquer qu'il s'agit d'une instance.

Modèle de cas d'utilisation

5. Descriptions graphiques des cas d'utilisation

● Fragments composés

- représentent les expressions spécifiques dans la séquence.
- doivent couvrir au moins une ligne de vie à tout moment, afin d'avoir une signification.
- Utilisés pour modéliser les scénarios alternatifs et d'exceptions.

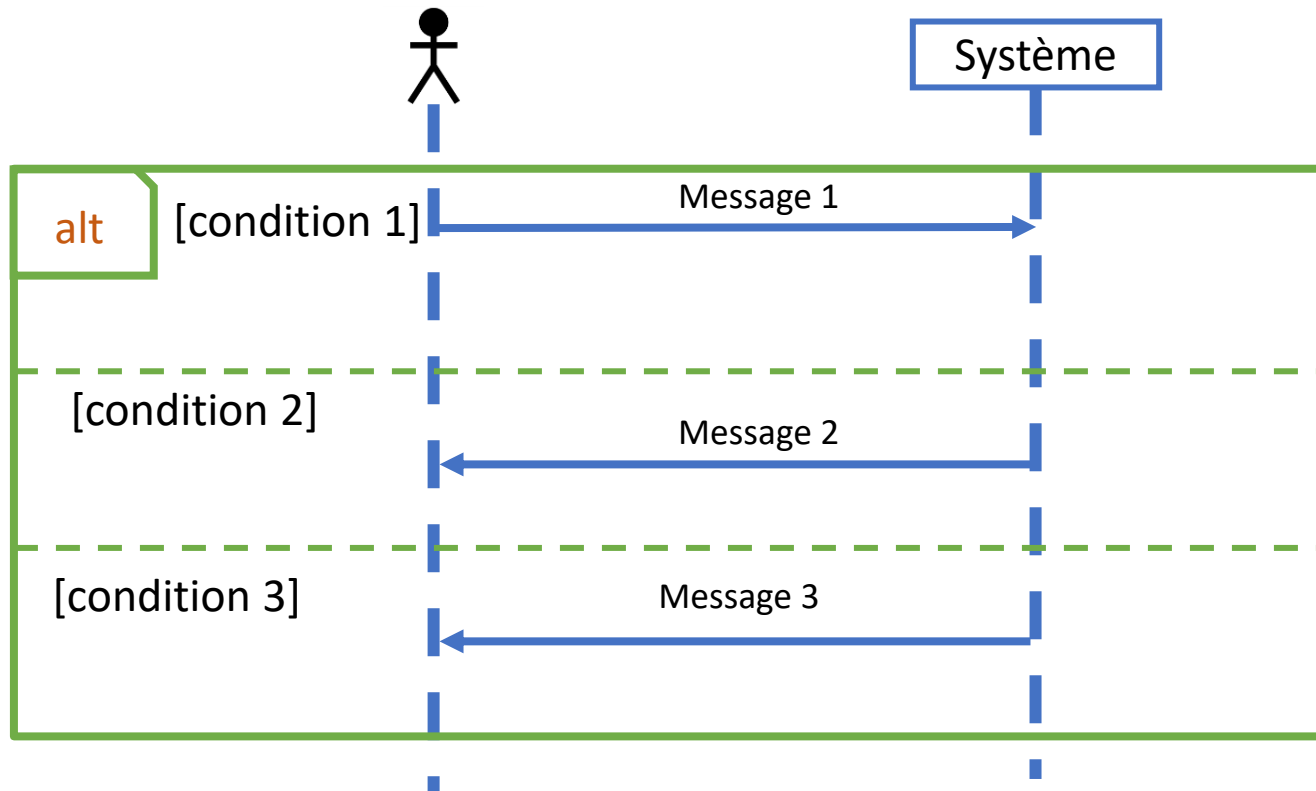


Modèle de cas d'utilisation

5. Descriptions graphiques des cas d'utilisation

● Fragment <<alt>>

- désigne un **choix** ou une **alternative**: équivalent à SI ... ALORS ... SINON

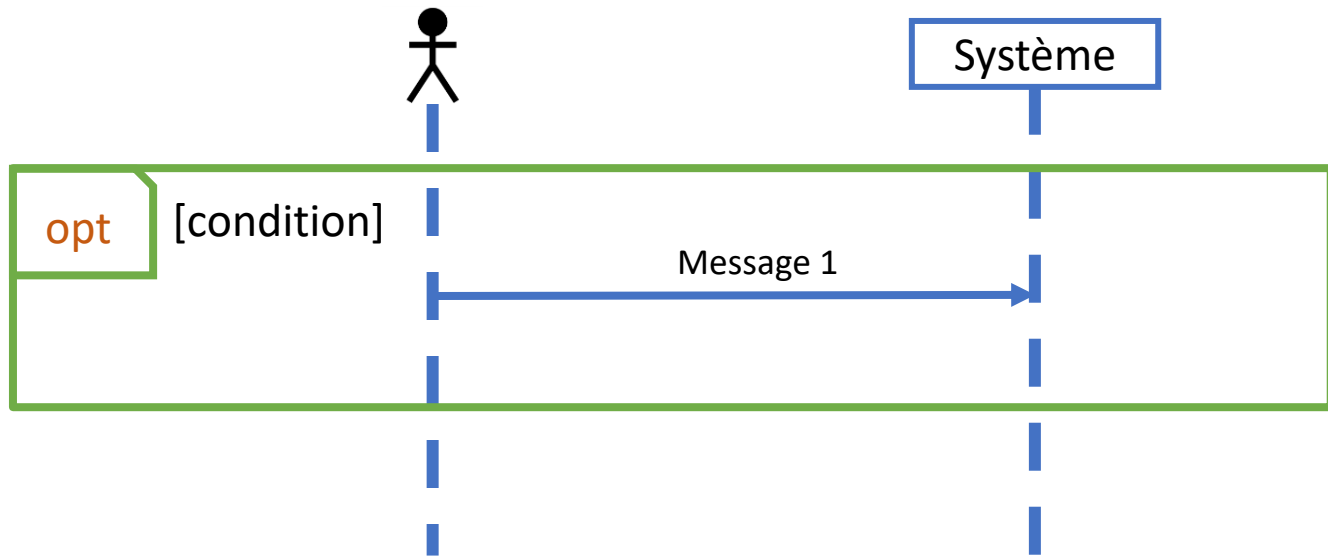


Modèle de cas d'utilisation

5. Descriptions graphiques des cas d'utilisation

- **Fragment <<opt>>**

- équivalent à SI ... ALORS ...
- équivalent à un « alt » sans [else]

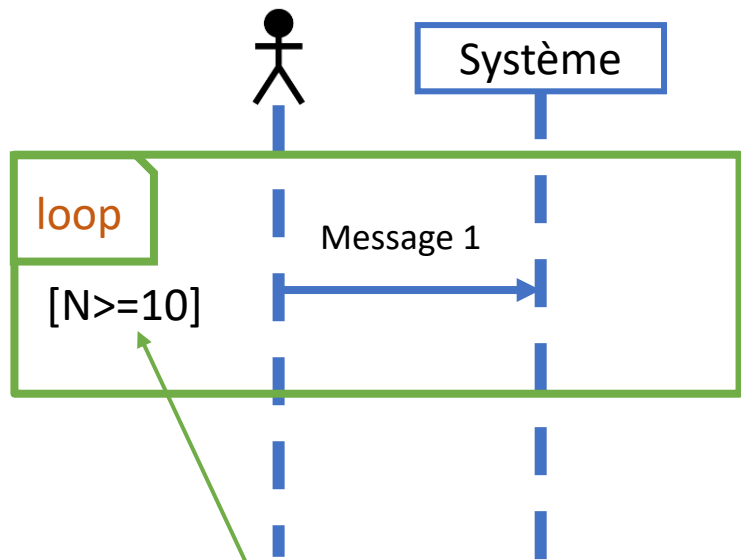


Modèle de cas d'utilisation

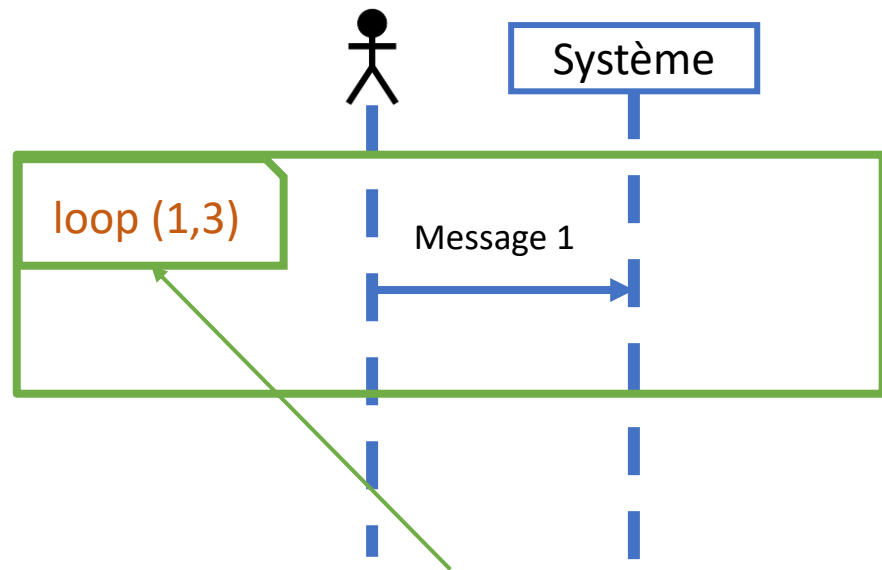
5. Descriptions graphiques des cas d'utilisation

● Fragment <<loop>>

- décrit un ensemble d'interaction qui *s'exécute en boucle*.
- rester dans la boucle tant que la condition entre crochet est vraie.



Condition d'exécution



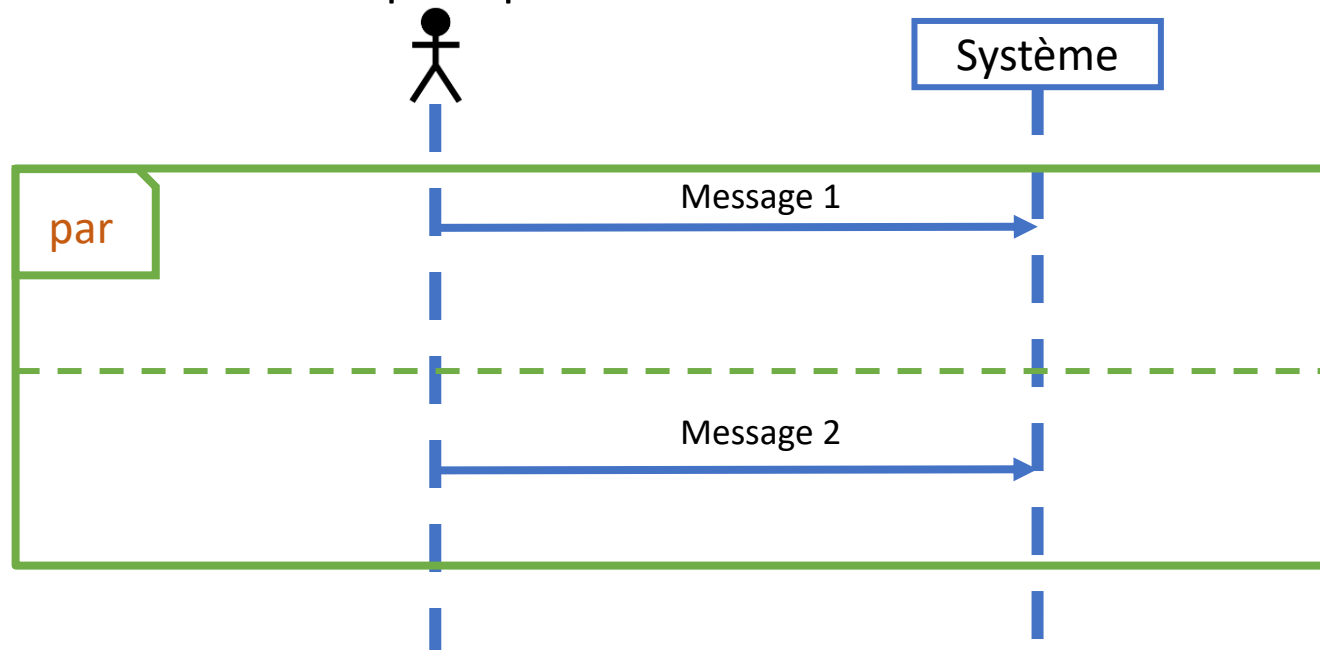
Répéter au moins une fois et 3 fois au maximum

Modèle de cas d'utilisation

5. Descriptions graphiques des cas d'utilisation

● Fragment <<par>>

- représenter des interactions qui se réalisent **en parallèle**
- Possède au moins deux sous-fragments exécutés simultanément.
- La concurrence est **logique** et n'est pas nécessairement physique : les exécutions concurrentes **peuvent s'entrelacer** sur un même chemin d'exécution dans la pratique.

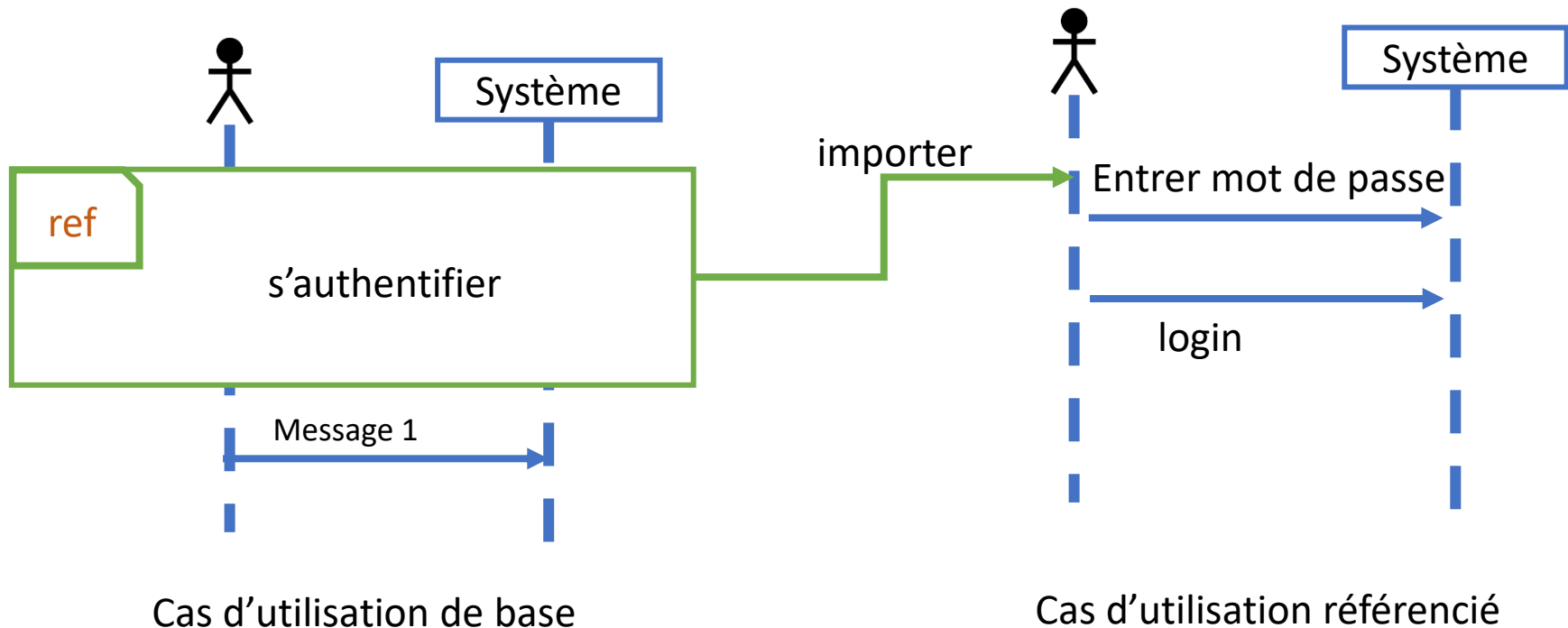


Modèle de cas d'utilisation

5. Descriptions graphiques des cas d'utilisation

● Fragment <<ref>>

- indiquer une **référence** vers un autre diagramme de séquence existant.

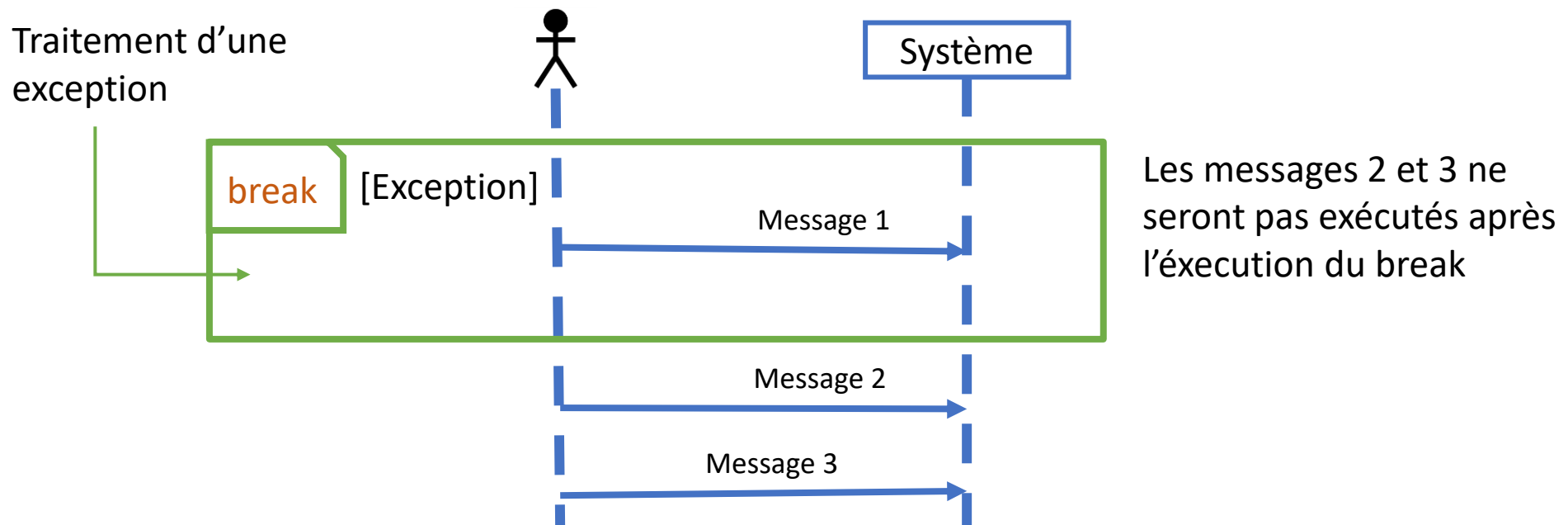


Modèle de cas d'utilisation

5. Descriptions graphiques des cas d'utilisation

● Fragment <<break>>

- Représente un **scenario d'exception**.
- Les interactions de ce fragment seront exécutées à la place des **interactions décrites en dessous**. Il y a donc une notion d'**interruption** du flot "normal" des interactions.



Modèle de cas d'utilisation

Description détaillée des CUs

- **Permet de :**

- Identifier des fonctionnalités communes entre cas d'utilisation pour en déduire des cas d'utilisation plus **précises** qui seront utilisées par **inclusion**, **extension** ou **généralisation/spécialisation**.
- Identifier des interactions de bas niveau.
- Eliminer des inconsistances potentielles.
- Valider les cas d'utilisation avec le client.
 - Les cas d'utilisation facilitent la compréhension des besoins et du système par toutes les parties prenantes.
- Optimiser le diagramme des cas d'utilisation.

Classement des CUs et planification des itérations

Modèle de cas d'utilisation

Classement des cas d'utilisation et planification des itérations

- Classifier les CUs **avant les descriptions** textuelle et graphique et **planifier les itérations** en tenant compte des deux facteurs :
 - la priorité fonctionnelle
 - le risque technique, estimé par le chef de projet
- Développer d'abord les cas primaires, principalement ceux mettant en cause l'acteur principal.
- Les cas secondaires et les cas optionnels seront déterminés ultérieurement.

Modèle de cas d'utilisation

Classement des cas d'utilisation et planification des itérations

- **Exemple : Bibliothèque en ligne**
- Si la priorité est haute et le risque également:
 - Il faut planifier le cas d'utilisation dans une des toutes premières itérations (exemple : Maintenir le catalogue).
- Si la priorité est basse et le risque également:
 - On peut reporter le cas d'utilisation à une des toutes dernières itérations (exemple : Consulter l'aide en ligne).
- Les choses se compliquent lorsque les deux critères sont antagonistes !
 - Le chef de projet doit alors décider en pesant le pour et le contre.

Modèle de cas d'utilisation

Classement des cas d'utilisation et planification des itérations

● Exemple :

Cas d'utilisation	Priorité	Risque	Itéra
Admettre un patient	Haute	Haut	1
Créer le dossier médical d'un patient	Haute	Moyen	2
Mettre à jour le dossier médical d'un patient	Moyenne	Bas	5
S'authentifier	Basse	Moyen	6
Saisir compte rendu d'une consultation d'un patient.	Haute	Bas	3
Enregistrer les soins offerts à un patient.	Haute	Bas	4

Production des maquettes IHM

Production de maquettes IHM

Maquette

- **Produit jetable** donnant aux utilisateurs une vue concrète mais non définitive de **la future interface** de l'application.
- **Simple dessins** d'écrans et descriptions de contenu de fenêtres réalisée avec n'importe quel outil graphique.

Admettre un patient

Nom	<input type="text"/>	Chambre disponible	Option 1 ▼
Prenom	<input type="text"/>	Lits disponible	Option 1 ▼
Date d'entree	<input type="text"/>		
Date de sortie	<input type="text"/>		
Service	<input type="text"/>		

Valider

Admettre le patient

Pourquoi si tôt dans le processus ?

- Aide à la description et validation des Cas d'Utilisation
- Moyen de communication avec le client
- Provoquer des retours de la part des utilisateurs

Conclusion

❑ **Activité très difficile**

❑ Problèmes de compréhension

- ❑ Les développeurs et le client ne parlent pas le même langage

❑ Problèmes de volatilité

- ❑ Une spécification valide peut ne plus l'être après une période de temps

❑ Problèmes humains

- ❑ multiplicité des intervenants (client, utilisateurs, développeurs...)
- ❑ Conflits, Rétention d'information, etc.

Conclusion

Modèle de cas d'utilisations

Diagramme de cas d'utilisation

L'identification des cas d'utilisation se fait lors de la phase d'**Inception** du UP. Il s'agit d'un travail préliminaire visant à définir le périmètre fonctionnel du projet.

Description des cas d'utilisation

La description détaillée (textuelle et graphique) des cas d'utilisation se fait principalement lors de la phase d'**Élaboration**, mais de façon itérative et incrémentale.

Phases

- Etude préliminaire

- Identification des acteurs et cas d'utilisation
- Structuration des CUs en Packages
- Identification des relations

- Description textuelle des CUs
- Description graphique des CUs

