



Génie logiciel

Diagramme de séquence

Louis-Edouard LAFONTANT



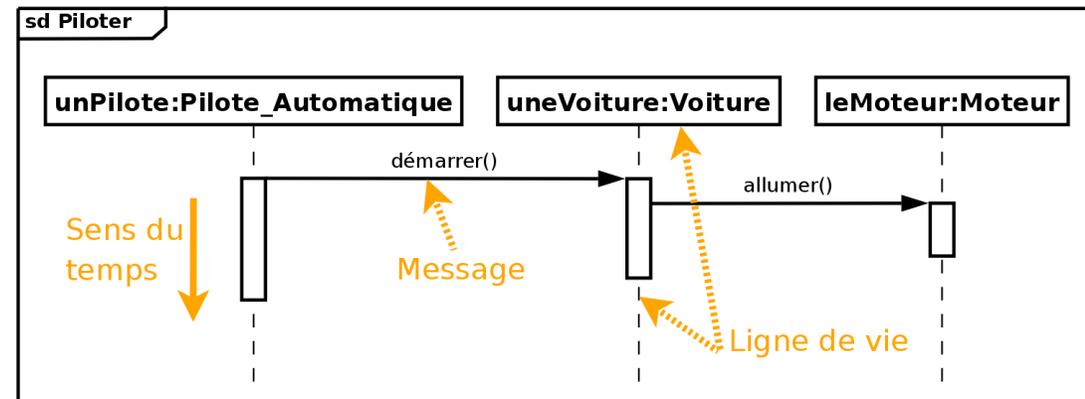


Du CU au diagramme de séquence

- Un CU est réalisé par une collaboration
 - Ensemble d'objets qui s'échangent des messages et travaillent ensemble pour accomplir une tâche
 - Un diagramme de séquence
 - Décrit les CU en mettant en évidence les **interactions** entre les instances des classes (objets) du logiciel
 - Montre la séquence **au cours du temps** des échanges de messages entre les objets participant à un scénario
-

Diagramme de séquence

- Capture le comportement dynamique du système
- Composé d'entités dynamiques
 - Les objets (pas de classes à l'exécution)
 - Leur ligne de vie
 - Les messages échangés



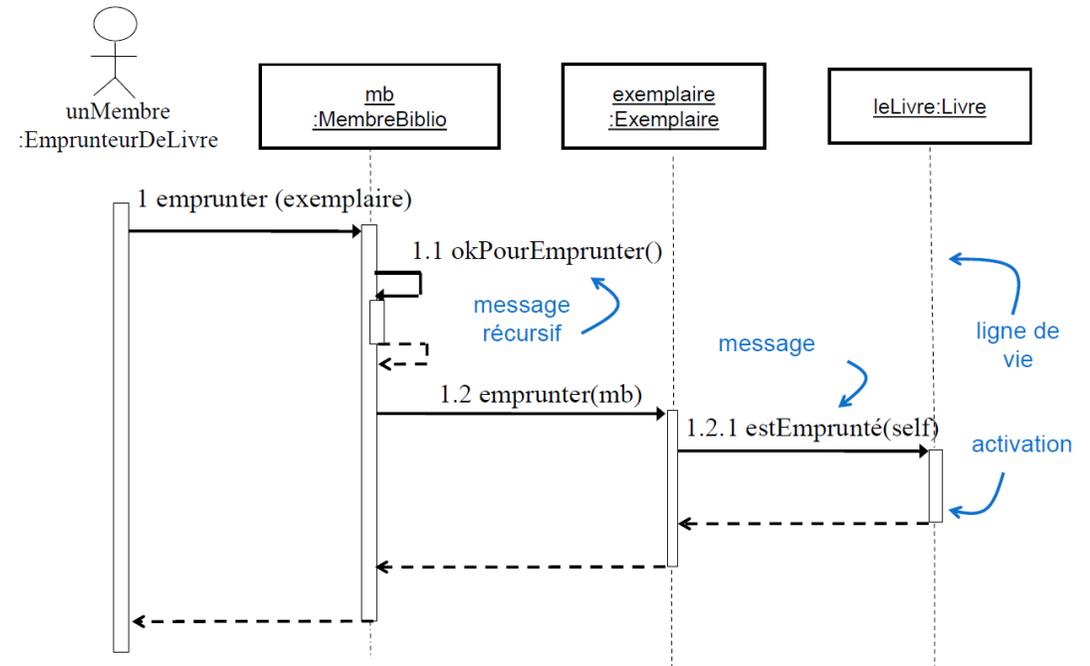
Dimensions

A thick yellow horizontal bar spans the width of the slide, with a vertical yellow bar extending downwards from its right end.

- Dimension verticale : le temps
 - L'ordre d'envoi d'un message est déterminé par sa position sur l'axe vertical du diagramme
 - Le temps s'écoule de haut en bas
- Dimension horizontale : les objets (et les acteurs)
 - L'ordre de disposition des objets sur l'axe horizontal est sans importance (visez l'intuitivité)

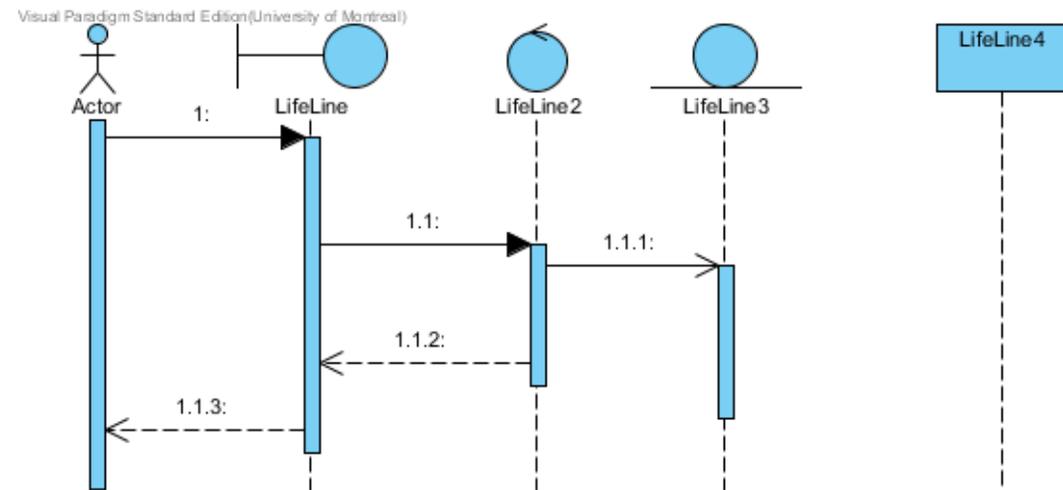
Éléments graphiques

- Les **objets** qui interagissent dans le scénario
- Représentation graphique de la **ligne de vie** de chaque objet et de ses activations
- Les différents **types de messages** envoyés
 - simple, synchrone, asynchrone
- Les **indications de contrôle**
 - Branchement conditionnel et itération
 - Création et destruction d'objets
 - Délais de transmission
 - Contraintes temporelles



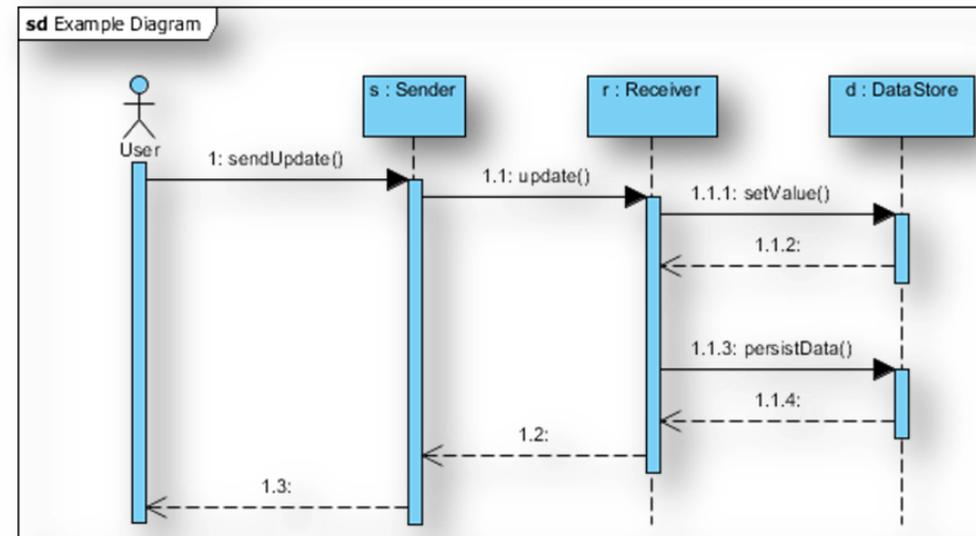
Acteurs et objets

- Chaque ligne de vie représente un acteur ou objet différent
- La ligne de vie représente la présence de l'instance
- L'activation représente une période d'activité ou d'exécution
- Peut avoir la forme des stéréotypes

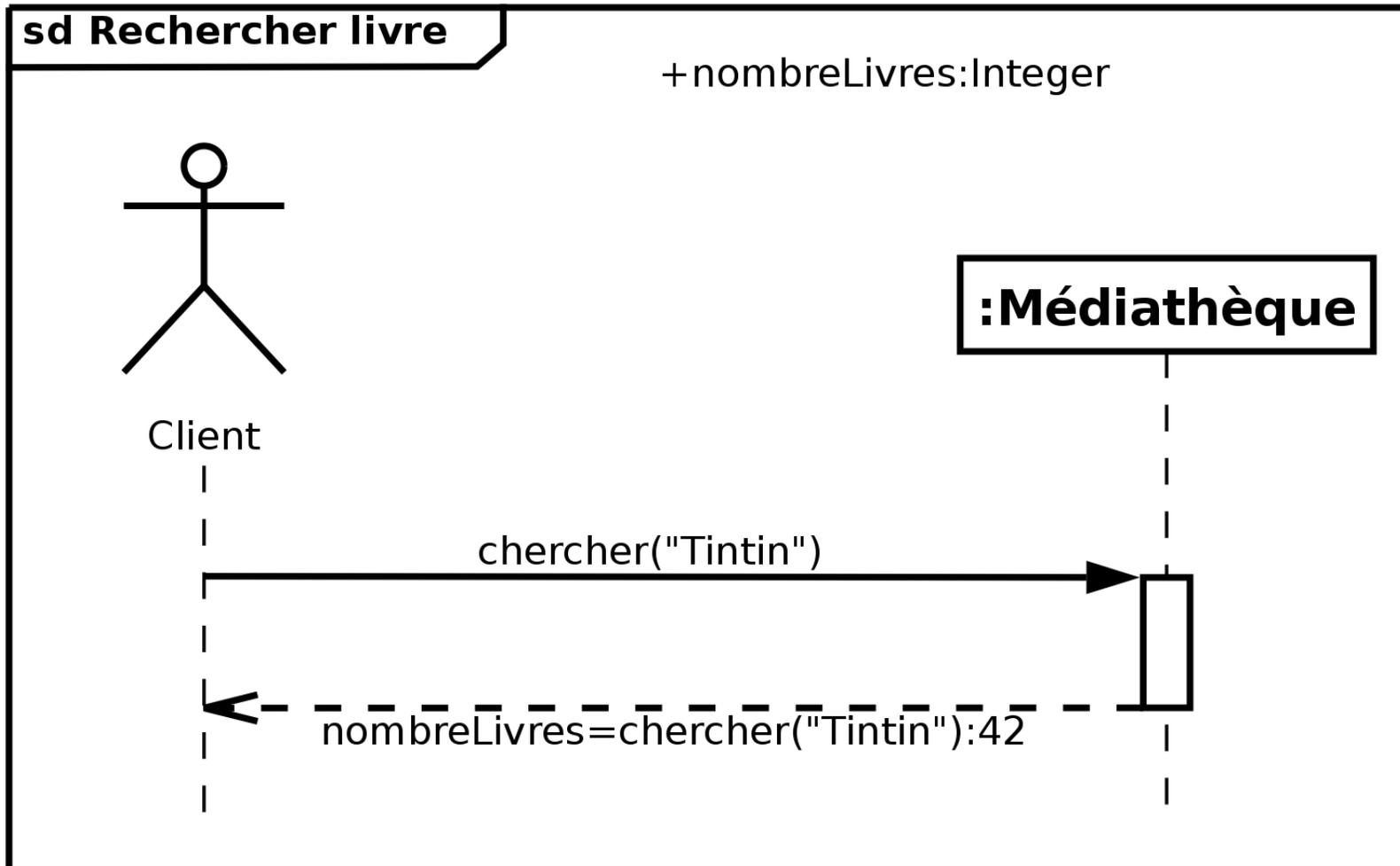


Messages

- Messages numérotés de façon à refléter l'imbrication des envois
 - Message 1.1.3 envoyé après que la réponse au message 1.1.2 ait été reçue
 - Attendre la réponse 1.1.4 avant de pouvoir répondre au message 1.1

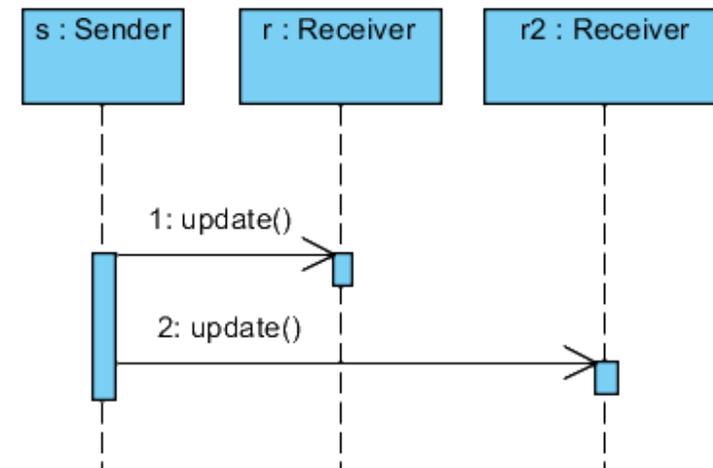
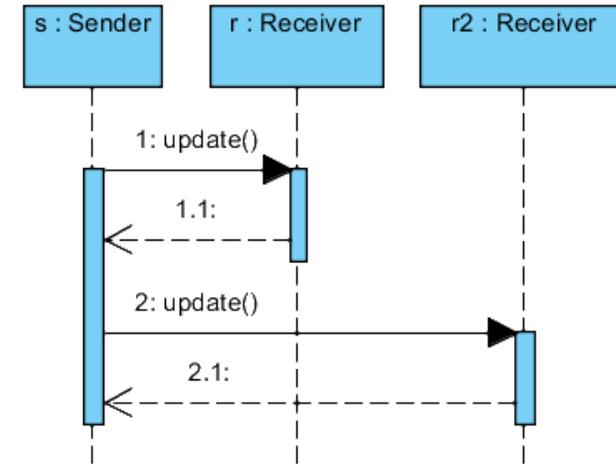


Message de retour



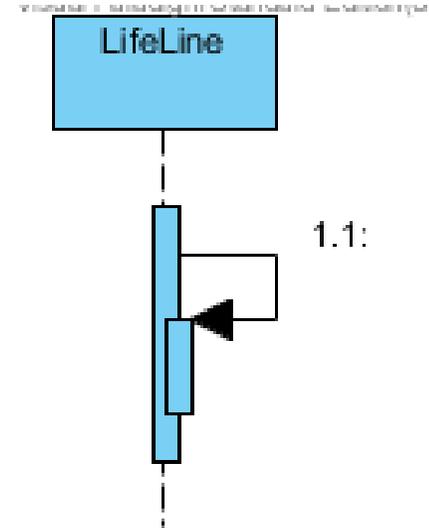
Flux de messages

- Les objets actifs envoient deux types de messages
 - **Synchrone** : attend la réponse du destinataire avant de poursuivre
 - **Asynchrone** : poursuit son activité sans attendre la réponse à son message; elle lui sera signalée ultérieurement



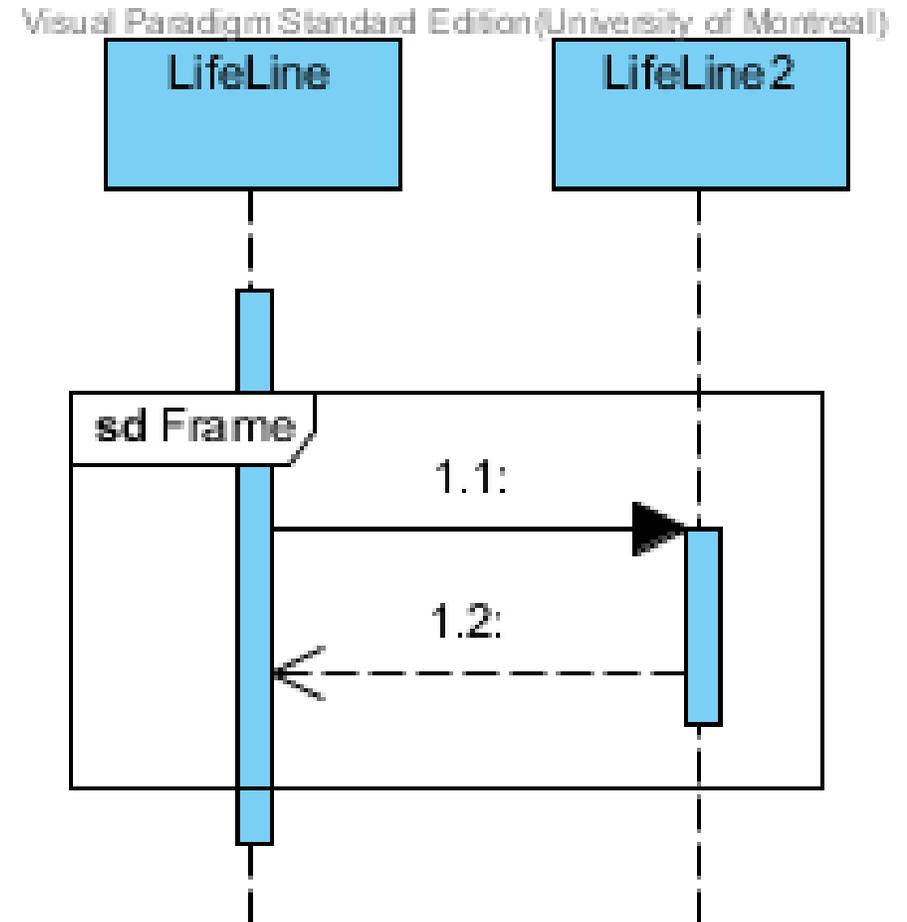
Messages à soi-même

- Message échangé avec le même objet
- Typiquement pour les appels de méthodes privées, mais pas nécessairement



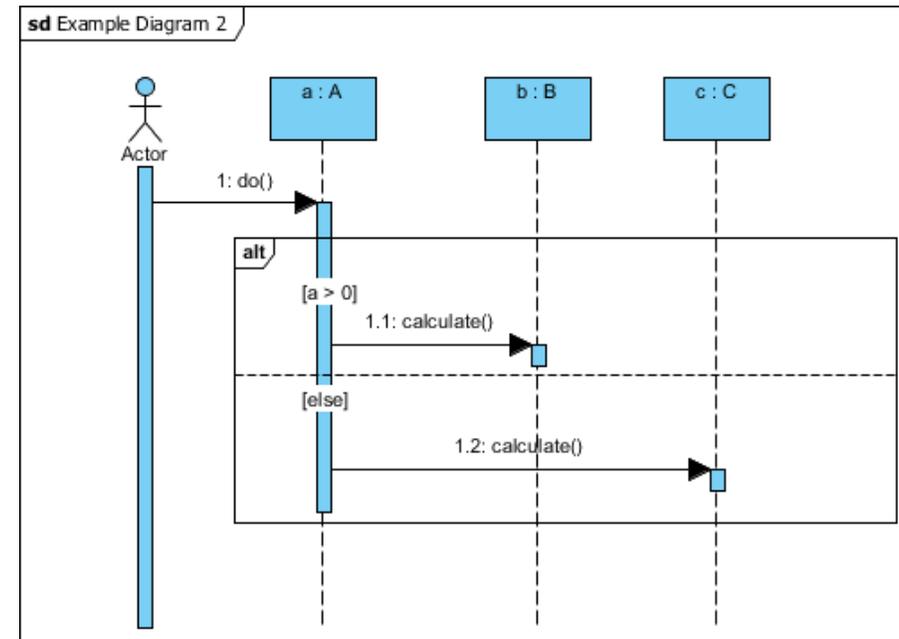
Fragments

- Sections démarquée d'un diagramme de séquence
- Indique:
 - un **comportement particulier**, comme un cycle
 - une **référence** à un autre diagramme de séquence
 - un segment **optionnel ou conditionnel**
- Défini par:
 - un rectangle qui démarque les éléments affectés par le fragment
 - une étiquette avec l'indication du type de fragment et le cas échéant un nom



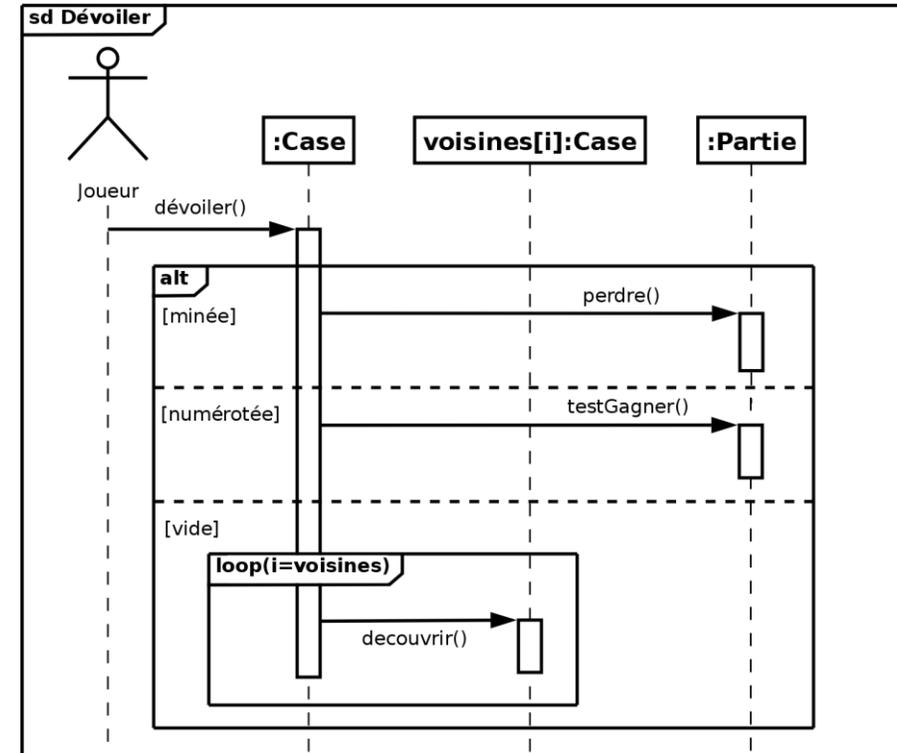
Fragment ALT

- **alt**: fragment avec des **alternatives** pour la logique conditionnelle
 - conditions sont exprimées dans les gardes
 - comportement if-then-else
 - seulement un des segments est exécuté



Fragment LOOP

- **loop**: possède un sous-fragment et spécifie un compte minimum et maximum (boucle) ainsi qu'une condition de garde.
 - comportement for, while





Exemple

Cohérence avec diagramme de classe

Quel serait un diagramme de classe pour ce diagramme de séquence?

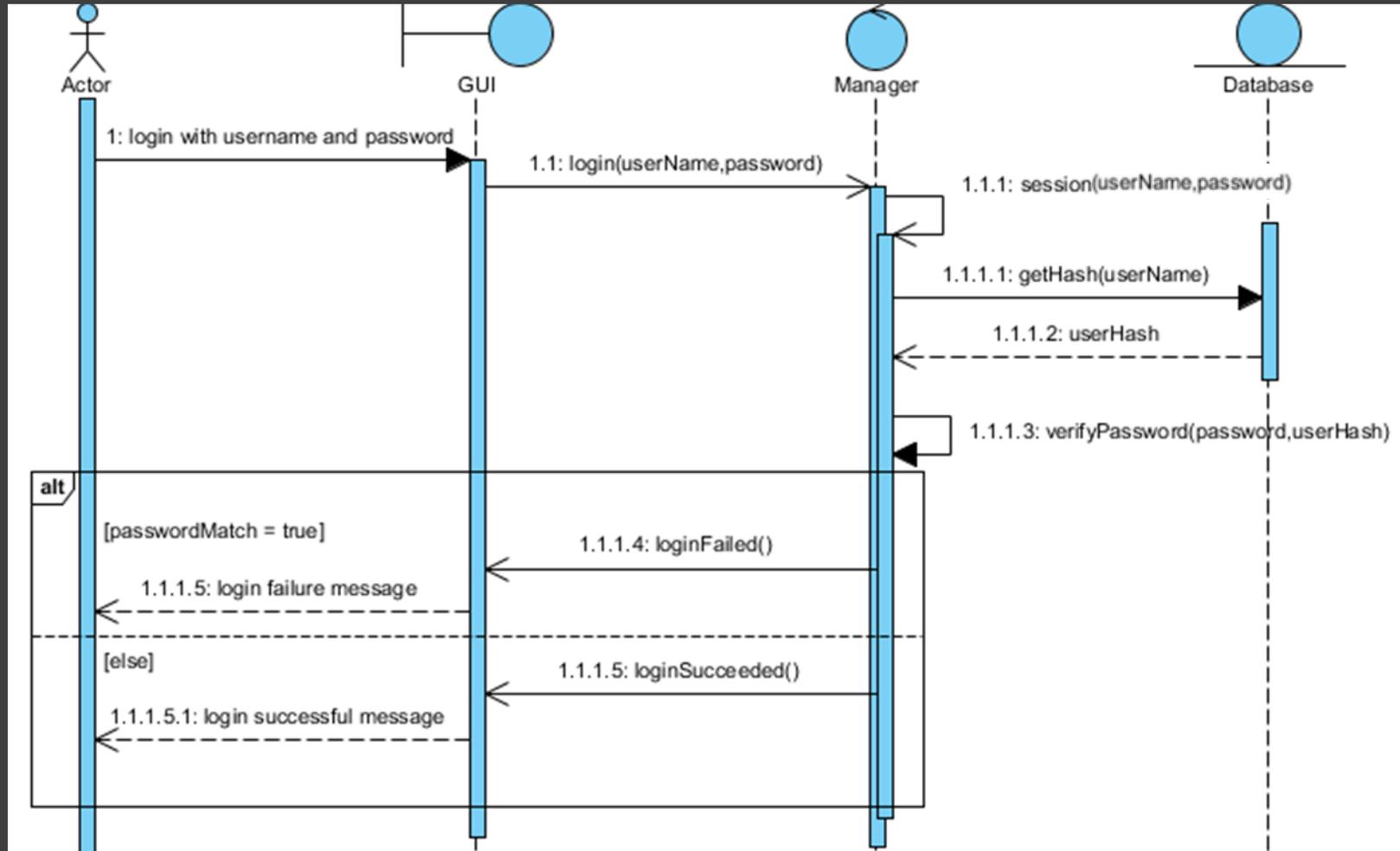


Diagramme de classe proposé

