



université  
PARIS-SACLAY

université  
PARIS-SACLAY  
IUT D'ORSAY

# QUALITÉ DE DÉVELOPPEMENT

## DIAGRAMME UML DE SÉQUENCE




🎓 2A - Bachelor Universitaire de Technologie  
🏛️ IUT d'Orsay - Université Paris-Saclay - 2024/2025



**Idir AIT SADOUNE**

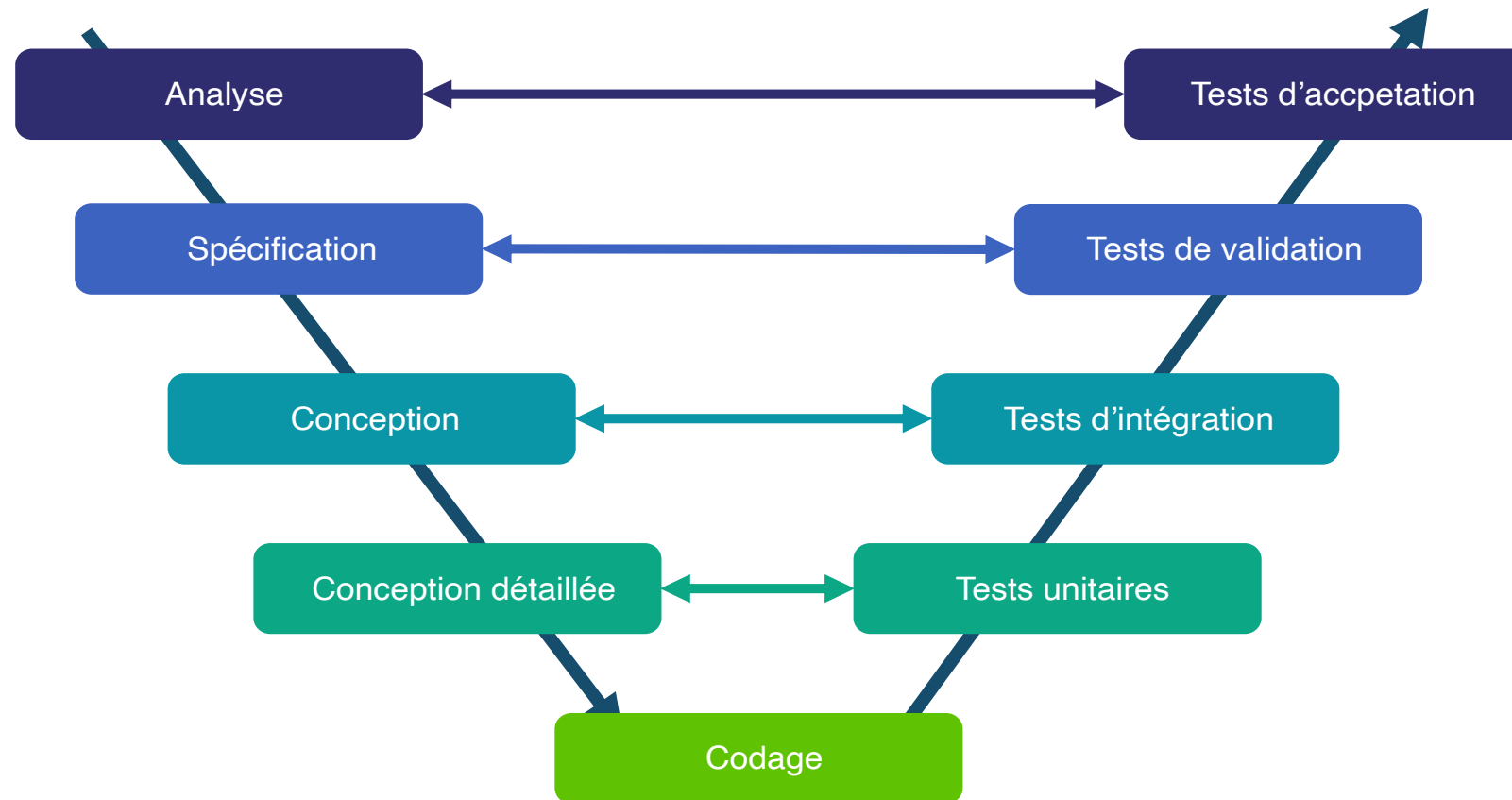
[idir.ait-sadoune@universite-paris-saclay.fr](mailto:idir.ait-sadoune@universite-paris-saclay.fr)

# PLAN

-  [Eléments de base](#)
-  [Fragments combinés](#)
-  [Utilisation du diagramme](#)

[Retour au plan](#) - [Retour à l'accueil](#)

# CYCLE DE DÉVELOPPEMENT



Ce cours se positionne dans les étapes **Spécification** et **Conception**.

# PLAN

> Eléments de base

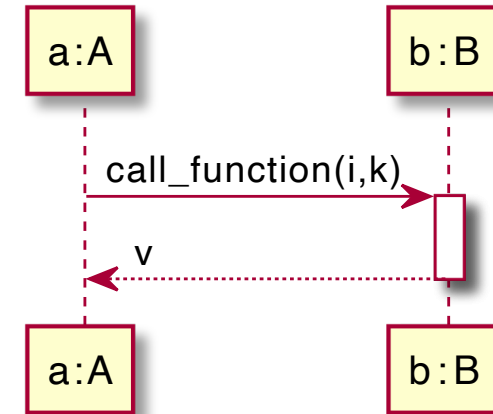
> Fragments combinés

> Utilisation du diagramme

[Retour au plan](#) - [Retour à l'accueil](#)

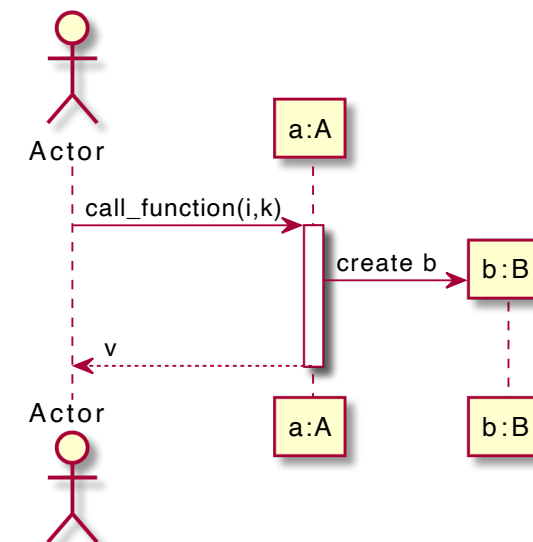
# DÉFINITIONS

- Les **diagrammes de séquence** ont été introduits pour permettre de décrire les **scénarios de communications entre objets**.
  - ▢ décrire les **messages échangés** pour la **réalisation d'une fonctionnalité**.
  - ▢ identifier **les liens** et **les méthodes** nécessaires pour les objets.
- Un **diagramme de séquence** représente :
  - les **entités** (objets/acteurs) **verticalement** par des **lignes de vie**
  - les **messages** **horizontalement** par **des flèches**



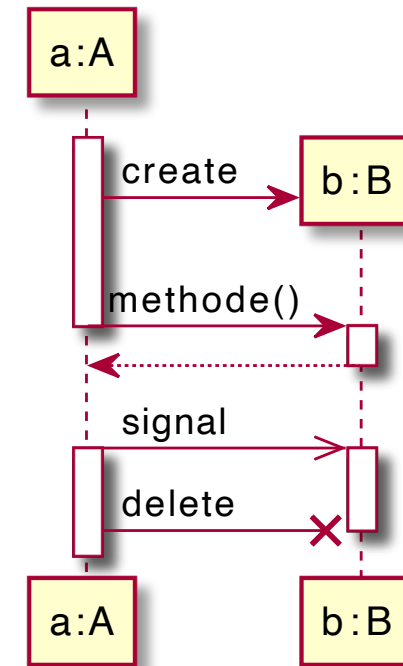
# LES ENTITÉS

- Un **diagramme de séquence** décrit des interactions entre deux **entités** principales :
  1. un **acteur** → une entité extérieur au système
  2. un **objet** → une entité du système
- Chaque **entité** a une **ligne de vie** représentée par une **ligne verticale**
  - ▀ en **pointillée** lorsque l'entité est **inactive**
  - ▀ en **bloc** lorsque l'entité est **active**



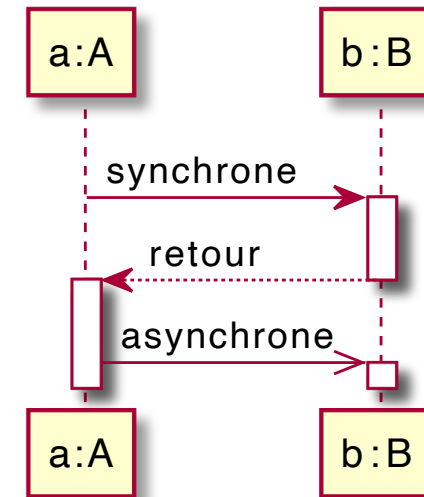
# LES MESSAGES

- Un **diagramme de séquence** permet de spécifier différents **types de communications** :
  - création d'un objet
  - appels de méthodes
  - envoi d'un signal
  - destruction d'un objet ...



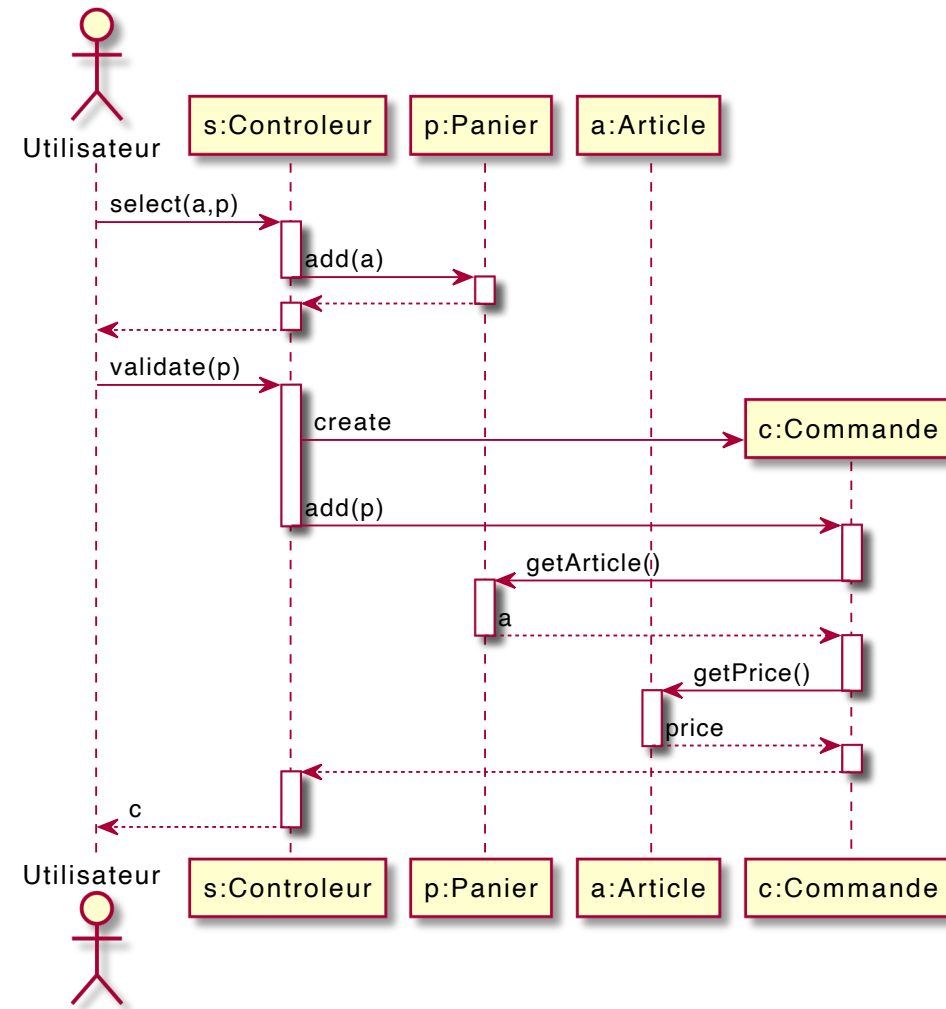
# LES MESSAGES

- Un **diagramme de séquence** utilise **trois types de messages** pour spécifier une communication :
  1. **message synchrone** :  
l'entité attend la réponse avant de continuer
  2. **message de retour** :  
la réponse d'une entité à un message
  3. **message asynchrone** :  
l'entité n'attend pas la réponse pour continuer





# EXEMPLE



# PLAN

- Éléments de base
- Fragments combinés
- Utilisation du diagramme

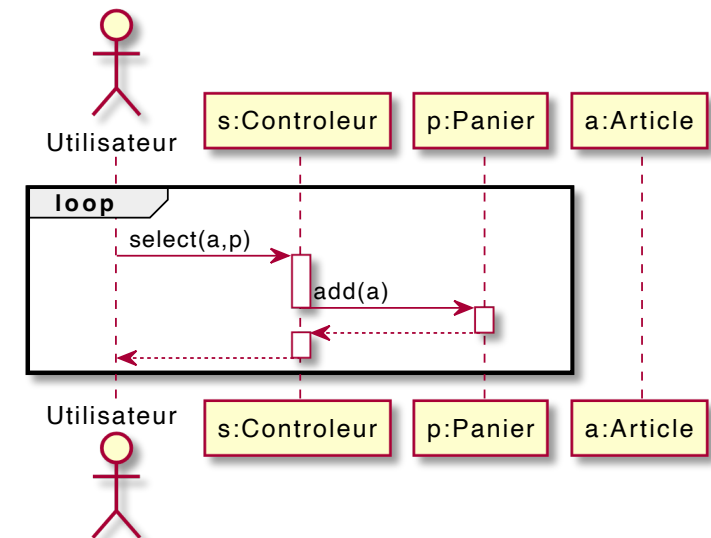
[Retour au plan](#) - [Retour à l'accueil](#)

# LES FRAGMENTS COMBINÉS

- Dans un **diagramme de séquence**, il est possible :
  - de représenter des **contraintes** ou des **propriétés** particulières (exécution atomique, répétition, ...).
  - de **décomposer une interaction complexe** en fragments simples.
- **Le fragment combiné** est l'élément graphique permettant de **représenter ce type d'information**.

# LES FRAGMENTS COMBINÉS

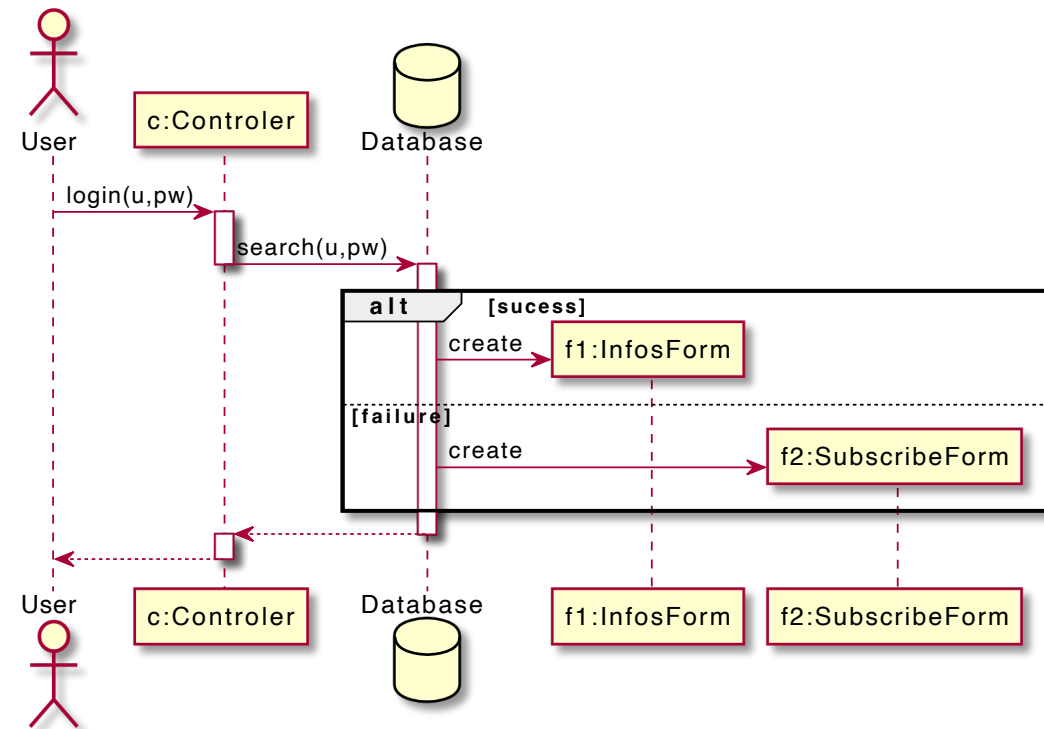
- **Un fragment combiné** est constitué de :
  - un **opérateur d'interaction** (type de la combinaison)
  - un ou plusieurs **fragments d'interaction** (une partie du diagramme)
- **L'opérateur d'interaction** est indiqué dans le coin supérieur gauche dans un **rectangle**.



# LES FRAGMENTS COMBINÉS

## ALTERNATIVE

**Alternative - alt :**  
sélectionner un comportement en fonction  
d'une condition.

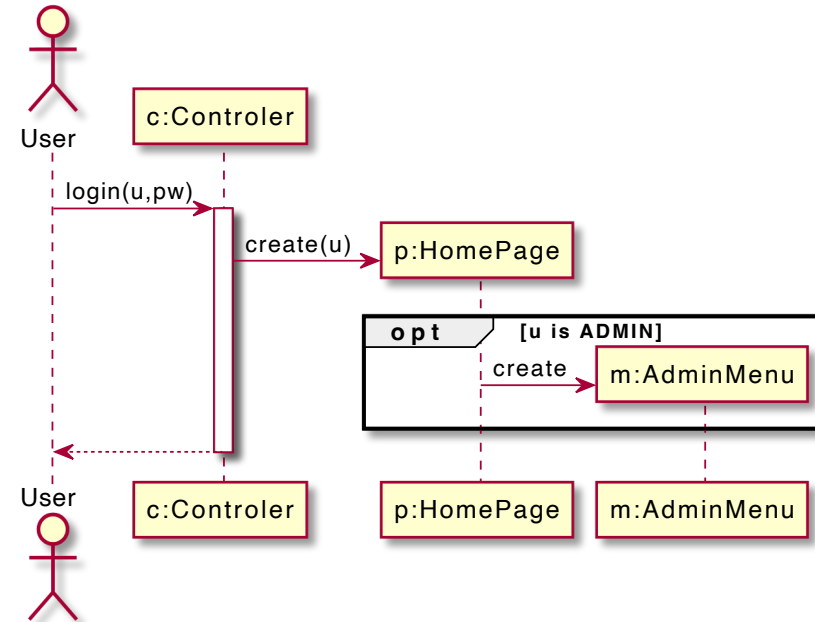


# LES FRAGMENTS COMBINÉS

## OPTION

### Option - opt :

exécuter un comportement si la condition de garde est vérifiée.

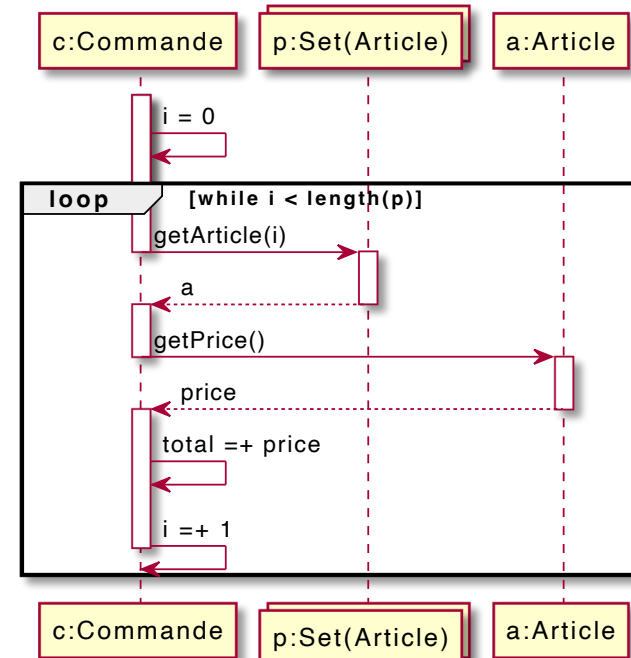


# LES FRAGMENTS COMBINÉS

## BOUCLE

### Boucle - loop :

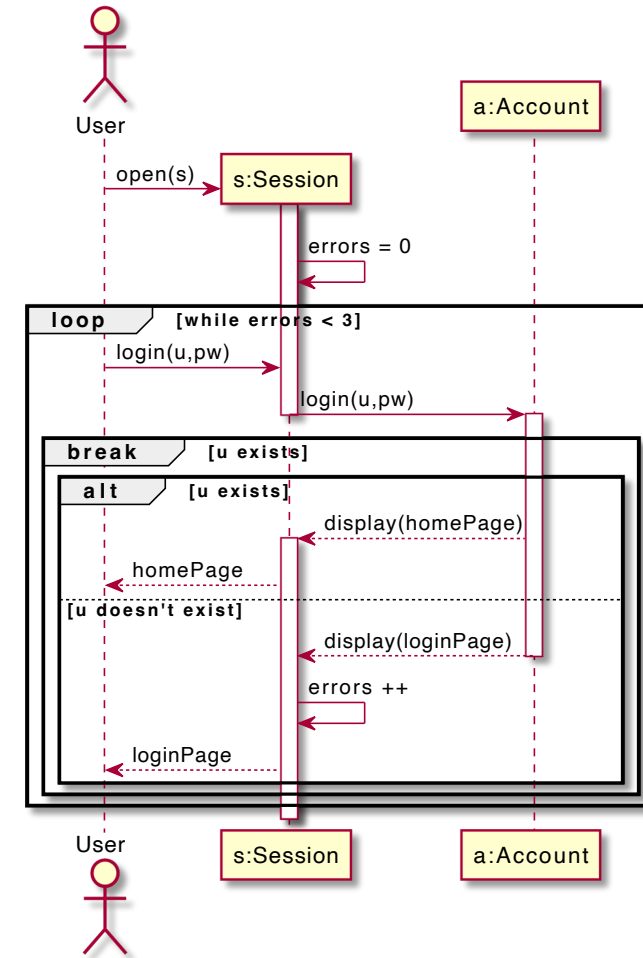
exécuter une interaction **tant qu'une condition est satisfaite**.



# LES FRAGMENTS COMBINÉS

## RUPTURE

**Rupture - break :**  
exécuter le fragment associé puis **mettre fin**  
à l'interaction englobante.

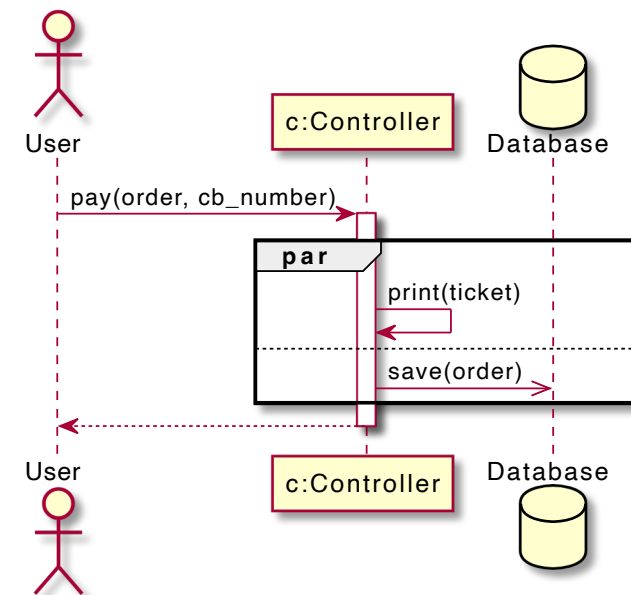




# LES FRAGMENTS COMBINÉS

## PARALLÈLE

**Parallèle - par :**  
les fragments d'interaction associés sont  
exécutés en parallèle.



# LES FRAGMENTS COMBINÉS

## QUELQUES OPÉRATEURS SUPPLÉMENTAIRES

- **ref** : appeler une interaction décrite par ailleurs.
- **strict** : l'ordre d'exécution doit être strictement respecté.
- **weak** : l'ordre d'exécution des opérations n'a pas d'importance.
- **ignore** : certains messages peuvent être absents sans incidence.
- **consider** : certains messages doivent être obligatoirement présents.
- **critical** : une séquence d'interactions ne peut être interrompue (séquence critique est atomique).

# PLAN

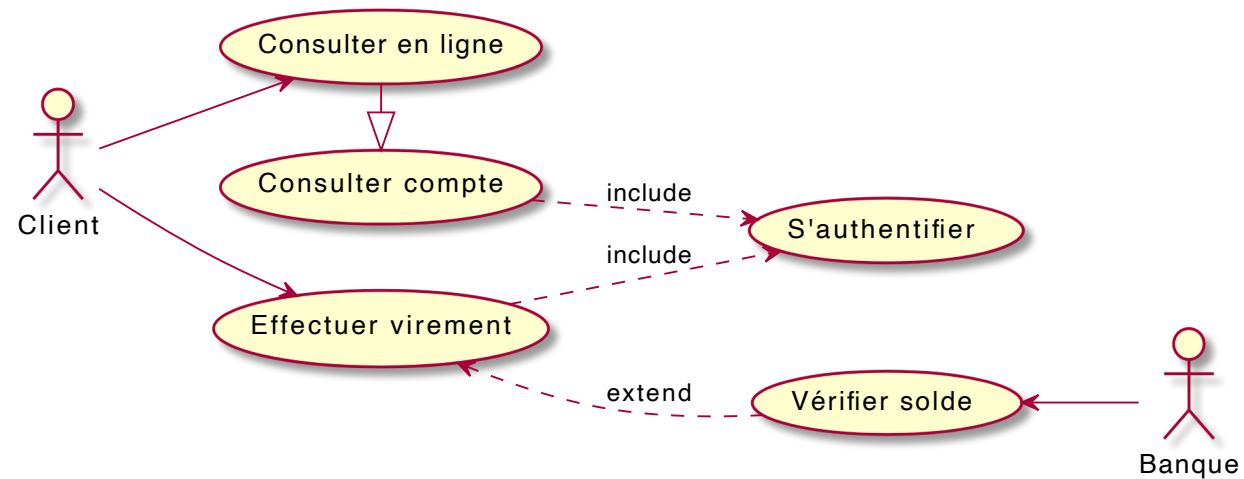
- Éléments de base
- Fragments combinés
- Utilisation du diagramme

[Retour au plan](#) - [Retour à l'accueil](#)

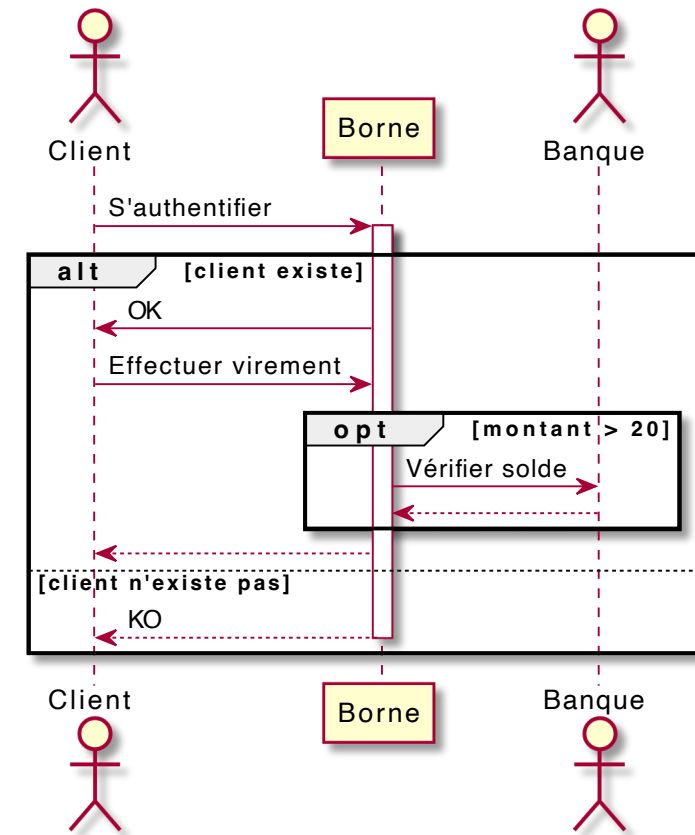
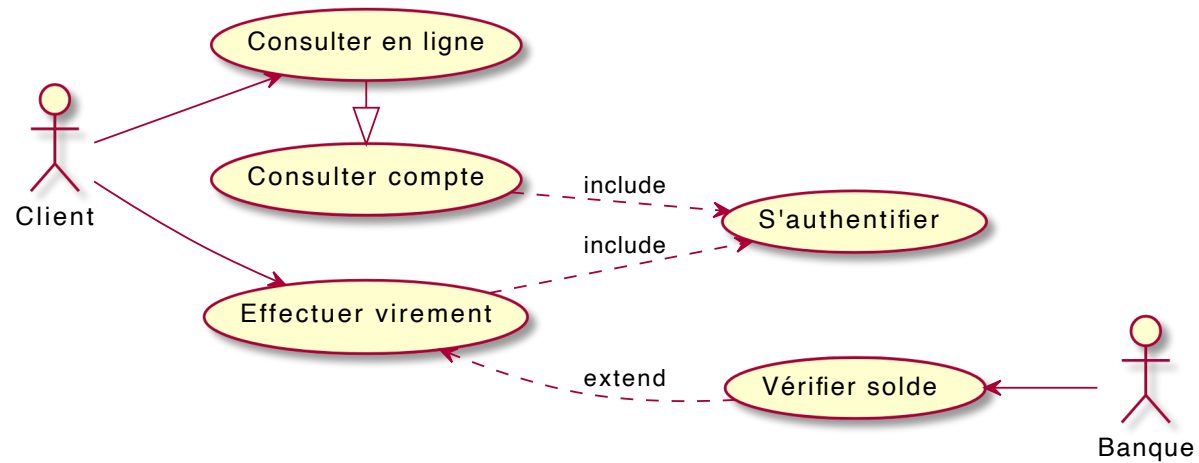
# EN COMPLÉMENT DU DIAGRAMME DE CAS D'UTILISATION SPÉCIFICATION

- à la place de **la description textuelle** ou du **diagramme d'activité** décrivant les scénarios du diagramme de cas d'utilisation.
- pour décrire les **flux d'informations échangés** pour la **réalisation d'un cas d'utilisation**.

# EN COMPLÉMENT DU DIAGRAMME DE CAS D'UTILISATION EXEMPLE



# EN COMPLÉMENT DU DIAGRAMME DE CAS D'UTILISATION EXEMPLE

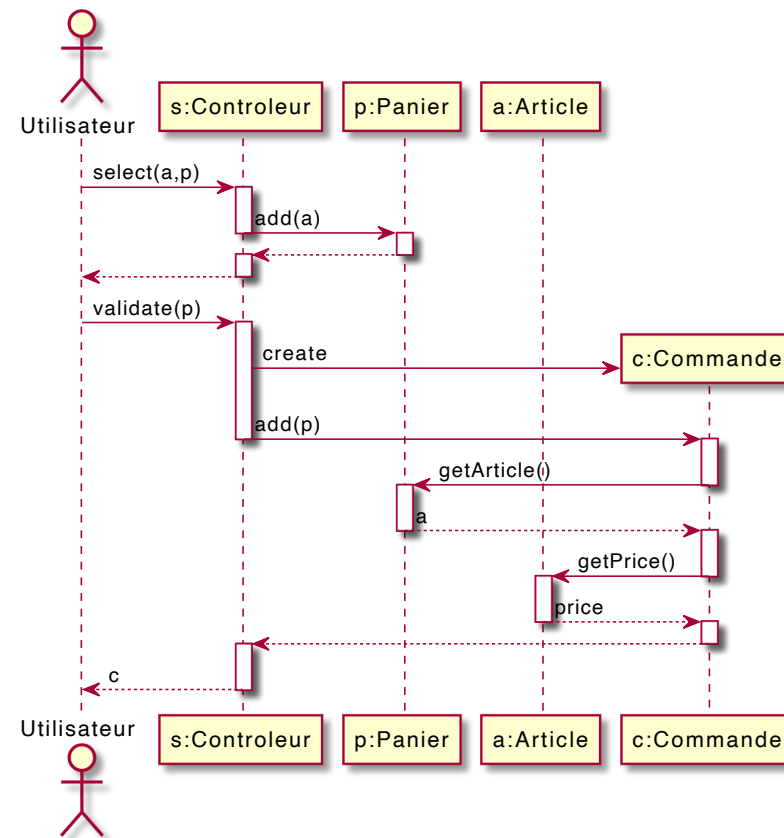


# EN COMPLÉMENT DU DIAGRAMME DE CLASSE

## CONCEPTION

- identifier **les liens** entre les objets (les classes).
- identifier **les méthodes** nécessaires pour les objets (les classes).

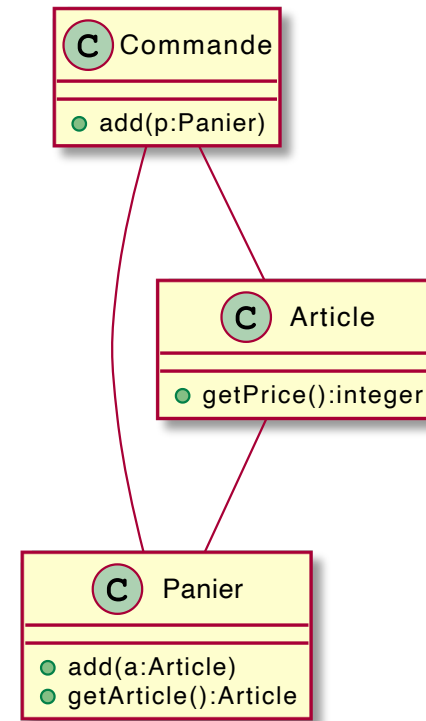
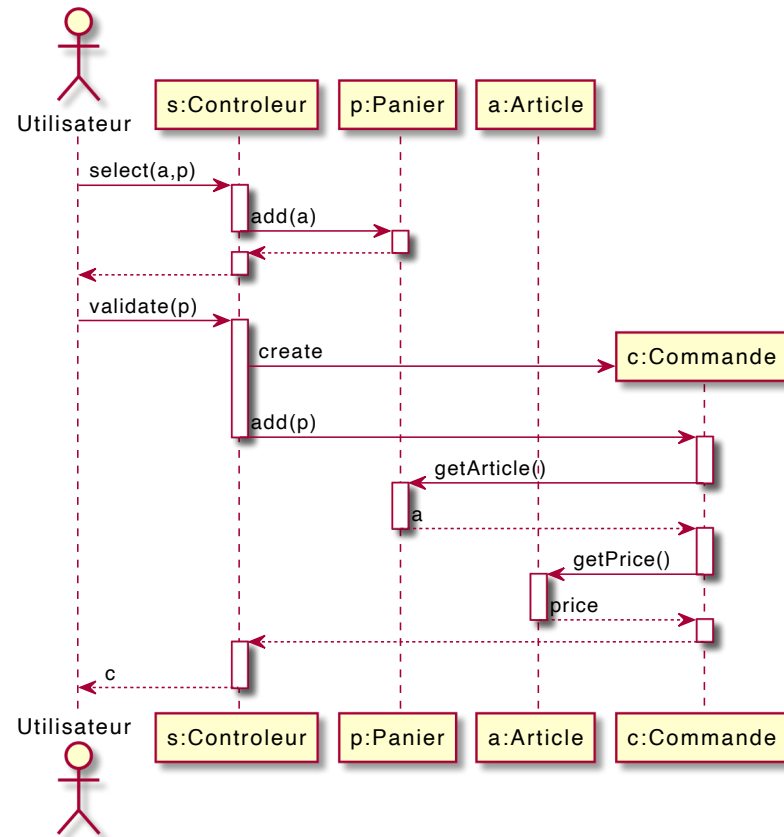
# EN COMPLÉMENT DU DIAGRAMME DE CLASSE EXEMPLE





# EN COMPLÉMENT DU DIAGRAMME DE CLASSE

## EXEMPLE



# MERCI

[Version PDF des slides](#)

[Retour à l'accueil](#) - [Retour au plan](#)