

# 4.1. Iterazione 4, Analisi

## 4.1.1 Introduzione

Per l'iterazione 4 sono stati scelti i seguenti requisiti:

- lo scenario principale di successo e le estensioni dei casi d'uso CRUD UC8 (Gestisci Consegne) e UC10 (Gestisci Componente);
- lo scenario principale di successo e le estensioni del caso d'uso UC7 (Ricerca Vendite).
- lo scenario principale di successo del caso d'uso UC12 (Arrivo Ordine).

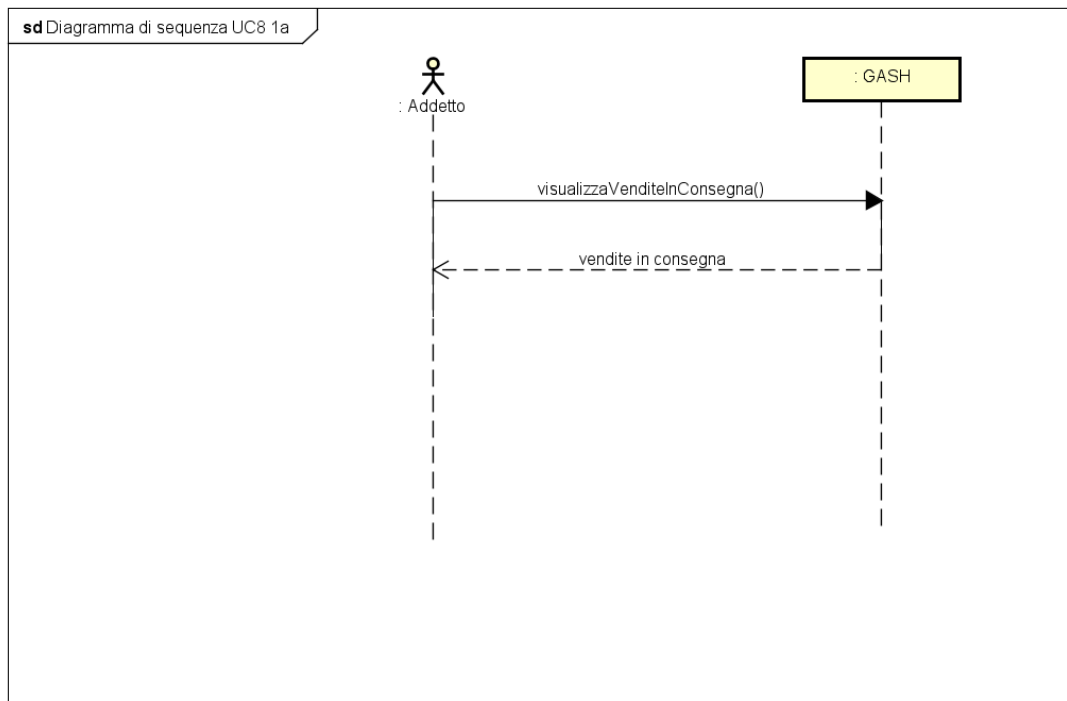
## 4.1.2 Caso d'uso UC8: Gestisci Consegne, Modello di dominio

In quest'iterazione del caso d'uso UC8 vengono analizzati lo scenario principale di successo e le estensioni 1a, 1b e 1c. Creare una nuova classe Consegna risulterebbe poco efficiente, in quanto tutte le informazioni relative alla consegna di porte per garage sono già memorizzate nella classe VenditaPorte, sottoclasse di Vendita. La soluzione a questo problema risiede nell'applicazione del pattern State. I dettagli verranno analizzati in seguito, durante la fase di progettazione.

Non è necessario realizzare le operazioni relative all'inserimento, alla modifica e all'eliminazione di una consegna, poiché sono già comprese nei casi d'uso UC3 (Nuova Vendita Porte per garage) e UC7 (Ricerca Vendite).

## 4.1.3 Caso d'uso UC8, Diagramma di sequenza di sistema

### 4.1.3.1 Estensione 1a



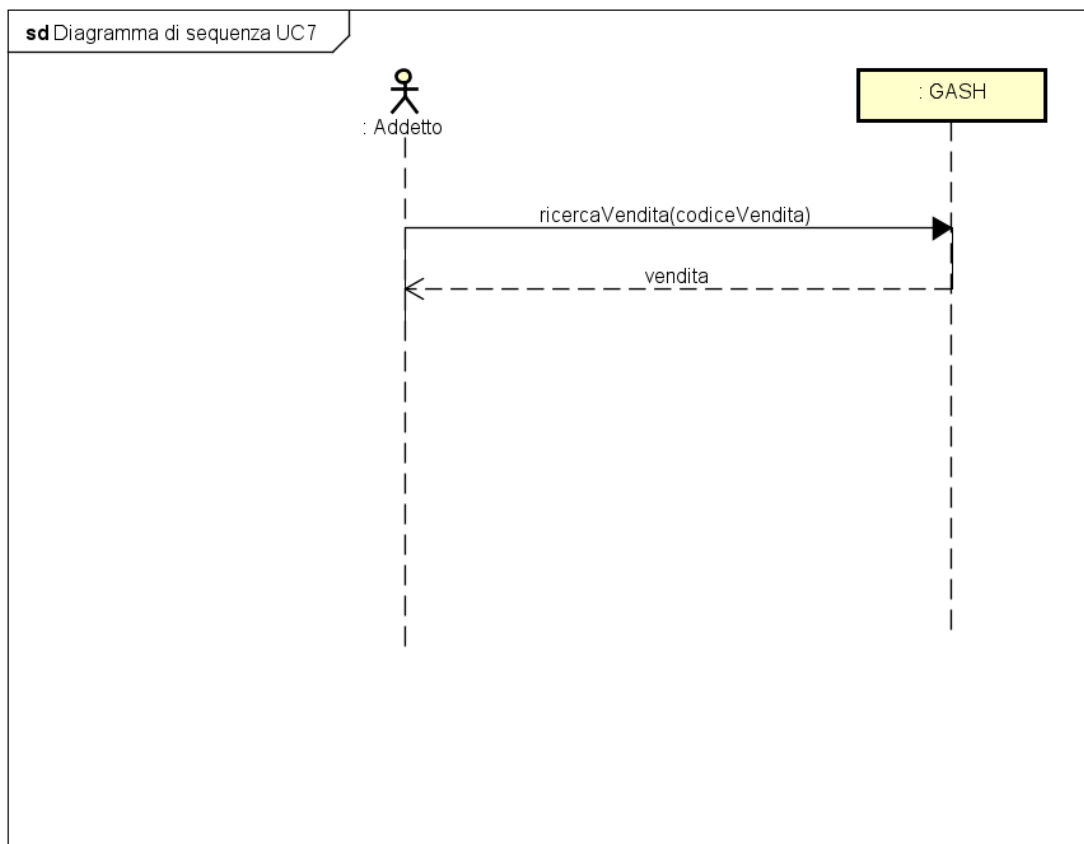
In modo del tutto analogo è possibile realizzare dei diagrammi di sequenza che mostrano come ricavare tutte le vendite che hanno stato “Consegnato”, tutti gli ordini con stato “InArrivo” e tutti gli ordini con stato “Processato”.

#### 4.1.4 Caso d’uso UC7: Ricerca Vendite, Modello di dominio

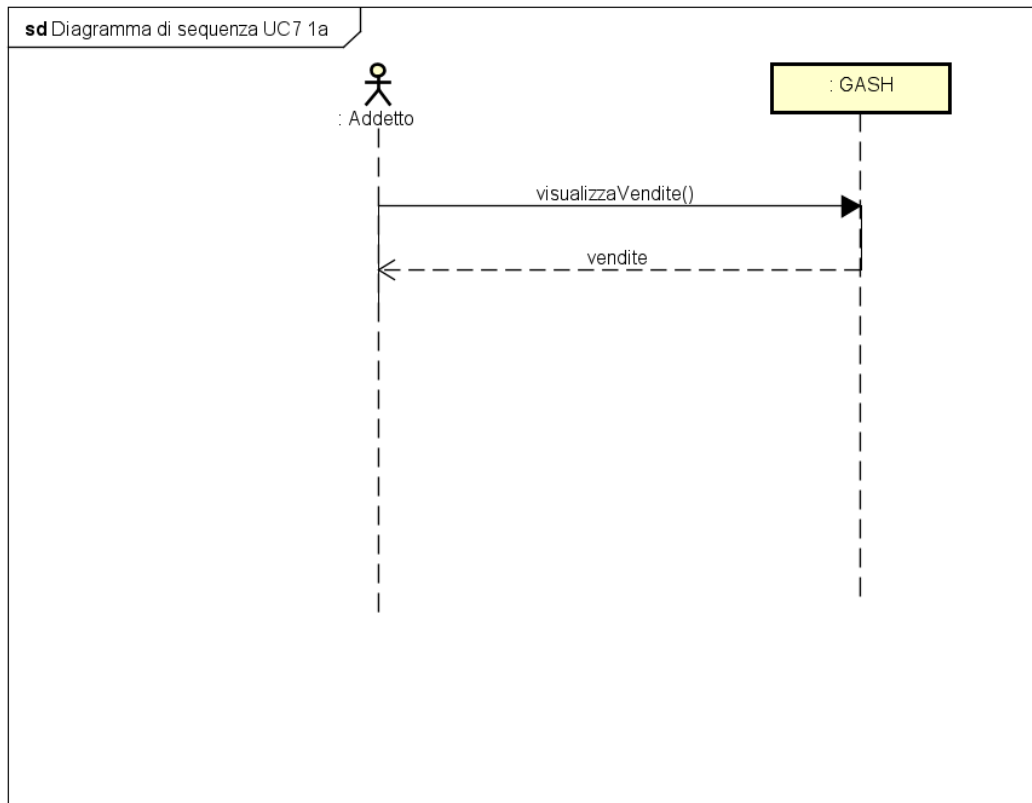
In quest’iterazione del caso d’uso UC7 vengono analizzati lo scenario principale e le estensioni 1a e 1d. Lo sviluppo di questi scenari non comporta l’aggiunta di nuove classi al modello di dominio.

#### 4.1.5 Caso d’uso UC7, Diagramma di sequenza di sistema

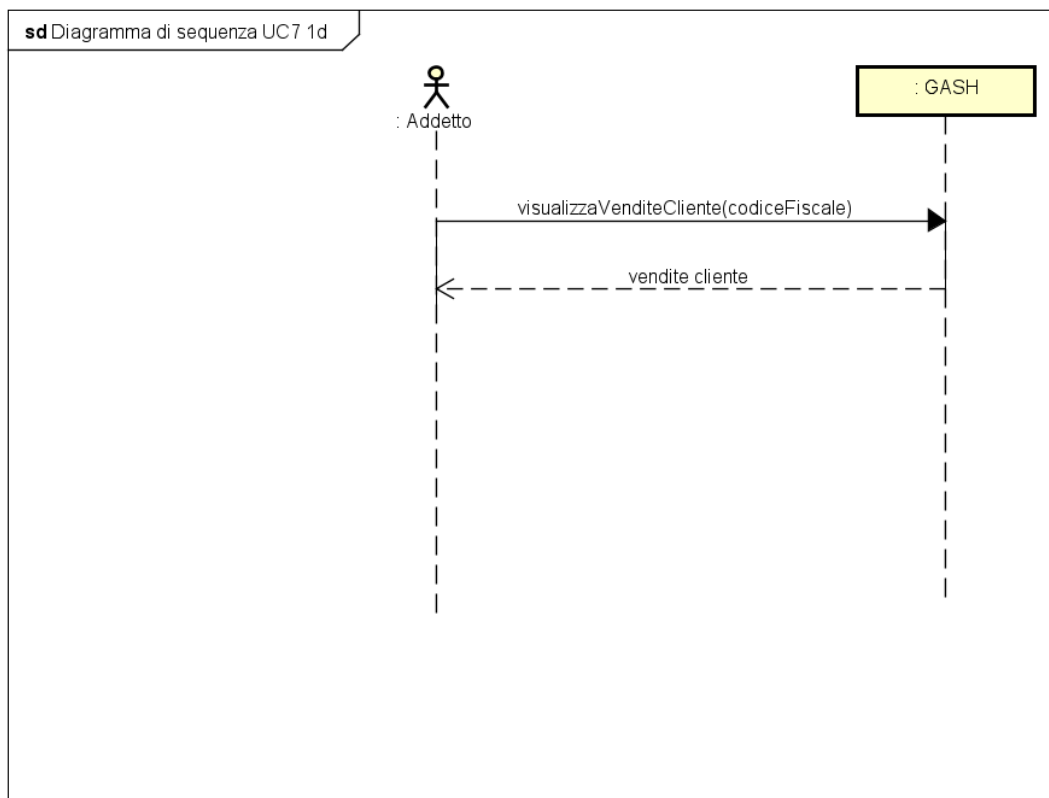
##### Scenario principale di successo



## Estensione 1a



## Estensione 1d



#### 4.1.6 Caso d'uso UC10: Gestisci Componente, Modello di dominio

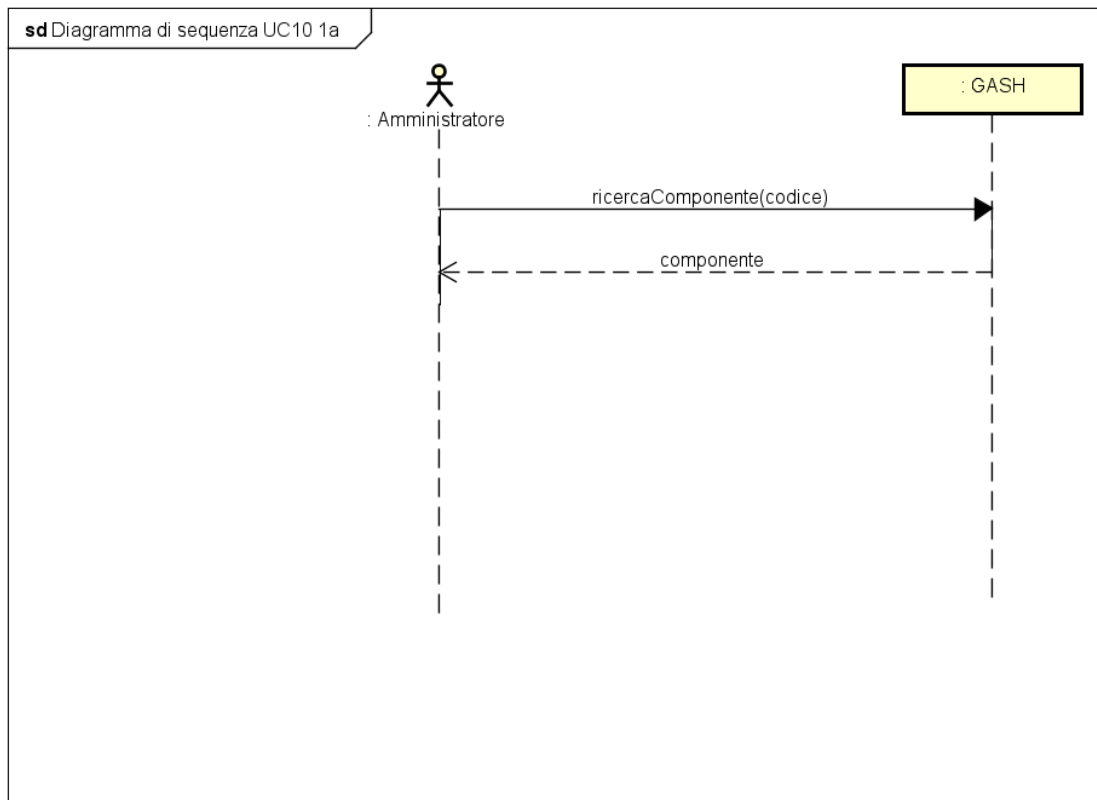
In quest'iterazione del caso d'uso UC10 vengono analizzati lo scenario principale e le estensioni 1a, 1b e 1c. Lo sviluppo di questi scenari non comporta l'aggiunta di nuove classi al modello di dominio.

#### 4.1.7 Caso d'uso UC10, Diagramma di sequenza di sistema

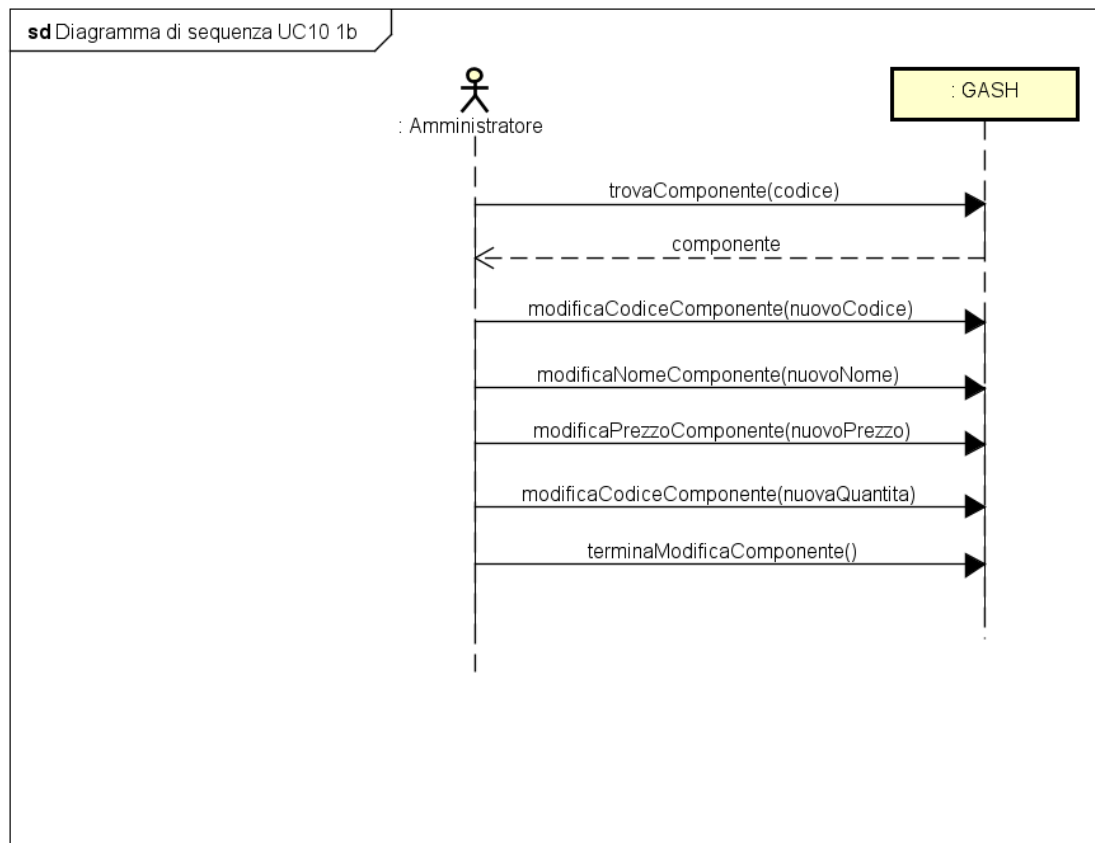
##### Scenario principale di successo (inserimento)

Si è deciso di non implementare lo scenario principale di successo di questo caso d'uso, in quanto l'inserimento di un componente avviene già in modo automatico al verificarsi del caso d'uso UC12 (Arrivo Ordine), in seguito all'ordine di uno o più componenti da un fornitore.

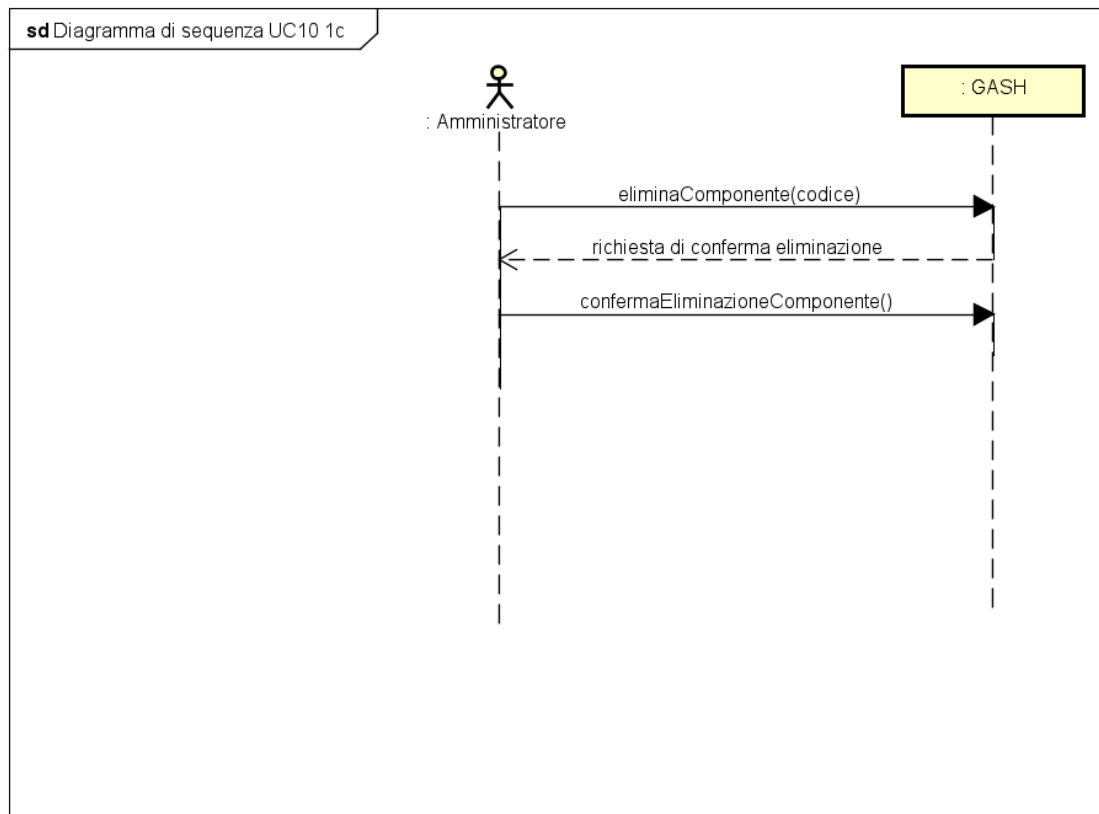
##### Estensione 1a (ricerca)



##### Estensione 1b (modifica)



### Estensione 1c (eliminazione)





## 4.2 Iterazione 4, Progettazione

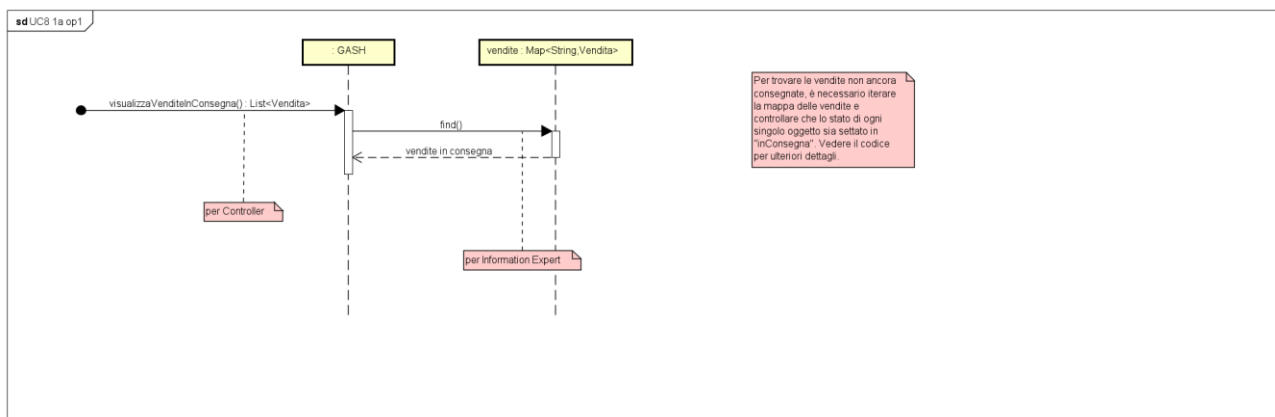
### 4.2.1 Introduzione

In quest'ultima iterazione, in aggiunta all'implementazione dei casi d'uso precedentemente menzionati, viene applicata la regola di dominio R3, che permette ad ogni cliente "abituale" (cioè ogni cliente che abbia effettuato almeno 10 vendite) di ottenere uno sconto del 10% su ogni vendita (vedere il codice per ulteriori dettagli).

### 4.2.2 Caso d'uso UC8, Diagrammi di interazione

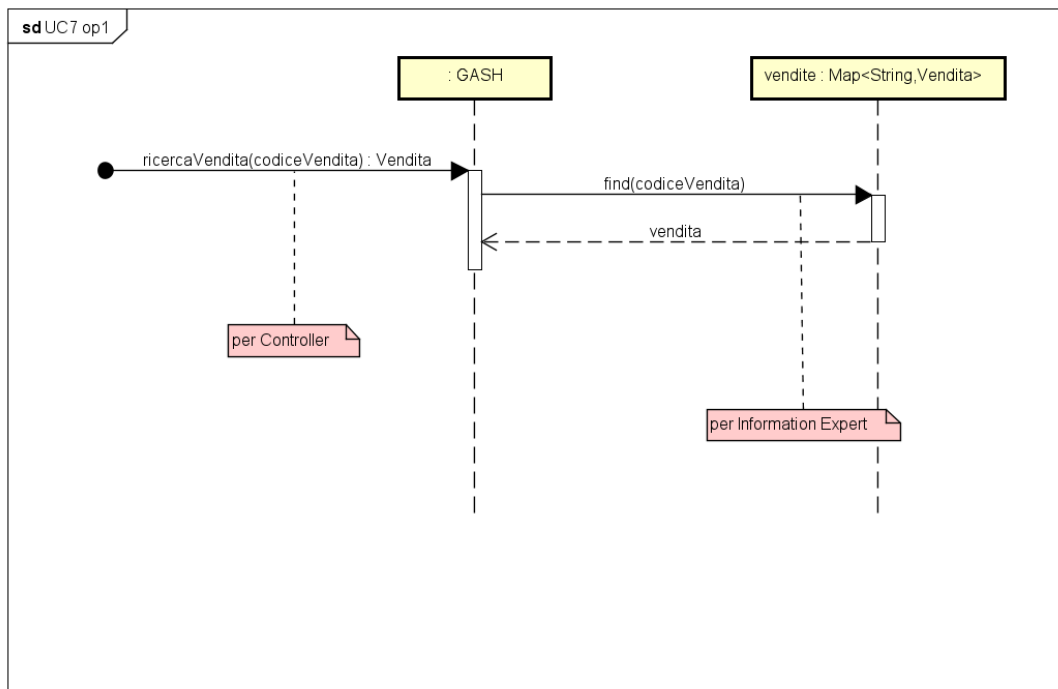
Per tracciare lo stato delle consegne, si è deciso di applicare il pattern State e più nello specifico creare una classe StatoVendita, collegata alla classe VenditaPorte tramite l'associazione "si-trova-in", a cui sono associati gli stati "InConsegna" e "Consegnato". Non appena viene effettuata una nuova vendita il suo stato viene impostato in "InConsegna", quando le porte per garage verranno consegnate al cliente esso sarà modificato in "Consegnato".

#### 4.2.2.1 visualizzaVenditeInConsegna()



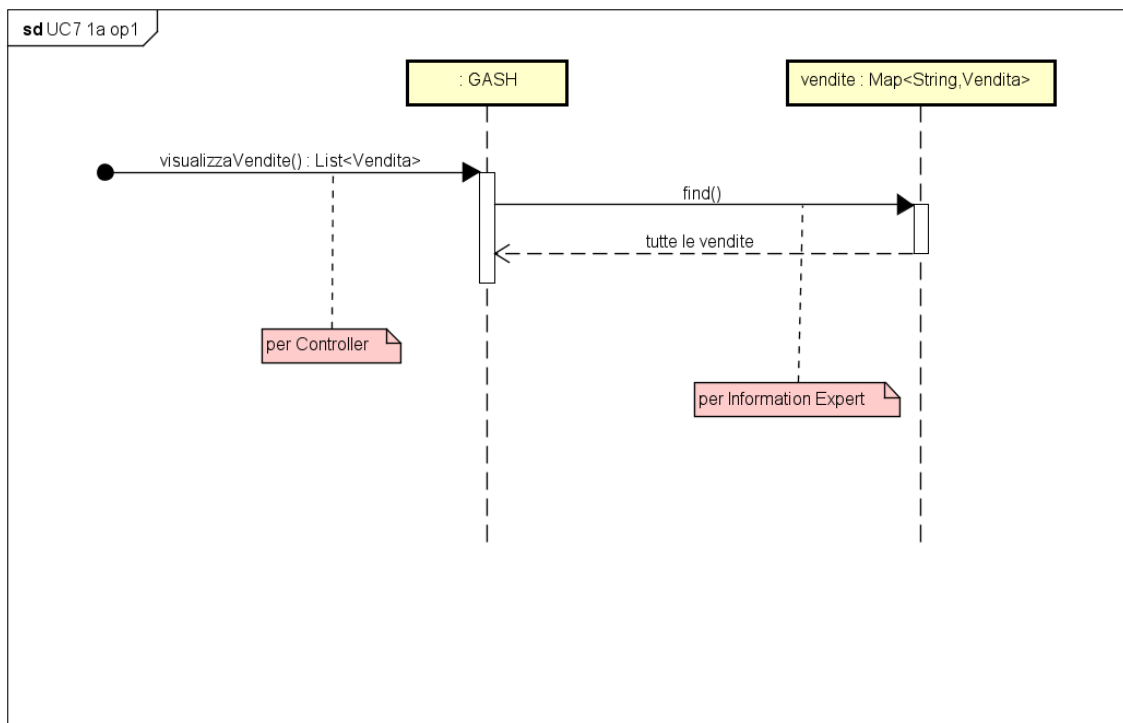
### 4.2.3 Caso d'uso UC7 (scenario principale di successo), Diagrammi di interazione

#### 4.2.3.1 ricercaVendita(codiceVendita:String)



## 4.2.4 Caso d'uso UC7 (estensione 1a), Diagrammi di interazione

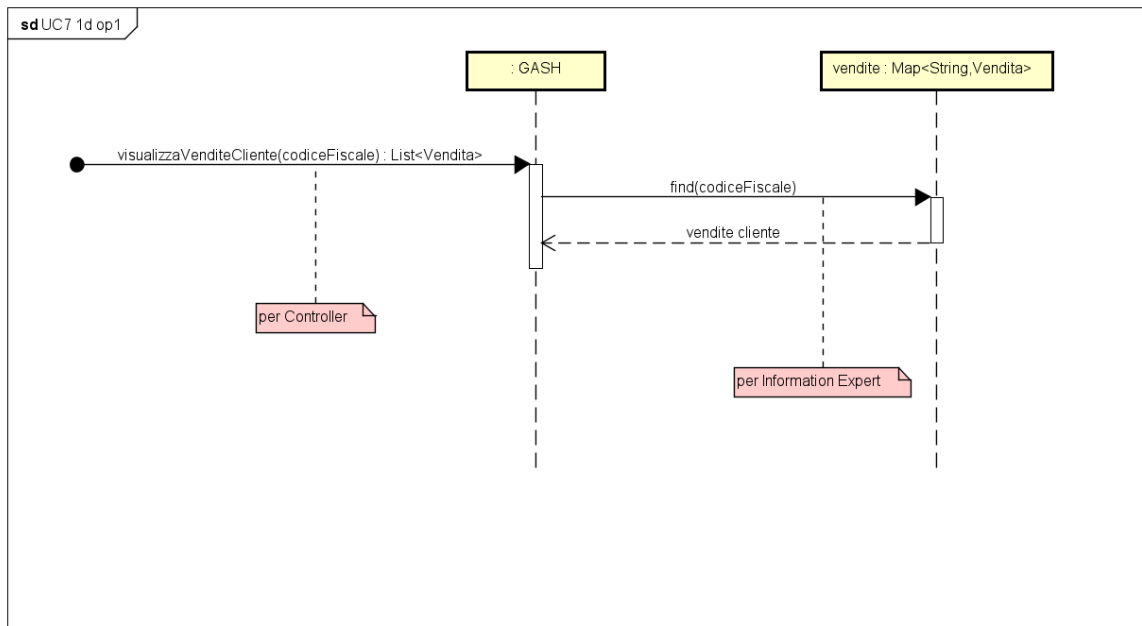
### 4.2.4.1 visualizzaVendite()





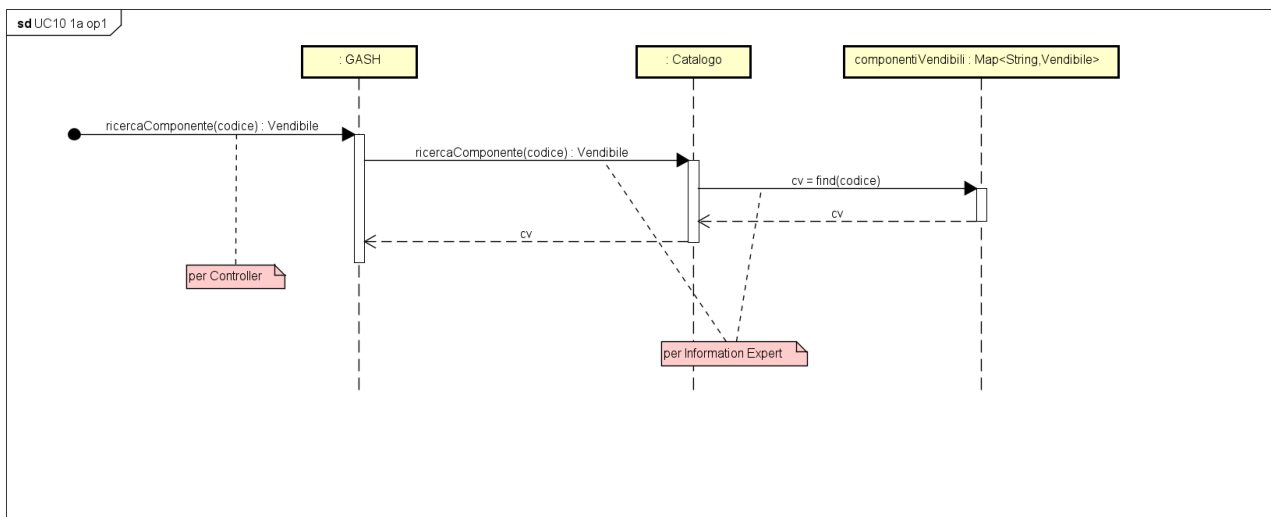
## 4.2.5 Caso d'uso UC7 (estensione 1d), Diagrammi di interazione

### 4.2.5.1 visualizzaVenditeCliente(codiceFiscale:String)



## 4.2.6 Caso d'uso UC10 (estensione 1a - ricerca), Diagrammi di interazione

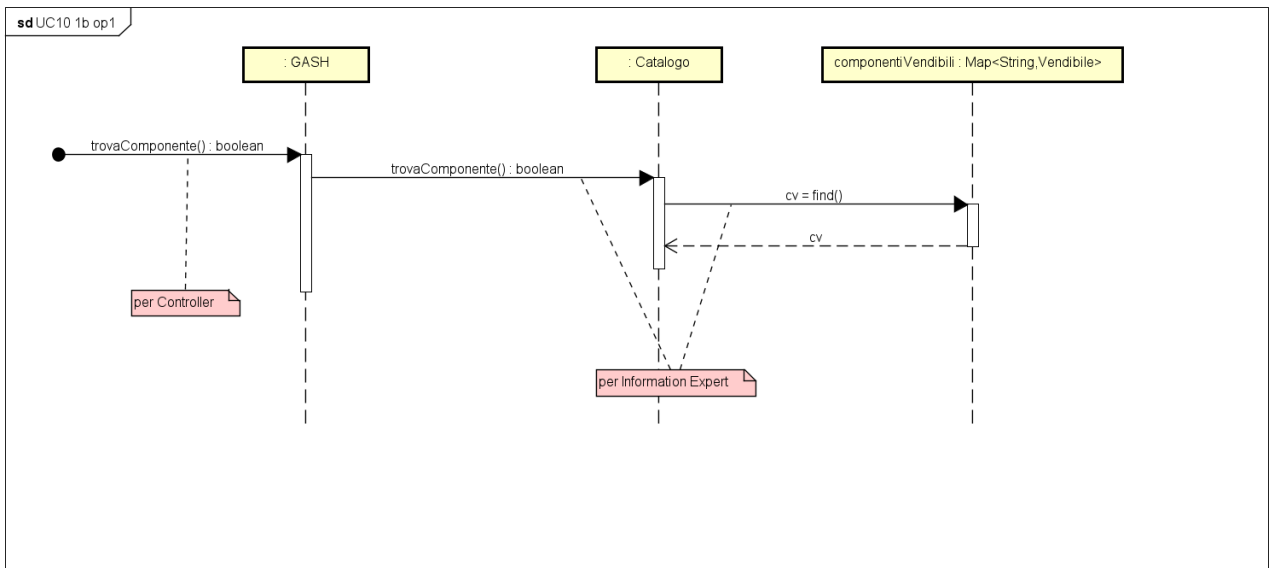
### 4.2.6.1 ricercaComponente(codice:String)



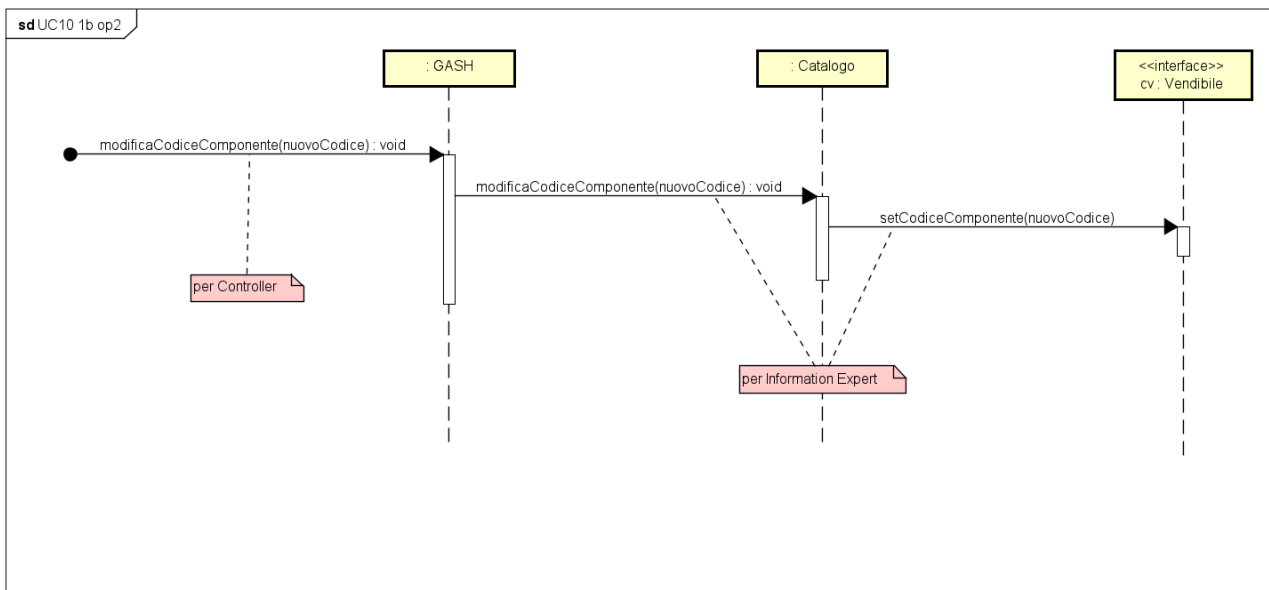
## 4.2.7 Caso d'uso UC10 (estensione 1b - modifica), Diagrammi di interazione

Poiché entrambi i tipi di componenti, ComponenteBase e Lamiera, condividono gli stessi attributi (che si intende modificare), è stato rappresentato un unico insieme di diagrammi, indicando una generica istanza con la sigla cv (che sta per “componente vendibile”, dal nome dell’interfaccia Vendibile).

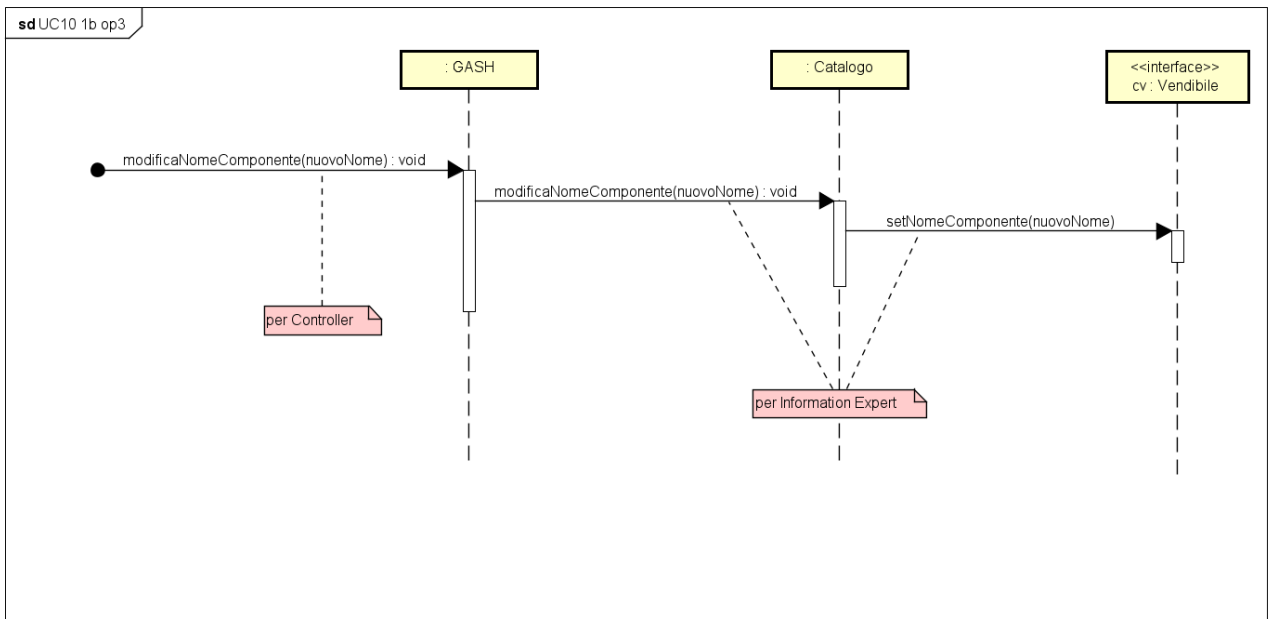
### 4.2.7.1 trovaComponente(codice:String)



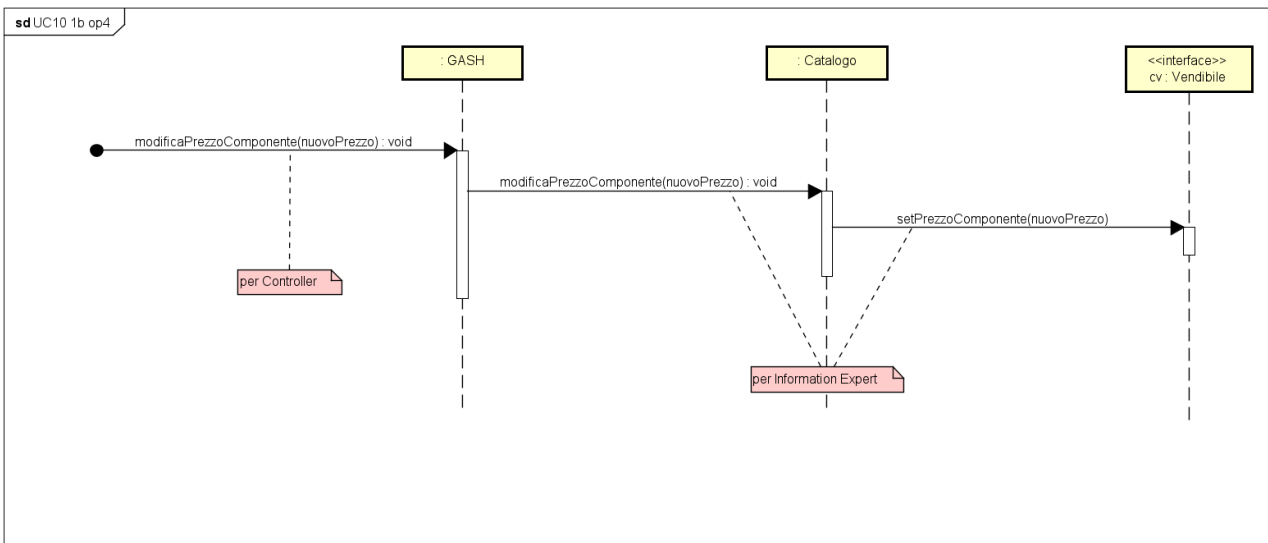
### 4.2.7.2 modificaCodiceComponente(nuovoCodice:String)



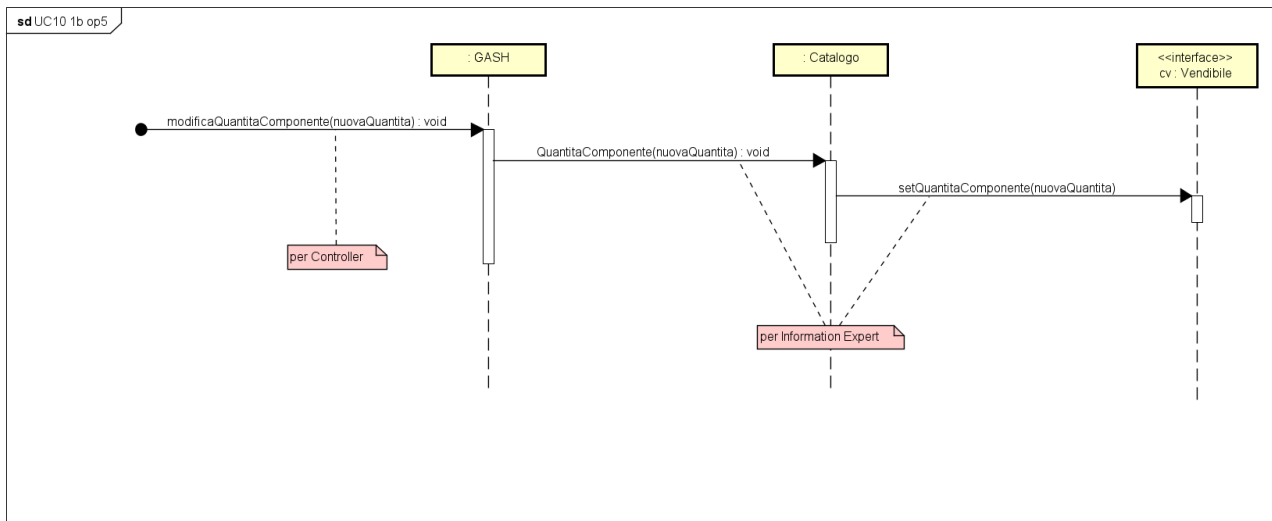
#### 4.2.7.3 modificaNomeComponente (nuovoNome:String)



#### 4.2.7.4 modificaPrezzoComponente(nuovoPrezzo:Double)

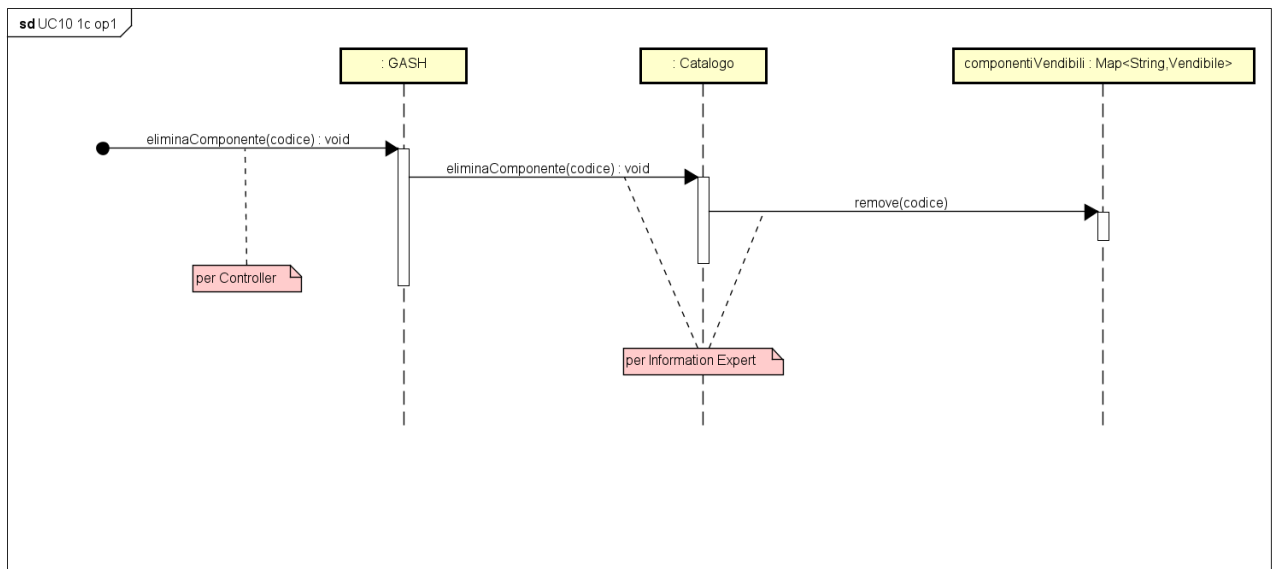


#### 4.2.7.5 modificaQuantitaComponente(nuovaQuantita:String)

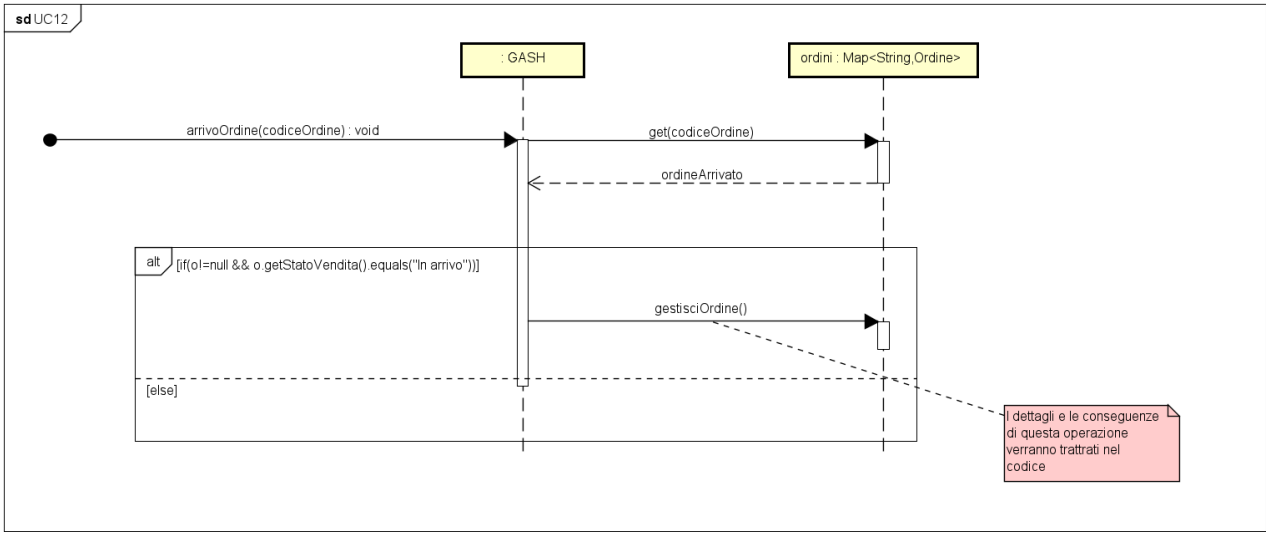


#### 4.2.8 Caso d'uso UC10 (estensione 1c - eliminazione), Diagrammi di interazione

##### 4.2.8.1 eliminaComponente(codice:String)



#### 4.2.9 Caso d'uso UC12, Diagrammi di interazione



#### 4.2.10 Diagramma delle classi di progetto

