**Ingegneria del Software**

**Corso di laurea Magistrale in Ingegneria Informatica**

**A.A. 2021-2022**



**Riccardo Simone Castorina**

**Philip També**

Sommario

[Introduzione 2](#_Toc1717073764)

[Analisi 3](#_Toc114933457)

[1. Modelli di Dominio 4](#_Toc238399788)

[1.1 Modello di Dominio di UC5, UC8 e UC11: 4](#_Toc1718302760)

[1.2 Diagramma di Oggetti di Dominio 4](#_Toc795902049)

[2. Diagrammi di Sequenza di Sistema (SSD) 5](#_Toc1272329330)

[2.1 Diagramma di Sequenza di Sistema di UC5: 5](#_Toc1201618350)

[2.2 Diagramma di Sequenza di Sistema di UC8: 5](#_Toc129676590)

[2.3 Diagramma di Sequenza di Sistema di UC11: 5](#_Toc1386424804)

[3. Contratti delle operazioni 5](#_Toc1520830181)

[Contratto CO2: scegliLuogoRitiro 5](#_Toc907574350)

[Contratto CO3: scegliPagamento 5](#_Toc1154192168)

[Contratto CO5: scegliVeicoloNoleggio 5](#_Toc1773861485)

[Contratto CO7: effettuaPagamentoNoleggio 5](#_Toc396407604)

[Contratto CO8: aggiungiOptional 5](#_Toc1394144099)

[Progettazione 5](#_Toc665604045)

[1. Caso d’uso 5 - Diagrammi di Sequenza 5](#_Toc707716579)

[1.1 – mostraAcquista() 5](#_Toc1734004928)

[1.2 – filtraVeicoliAcquisto(produttore : String, modello : String, tipoVeicolo : String) 5](#_Toc58141381)

[1.3 – scegliVeicoloAcquisto(codice : int, parco : Parco) 5](#_Toc1195766596)

[1.4 – personalizzaVeicolo(veicoloPersonalizzato : VeicoloPersonalizzato) 5](#_Toc259260436)

[1.5 – scegliLuogoRitiro(luogoRitiro : String) 5](#_Toc897384590)

[1.6 – scegliPagamento(codicePagamento : int) 5](#_Toc238388996)

[1.7 – impostaOrdine(metodoPagamento : MetodoPagamento) 5](#_Toc1328938460)

[1.8 – effettuaPagamentoAcquisto(prezzoFinale : float) 5](#_Toc1785664713)

[2. Caso d’Uso 5 – Diagramma delle Classi 5](#_Toc226914425)

[3. Caso d’Uso 8 – Diagrammi di Sequenza 5](#_Toc2084047713)

[3.1 mostraNoleggia() 5](#_Toc746661796)

[3.2 – filtraVeicoliNoleggio(produttore : String, modello : String, tipoVeicolo : String) 5](#_Toc451918230)

[3.3 – scegliVeicoliNoleggio(veicoloNoleggiabile : VeicoloNoleggiabile) 5](#_Toc1114611912)

[3.4 – scegliPagamento(codicePagamento : int) 5](#_Toc948503317)

[3.5 – impostaOrdine(metodoPagamento : MetodoPagamento) 5](#_Toc375170478)

[3.6 – effettuaPagamentoNoleggio(prezzoFinale : float) 5](#_Toc2036405382)

[4. Caso d’Uso 8 – Diagramma delle Classi 5](#_Toc924888163)

[5. Caso d’Uso 11 – Diagrammi di Sequenza 5](#_Toc1535529967)

[5.1 – mostraDescrizioneOptional(descrizioneOptional : DescrizioneOptional) 5](#_Toc1871002318)

[5.2 – aggiungiOptional(nomeOptional : String) 5](#_Toc674675436)

[5.3 – terminaPersonalizzazione(veicoloPersonalizzato : VeicoloPersonalizzato) 5](#_Toc1366613403)

[6. Caso d’Uso 8 – Diagramma delle Classi 5](#_Toc961273017)

Introduzione

Dai test effettuati, ci siamo resi conto che A4R ha decisamente troppe responsabilità e riteniamo più sensato delegarne alcune.  
A questo fine abbiamo rimodellato lo scopo dell’entità “Parco”, adesso unica e non più considerata come un contenitore di veicoli appartenenti ad un solo utente concessionario: ciò ne aumenterà indubbiamente la *coesione* e diminuirà l’*accoppiamento* dell’entità A4R.  
Peraltro, è stato ritenuto opportuno considerarlo un “Singleton”.

L’entità “email” è stata rimossa: sarà il sistema stesso a occuparsi del loro invio quando necessario ed essendo casi estremamente semplici, non verranno ulteriormente trattati.

La classe “Ordine” ci è sembrata poco coesa e ciò ci ha spinto ad effettuare altre modifiche: adesso è una classe astratta implementata tramite un pattern *GRASP* “Polimorph”.  
Alcune delle sue operazioni, adesso sono svolte dalle entità figlie “Acquisto” e “Noleggio”.

Similmente, “MetodoPagamento” è stato modificato in *interfaccia*, in modo da proteggere il sistema da eventuali modifiche dei metodi di pagamento reali esterni e forniti da terzi, in concordanza con i princìpi di “Polimorph”, “Protected Variations” e “Indirection”. Il *pattern GoF* che abbiamo scelto per implementarlo è l’ “Adapter”.

Inoltre, è stato implementato il caso d’uso UC11 (precedentemente rimandato) e abbiamo deciso di trattare lo *scenario alternativo* identico per i casi d’uso *UC5* e *UC8*, identificato rispettivamente come *10a* e *11a*  
Ne riportiamo sotto il testo affinché possa servire da riferimento rapido:

**10a (UC5) /11a (UC8)**. Il servizio di pagamento scelto non è disponibile:  
 1) Il sistema notifica all’utente il fallimento della transazione di pagamento.  
 2a) L’utente sceglie un altro metodo di pagamento.  
 3) Il sistema riprende dal passo *8 (UC**5)/9 (UC8)* dello *scenario principale di successo*.

Analisi

Il *refactoring* ha conseguito delle modifiche sostanziali al *modello di dominio* e ad alcuni *contratti delle operazioni*. Mentre, le modifiche apportate ai *Diagrammi di Sequenza di Sistema* sono piuttosto lievi.  
Lo *scenario alternativo* non ha influito in alcun modo sull’analisi di questa iterazione.

## 1. Modelli di Dominio

### 1.1 Modello di Dominio di UC5, UC8 e UC11:

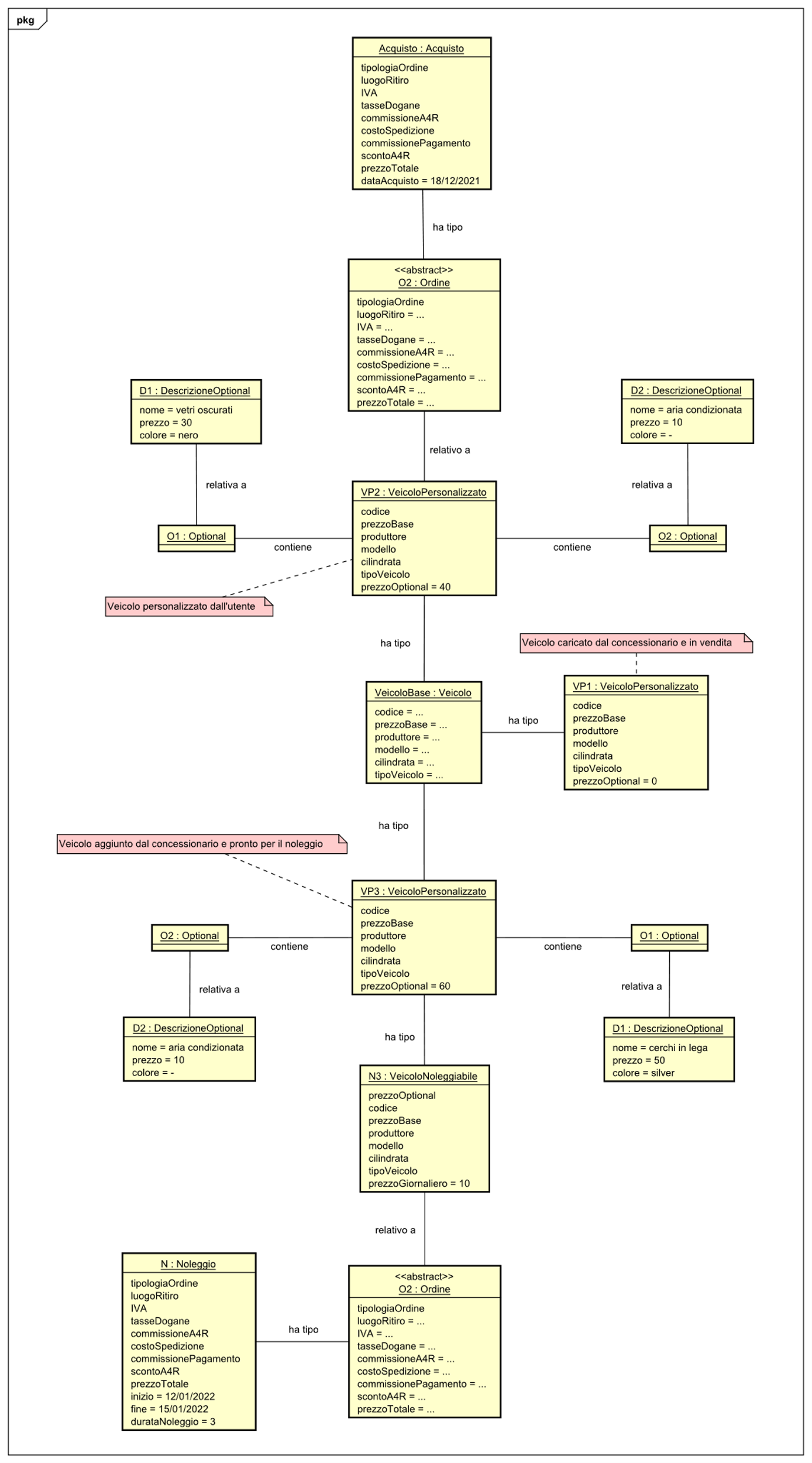
Di seguito proponiamo il *Modello di Dominio* rivisto in seguito al *refactoring* effettuato.

**

*Figura 1.1 – Modello di Dominio dei casi d’uso*

### 1.2 Diagramma di Oggetti di Dominio

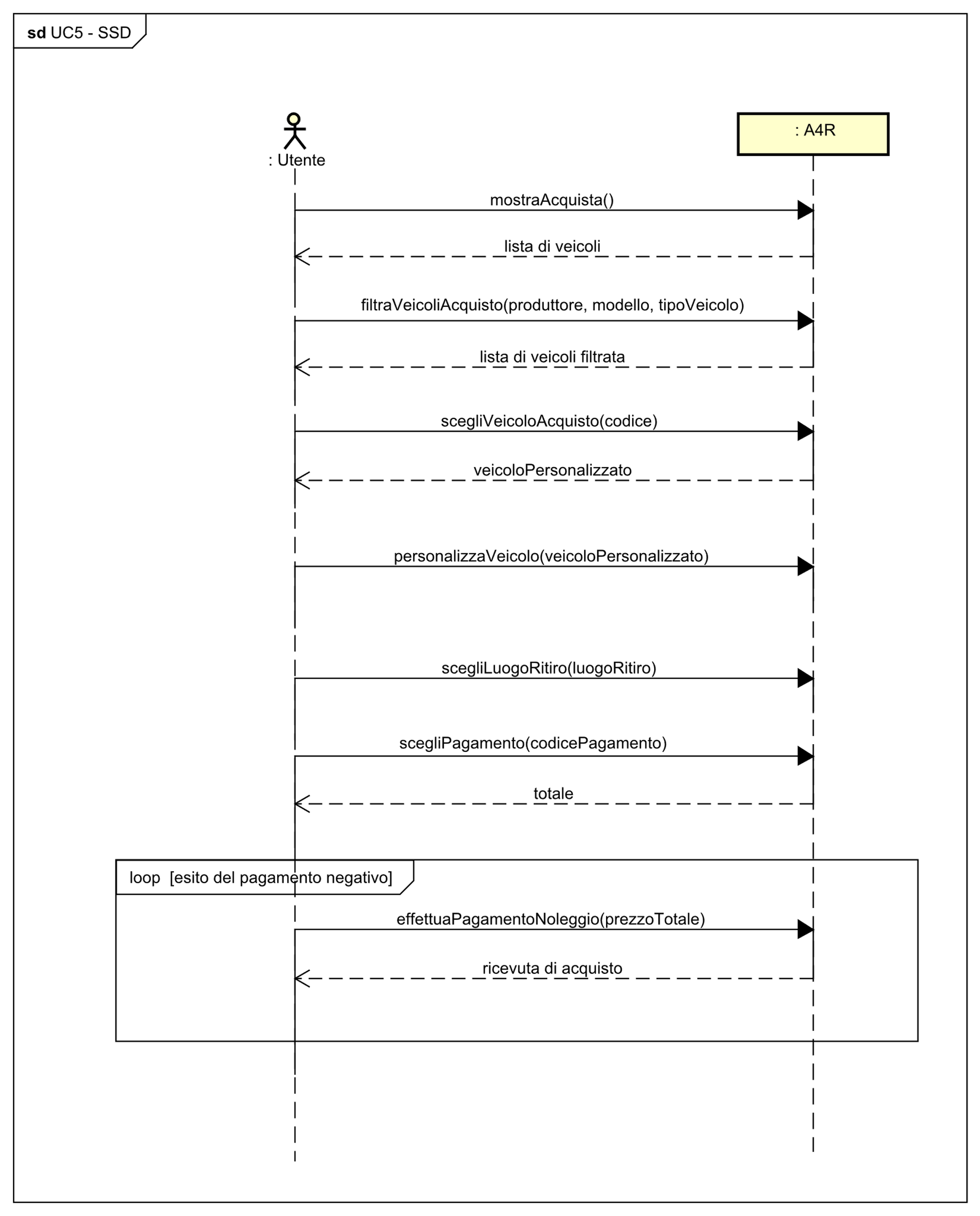
Riguardo entrambi i casi d’uso, si è reso necessario chiarire le relazioni che intercorrono tra gli oggetti del dominio. Di seguito ne è proposto un esempio di *Diagramma* (anch’esso in formato SVG) prodotto allo scopo di fare luce sull’argomento.

  
*Figura 1.2 – Diagramma di Oggetti di Dominio pertinente ai casi d’uso UC5, UC8 e UC11*

## 2. Diagrammi di Sequenza di Sistema (SSD)

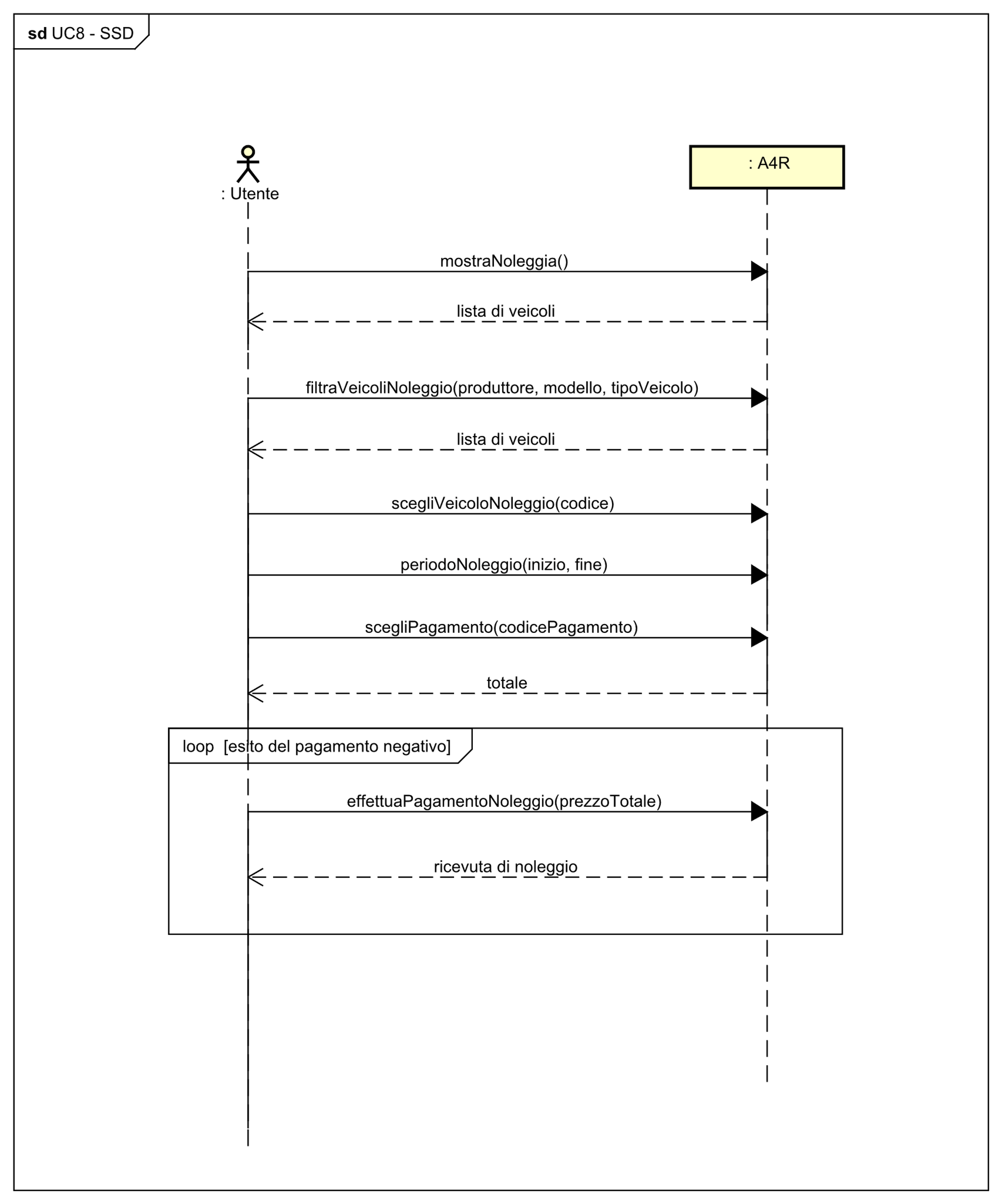
Illustriamo (in formato SVG) i *Diagrammi di Sequenza di Sistema* poiché sono state raffinate le operazioni della precedente operazione ed è aggiunto anche uno scenario alternativo in entrambi gli SSD.

### 2.1 Diagramma di Sequenza di Sistema di UC5:



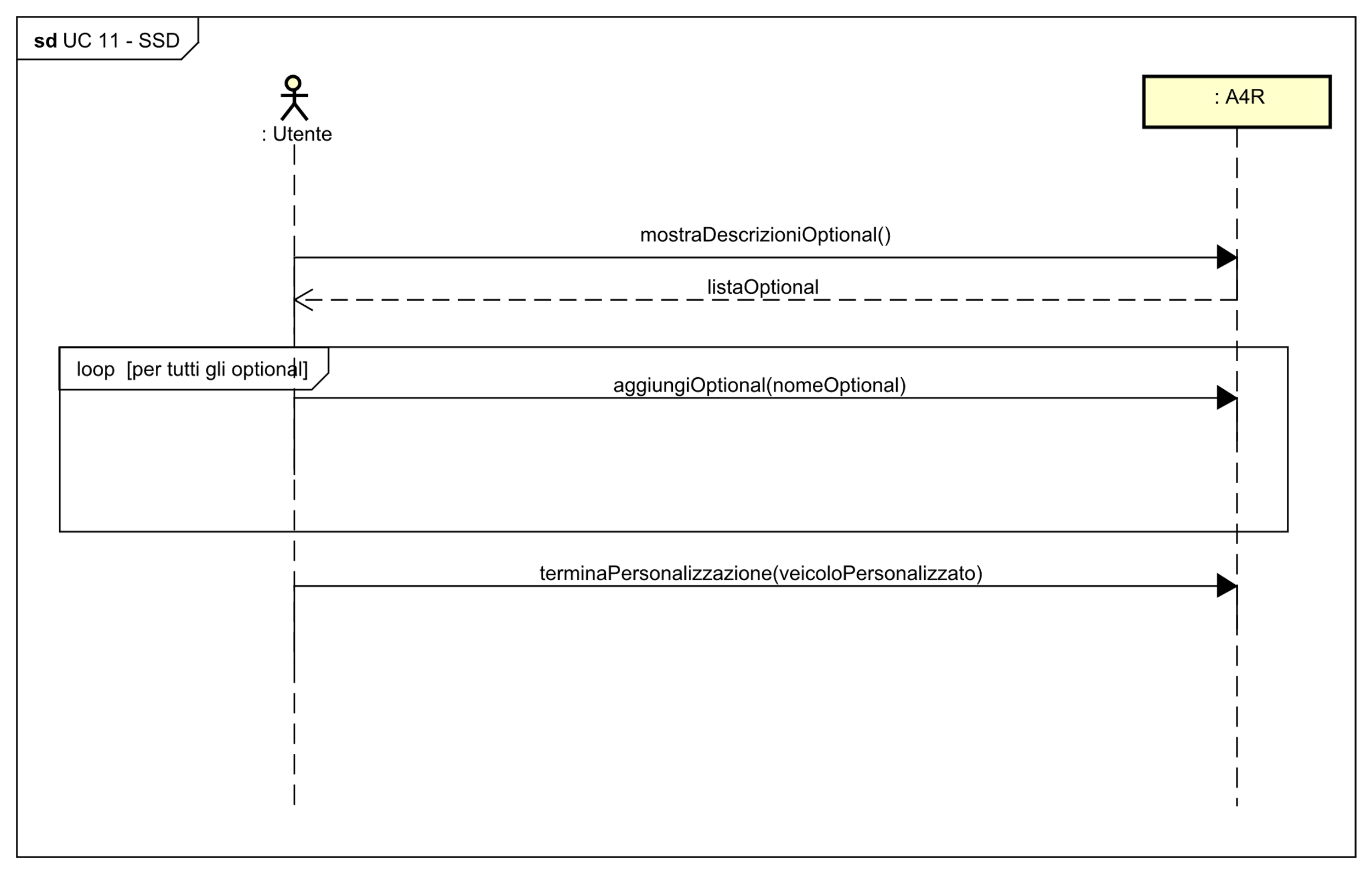
*Figura 2.1 – Diagramma di Sequenza di Sistema del caso d’uso UC5*

### 2.2 Diagramma di Sequenza di Sistema di UC8:



*Figura 2.2 – Diagramma di Sequenza di Sistema del caso d’uso UC8*

### 2.3 Diagramma di Sequenza di Sistema di UC11:



*Figura 2.2 – Diagramma di Sequenza di Sistema del caso d’uso UC11*

## 3. Contratti delle operazioni

Adesso, “Acquisto” e “Noleggio” sono generati rispettivamente in *scegliVeicoloAcquisto* e *scegliVeicoloNoleggio*. Presentiamo solo i contratti che hanno subìto dei cambiamenti.

**Contratto CO1: scegliVeicoloAcquisto**  
*Operazione*: **scegliVeicoloAcquisto(codice : int)**

*Riferimenti*: UC5: Acquisto di un veicolo con prezzo fisso

*Pre-condizioni*: Il sistema conosce l’utente U che ha scelto l’attività “Acquista veicolo” e il concessionario C che possiede il veicolo V all’interno del parco del sistema.

*Post-condizioni*:

* Viene generata un’istanza di VeicoloPersonalizzato VP
* Viene generata l’istanza ricevutaAcquisto di Acquisto
* Gli attributi di ricevutaAcquisto e di VP sono inizializzati
* VP viene associato al parco tramite “corrente”
* U viene associato ad Acquisto tramite “effettua”
* ricevutaAcquisto viene associato a VP tramite “può avere”
* ricevutaAcquisto viene associato al sistema A4R tramite “corrente”

### Contratto CO2: scegliLuogoRitiro

*Operazione*: **scegliLuogoRitiro(luogoRitiro : String)**

*Riferimenti*: UC5: Acquisto di un veicolo con prezzo fisso

*Pre-condizioni*: l’utente U ha finito di personalizzare il veicolo VP. L’istanza “ricevutaAcquisto” di Acquisto è già stata generata.

*Post-condizioni*:

* ricevutaAcquisto.luogoRitiro viene aggiornato
* ricevutaAcquisto.costoSpedizione viene aggiornato
* ricevutaAcquisto.tasseDogane viene aggiornato

### Contratto CO3: scegliPagamento

Questa operazione sfrutta la generalizzazione delle classi per adattarsi sia all’ UC5 nel caso si stia effettuando il pagamento di un acquisto, sia all’ UC8 nel caso di un noleggio. Nello specifico accadrà che l’entità Veicolo sarà un VeicoloPersonalizzato nel caso di un acquisto e un VeicoloNoleggiabile nel caso di un noleggio. Analogamente a Veicolo, anche Ordine sfrutterà la generalizzazione consentendo l’uso di un’entità Acquisto o eventualmente di un'entità Noleggio.

*Operazione*: **scegliPagamento(codice : int) <<abstract>>**

*Riferimenti*:

* UC5: Acquisto di un veicolo con prezzo fisso
* UC8: Noleggia veicolo

*Pre-condizioni*: Il sistema conosce l’utente U e il concessionario C che possiede il veicolo V all’interno del parco del sistema.

*Post-condizioni*:

* ricevutaAcquisto/ricevutaNoleggio.scontoPremium viene aggiornato
* ricevutaAcquisto/ricevutaNoleggio.prezzoBase viene aggiornato
* ricevutaAcquisto/ricevutaNoleggio.scontoConcessionario viene aggiornato
* ricevutaAcquisto/ricevutaNoleggio.commissionePagamento viene aggiornato

**Contratto CO4: effettuaPagamentoAcquisto**  
*Operazione*: **effettuaPagamentoAcquisto(prezzoFinale : float)**

*Riferimenti*: UC5: Acquisto di un veicolo con prezzo fisso

*Pre-condizioni*: l’utente U ha confermato di voler continuare con l’acquisto del veicoloPersonalizzato VP, ha scelto il luogo di ritiro L e il metodo di pagamento MP con cui effettuarlo. Al passo precedente è stata scelta la modalità di pagamento ed è stato calcolato il prezzo totale. È stata già generata l’istanza di ricevutaAcquisto di Acquisto.

*Post-condizioni*:

* A4R viene associato a ricevutaAcquisto tramite “eroga”
* A4R viene associato a MetodoPagamento MP tramite “usa”
* Viene eliminata l’associazione “corrente” tra A4R e ricevutaAcquisto.
* ricevutaAcquisto.dataAcquisto viene aggiornato.

### Contratto CO5: scegliVeicoloNoleggio

*Operazione*: **scegliVeicoloNoleggio (codice : int)**

*Riferimenti*: UC8: Noleggia veicolo

*Pre-condizioni*: Il sistema conosce l’utente U che ha scelto l’attività “Noleggia veicolo” e il veicolo di cui desidera analizzare le caratteristiche. Conosce anche il concessionario C che possiede il veicolo noleggiabile VN. Nel noleggio dell’auto esiste un'unica modalità di ritiro ovvero il ritiro presso il concessionario, a differenza dell’acquisto in cui si può scegliere tra diverse modalità.

*Post-condizioni*:

* Viene generata un’istanza ricevutaNoleggio di Noleggio.
* ricevutaNoleggio viene associato a U tramite “effettua”.
* ricevutaNoleggio viene associato ad A4R tramite “corrente”.
* Gli attributi di ricevutaNoleggio sono inizializzati.
* ricevutaNoleggio viene associato al veicolo VN tramite “può avere”
* ricevutaNoleggio.luogoRitiro diventa C.luogo

### Contratto CO7: effettuaPagamentoNoleggio

*Operazione*: **effettuaPagamentoNoleggio(prezzoTotale : float)**

*Riferimenti*: UC8: Noleggia Veicolo

*Pre-condizioni*: l’utente U ha scelto il veicoloNoleggiabile VN di suo interesse, ha scelto come ritirare il veicolo, il metodo di pagamento MP che preferisce usare e le date di inizio e fine noleggio.  
Il concessionario ha già caricato il veicolo oggetto del noleggio che è un “VeicoloNoleggiabile” derivato dalla specializzazione di “VeicoloPersonalizzato”. Al passo precedente è stato calcolato il prezzo totale per poter effettuare il pagamento.

*Post-condizioni*:

* Viene associato il VeicoloNoleggiabile VN al noleggio “ricevutaNoleggio” tramite l’associazione “attualmente in”
* A4R viene associato a MP tramite “usa”
* A4R viene associato a ricevutaNoleggio tramite “eroga”
* Viene eliminata l’associazione “corrente” tra A4R e ricevutaNoleggio

### Contratto CO8: aggiungiOptional

*Operazione*: **aggiungiOptional(nomeOptional : String)**

*Riferimenti*: UC5: Acquisto di un veicolo con prezzo fisso

*Pre-condizioni*: l’utente U ha scelto il veicolo V da personalizzare, a seguito della scelta è stata creata un’istanza di VeicoloPersonalizzato VP al quale si aggiungeranno gli Optional presi dalla lista degli optional definiti in DescrizioneOptional che sono gli unici aggiungibili al dato Veicolo. L'utente ha scelto di aggiungere un optional al veicolo personalizzato VP. L’istanza DO di DescrizioneOptional è già presente poiché generata durante il caricamento del mezzo nella piattaforma.

*Post-condizioni*:

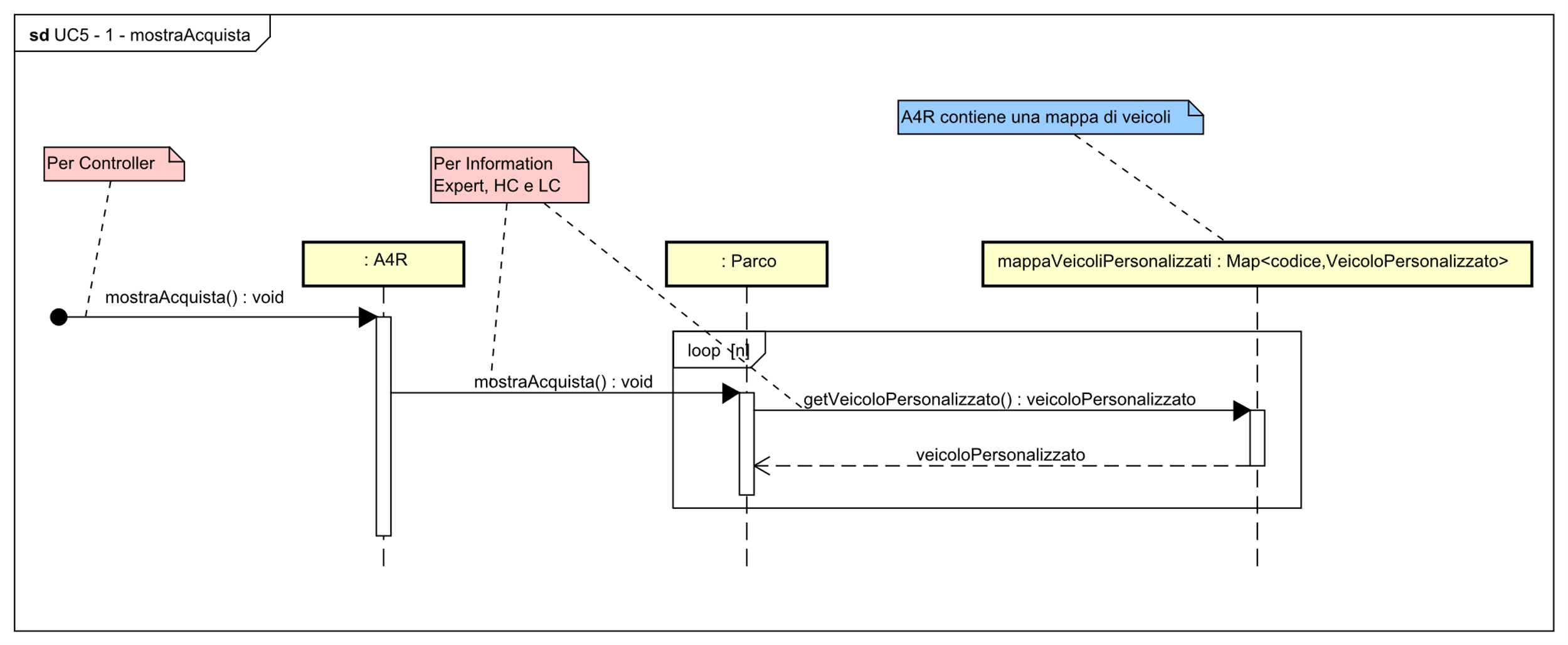
* Viene generata un’istanza di Optional OP
* VP viene associato a OP attraverso “contiene”
* OP viene associato a DO attraverso “ha”

Progettazione

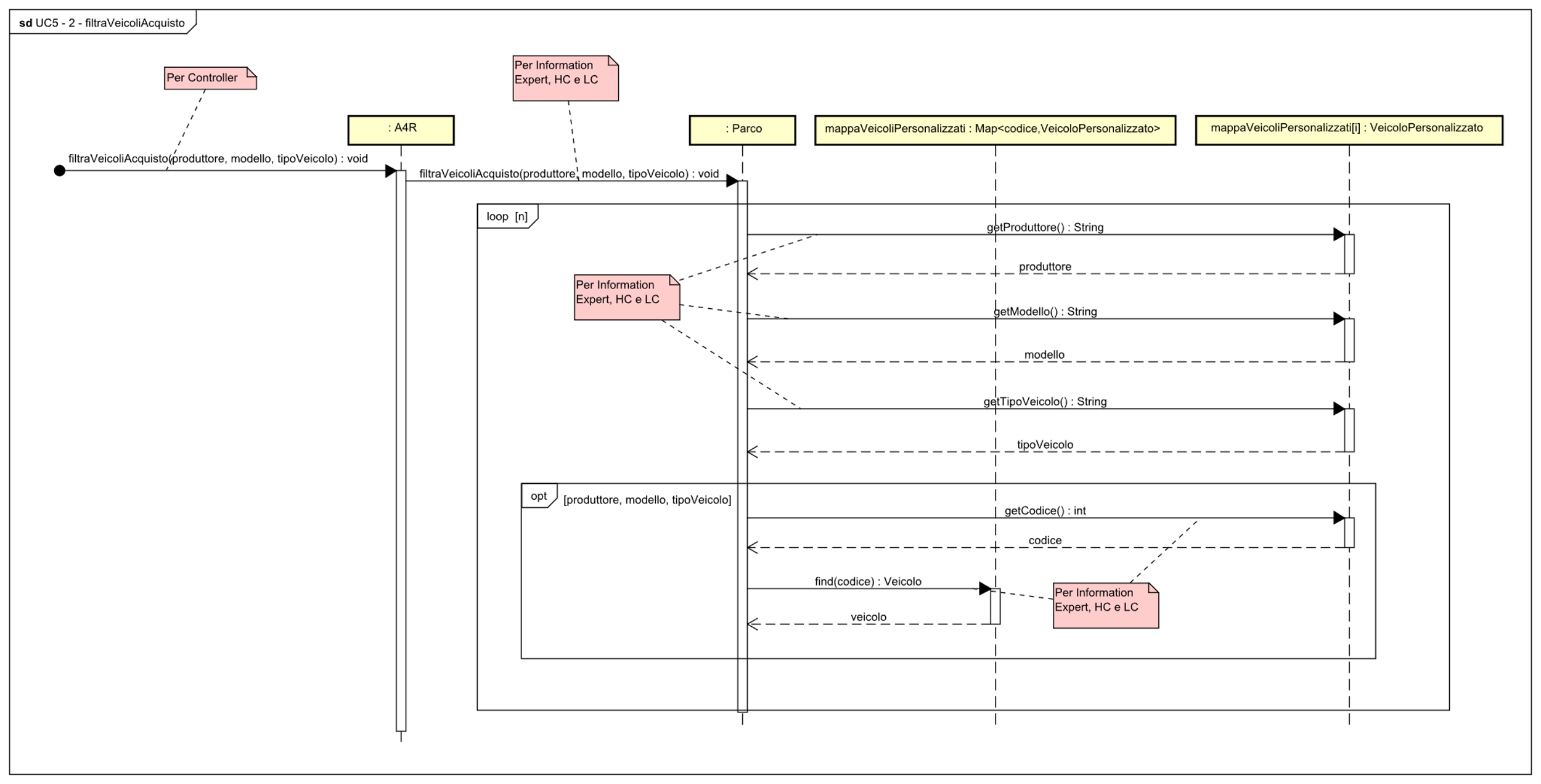
Costruiamo il nuovo *Modello di Progetto*, mostrando solo i Diagrammi di Interazione che hanno subìto modifiche e quelli totalmente nuovi.

## 1. Caso d’uso 5 - Diagrammi di Sequenza

### 1.1 – mostraAcquista()

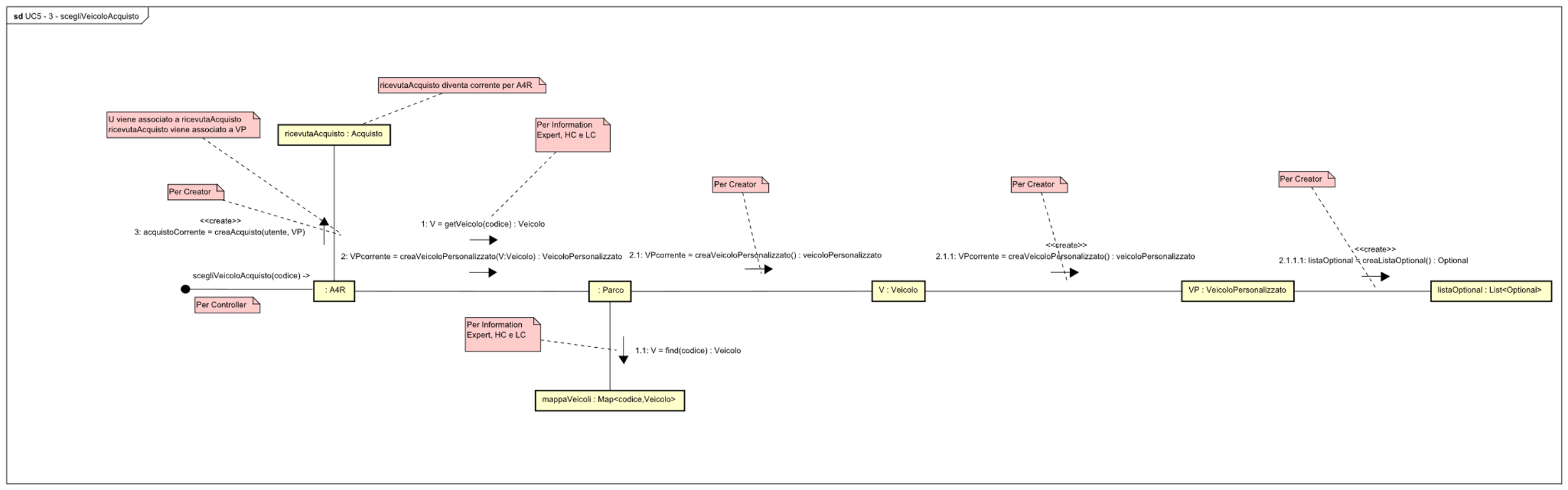
*Figura 3.1*

### 1.2 – filtraVeicoliAcquisto(produttore : String, modello : String, tipoVeicolo : String)

**

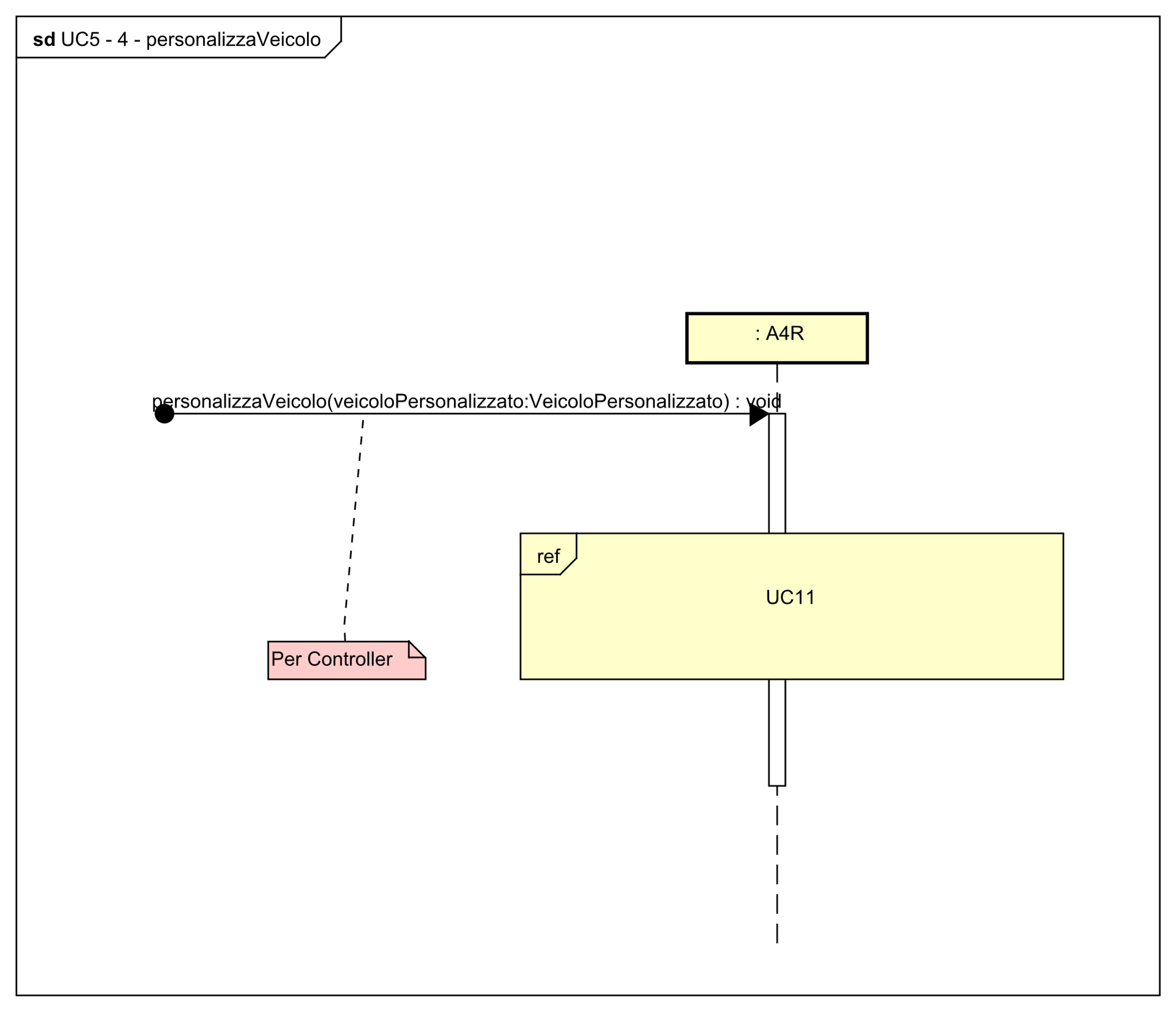
*Figura 3.2*

### 1.3 – scegliVeicoloAcquisto(codice : int, parco : Parco)



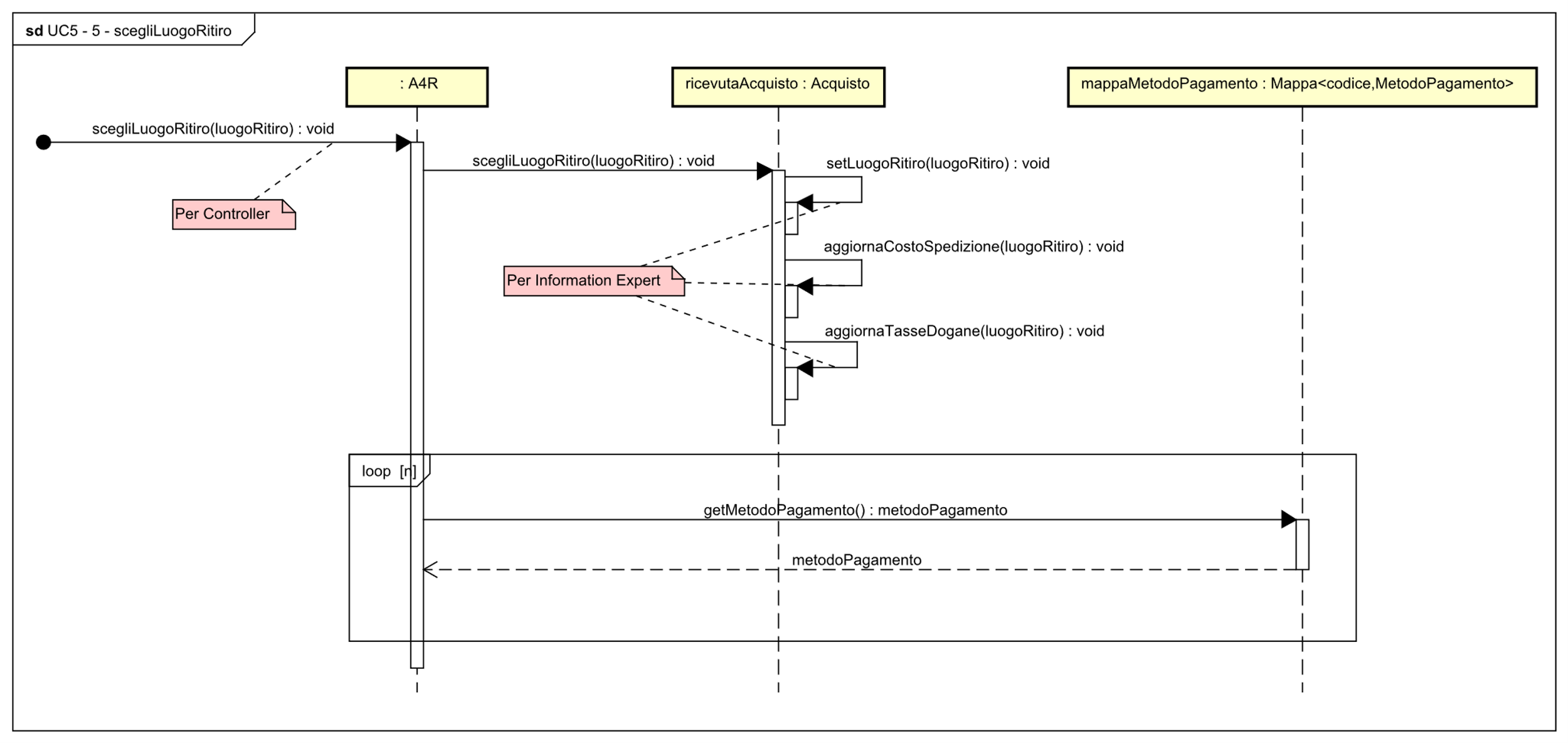
*Figura 3.3*

### 1.4 – personalizzaVeicolo(veicoloPersonalizzato : VeicoloPersonalizzato)



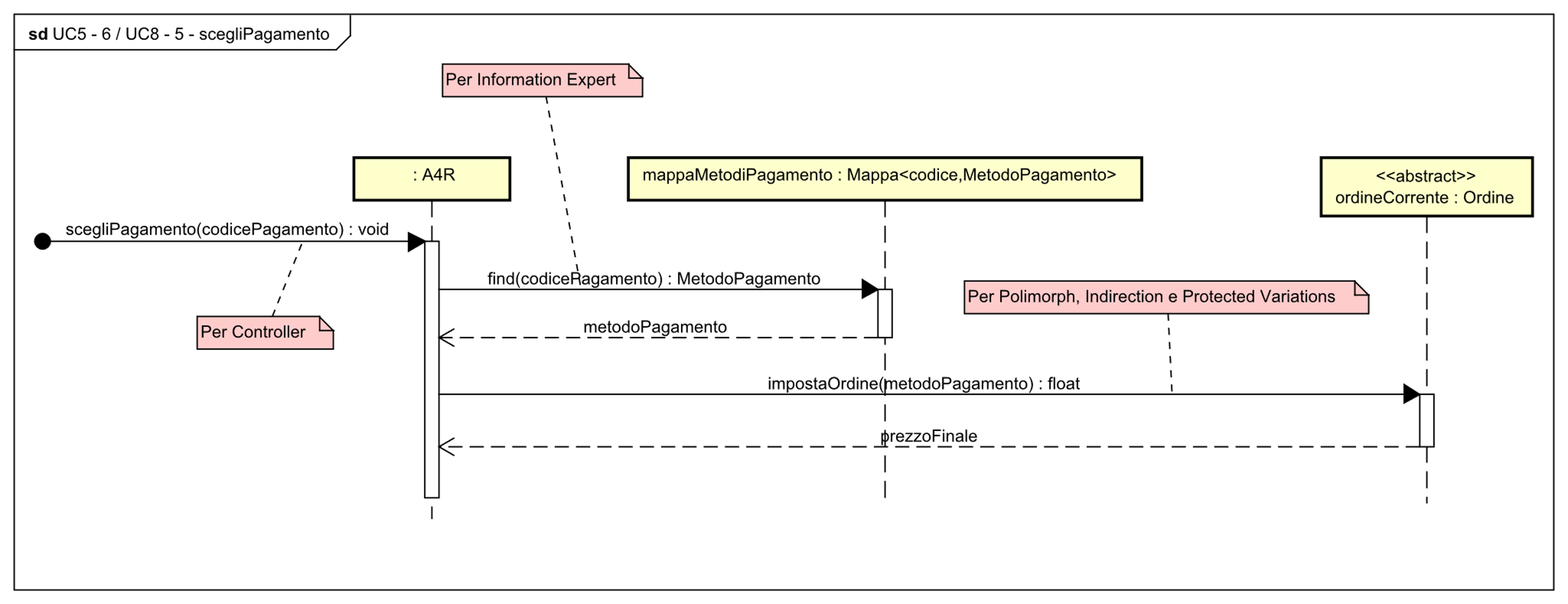
*Figura 3.4*

### 1.5 – scegliLuogoRitiro(luogoRitiro : String)



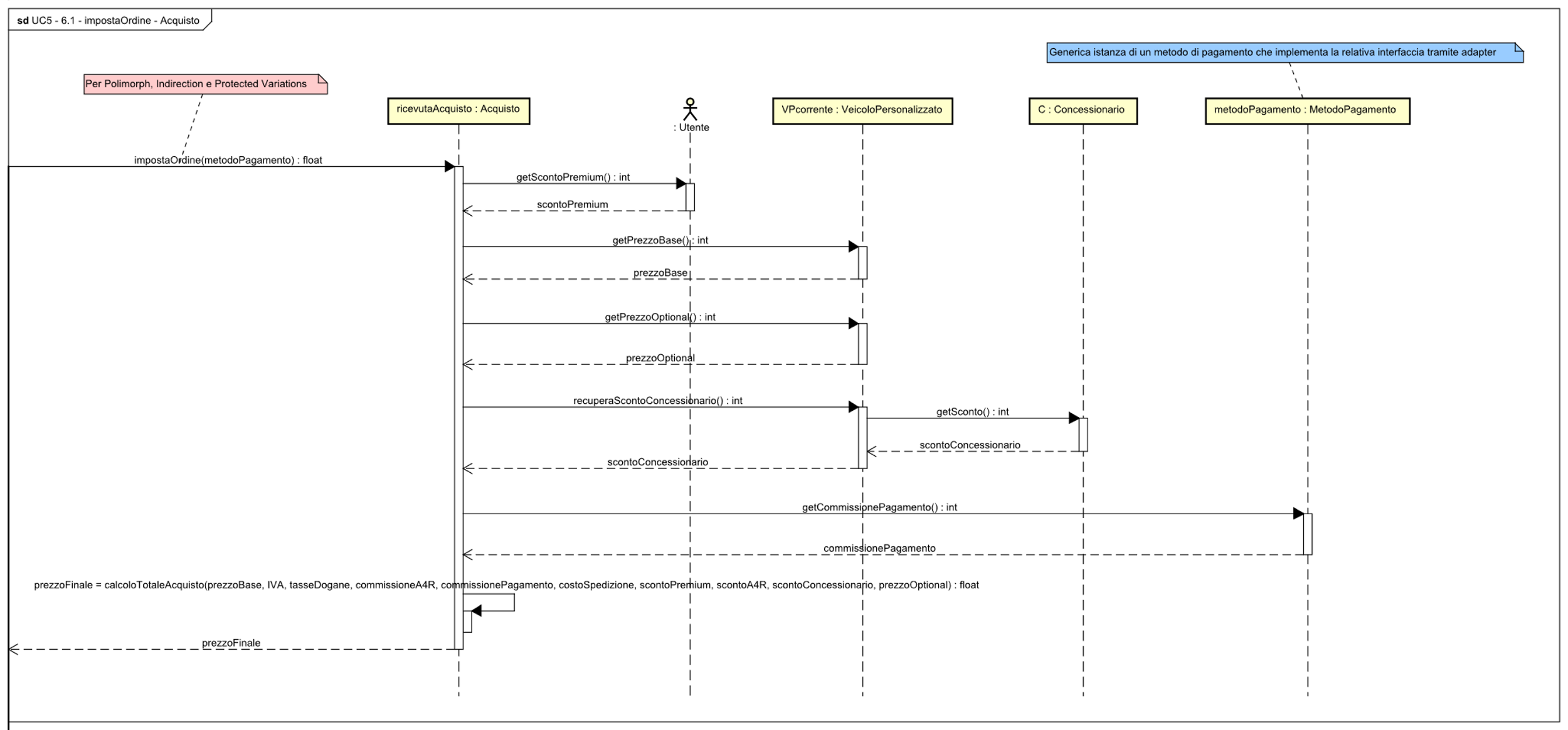
*Figura 3.5*

### 1.6 – scegliPagamento(codicePagamento : int)



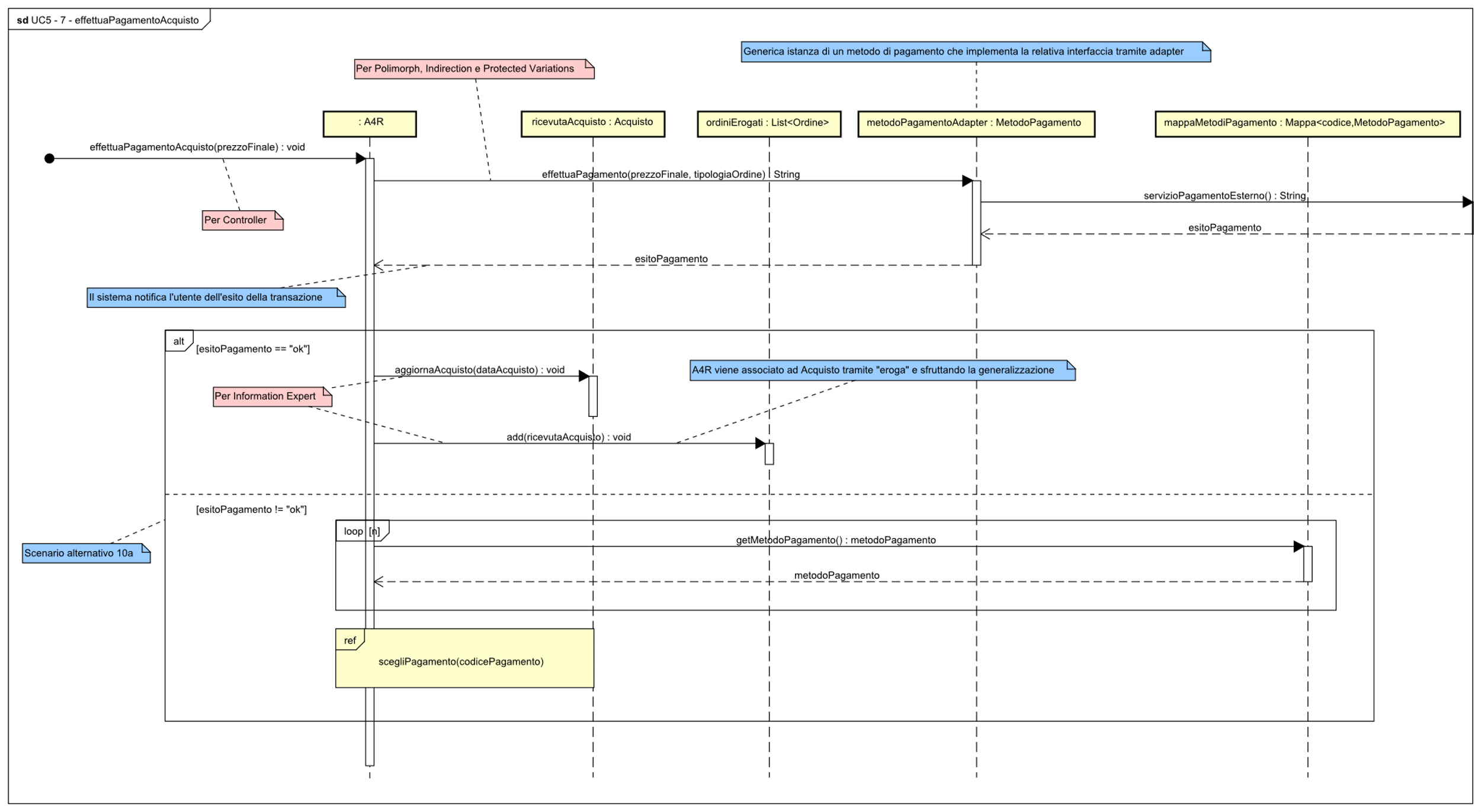
*Figura 3.6*

### 1.7 – impostaOrdine(metodoPagamento : MetodoPagamento) - Acquisto



*Figura 3.7*

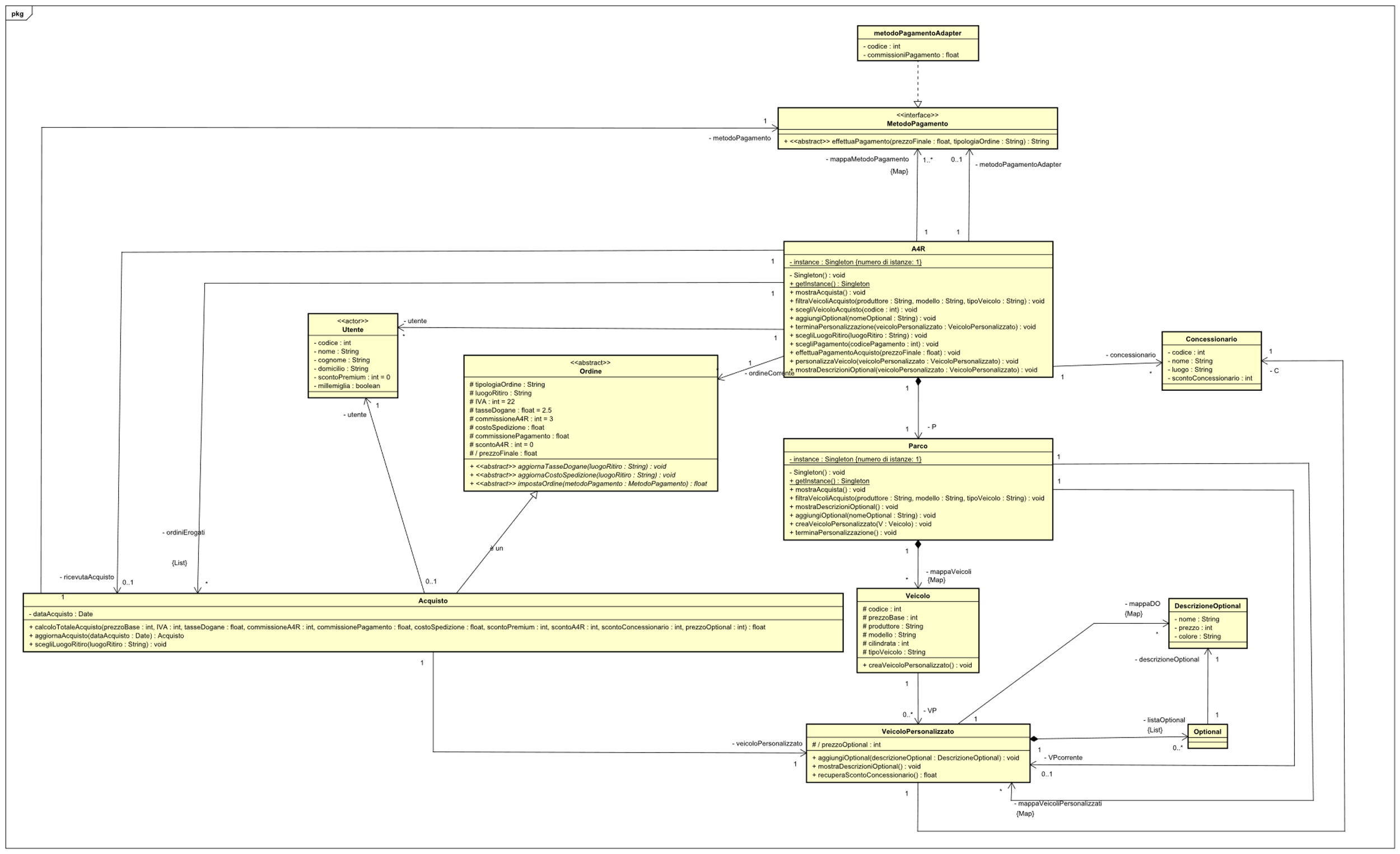
### 1.8 – effettuaPagamentoAcquisto(prezzoFinale : float)



*Figura 3.8*

## 2. Caso d’Uso 5 – Diagramma delle Classi

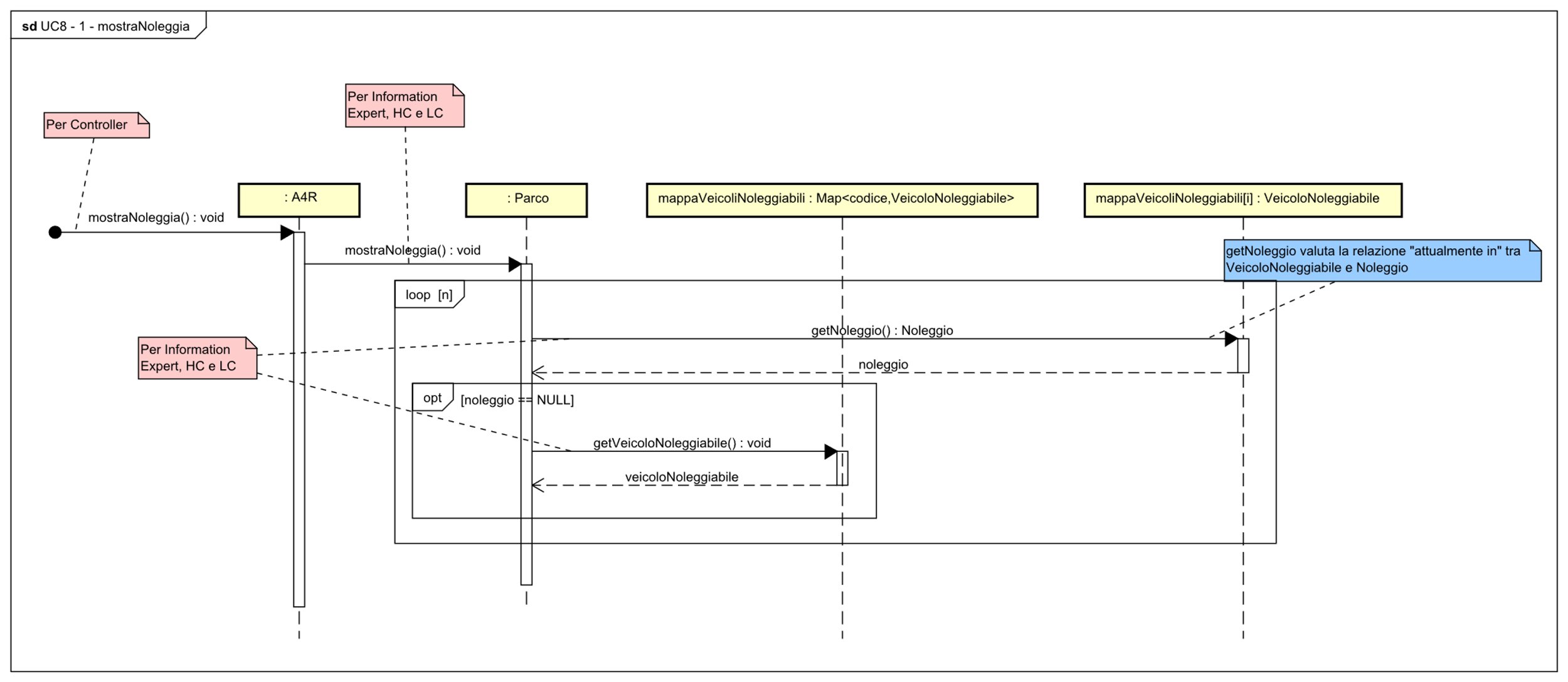
Abbiamo ritenuto opportuno rappresentare i *diagrammi delle classi* di *UC5* e *UC11* assieme, essendo il secondo una sotto-funzione del primo.



*Figura 4.1 - Diagramma delle classi dei casi d’uso UC5 e UC11*

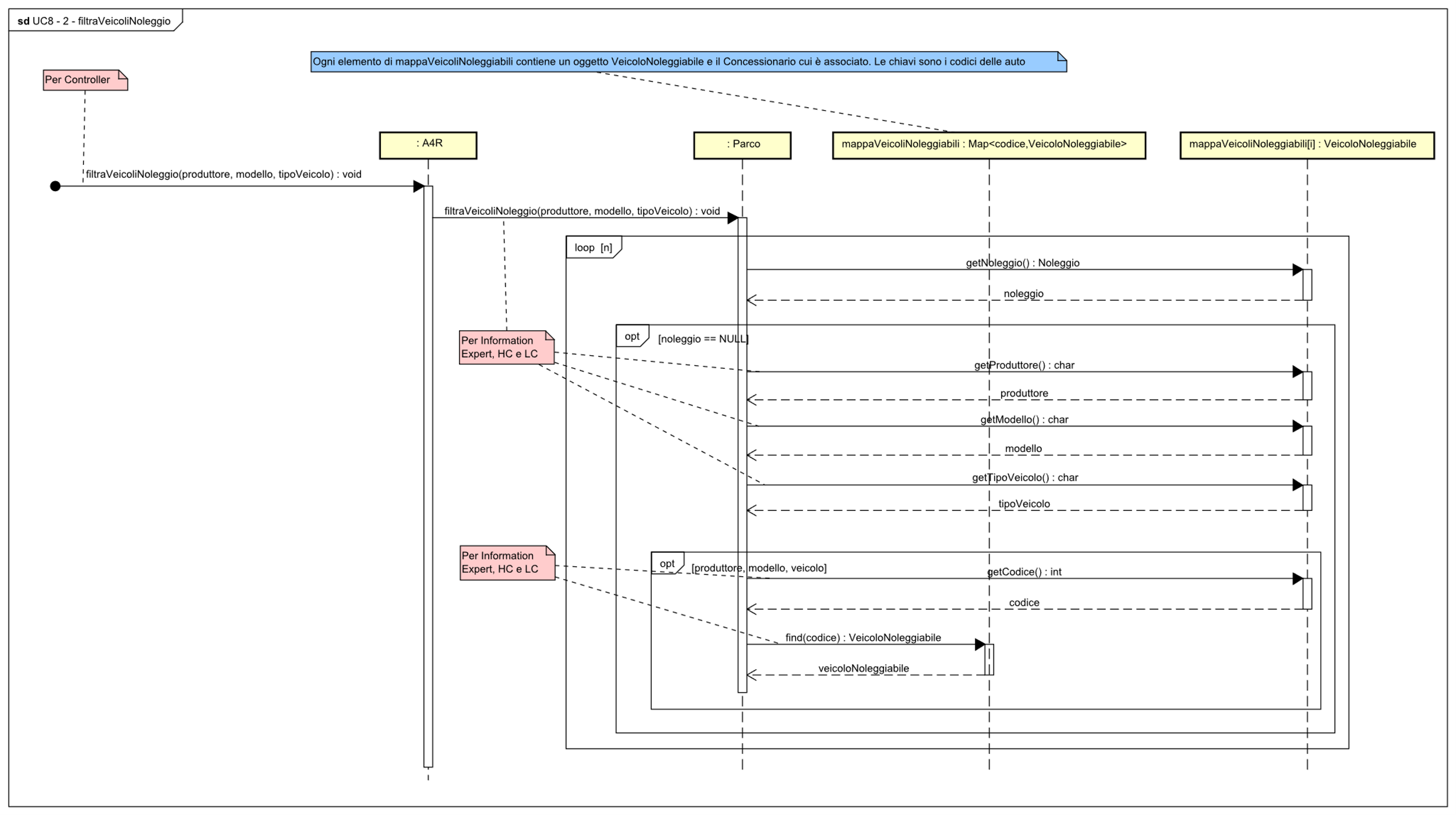
## 3. Caso d’Uso 8 – Diagrammi di Sequenza

### 3.1 mostraNoleggia()



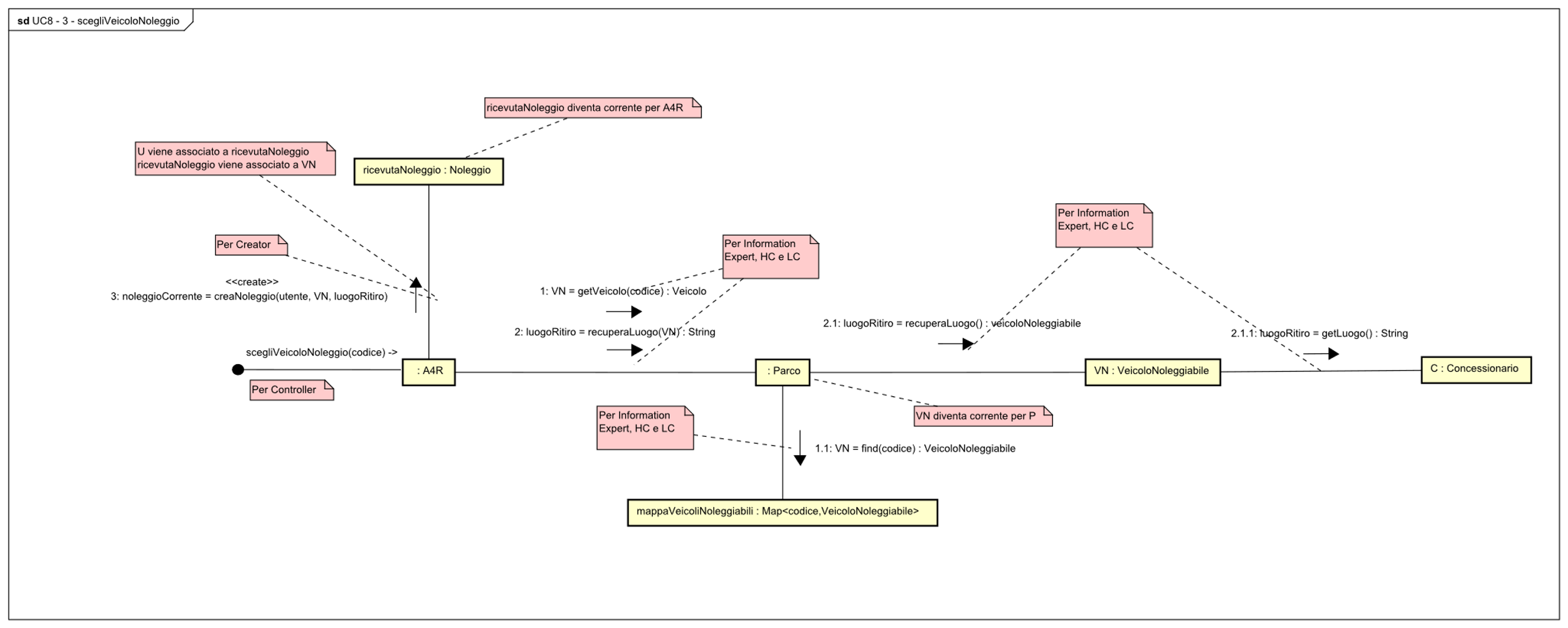
*Figura 5.1*

### 3.2 – filtraVeicoliNoleggio(produttore : String, modello : String, tipoVeicolo : String)

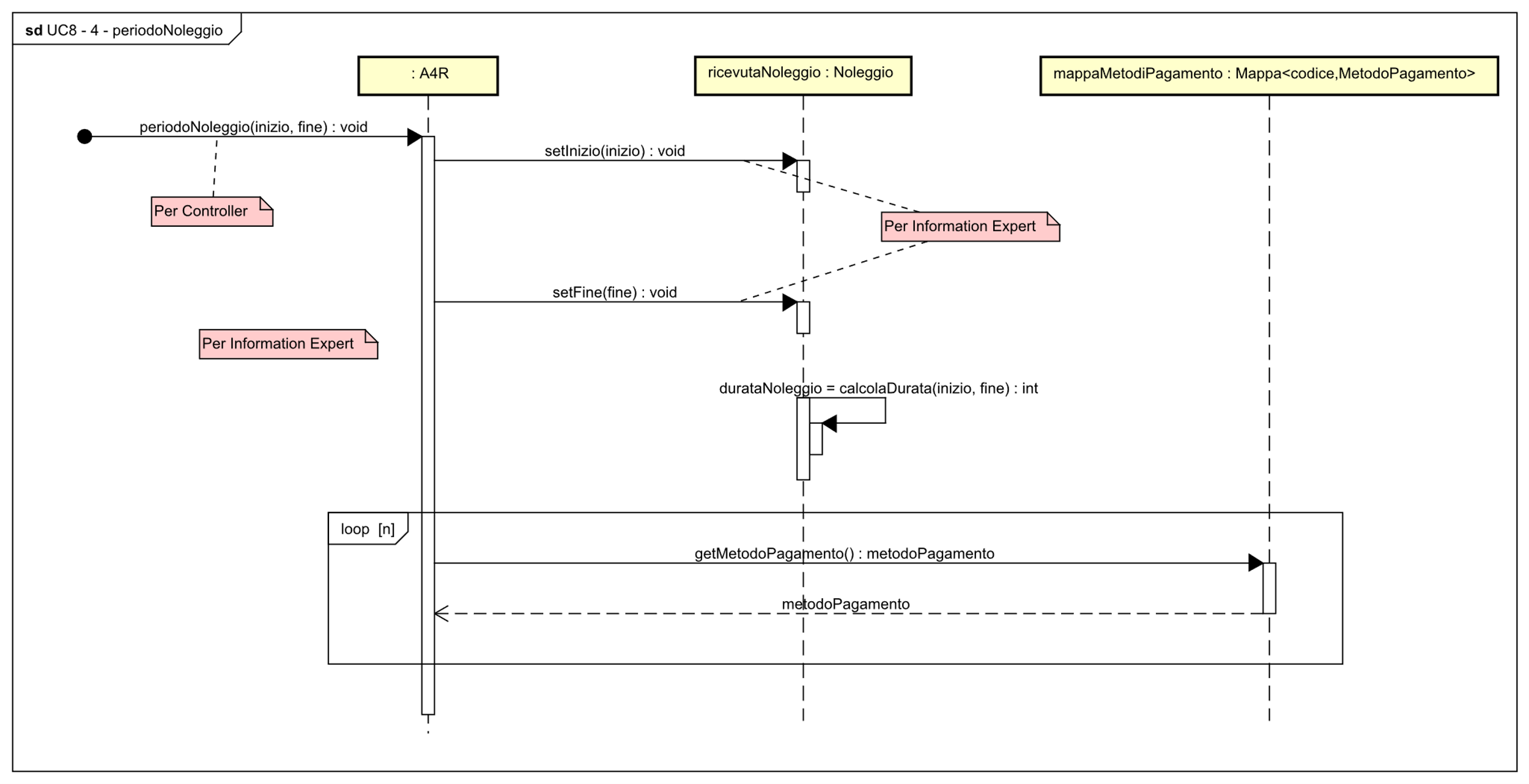


*Figura 5.2*

### 3.3 – scegliVeicoliNoleggio(veicoloNoleggiabile : VeicoloNoleggiabile)

*Figura 5.3*

### 3.4 – periodoNoleggio(inizio:date, fine:date )

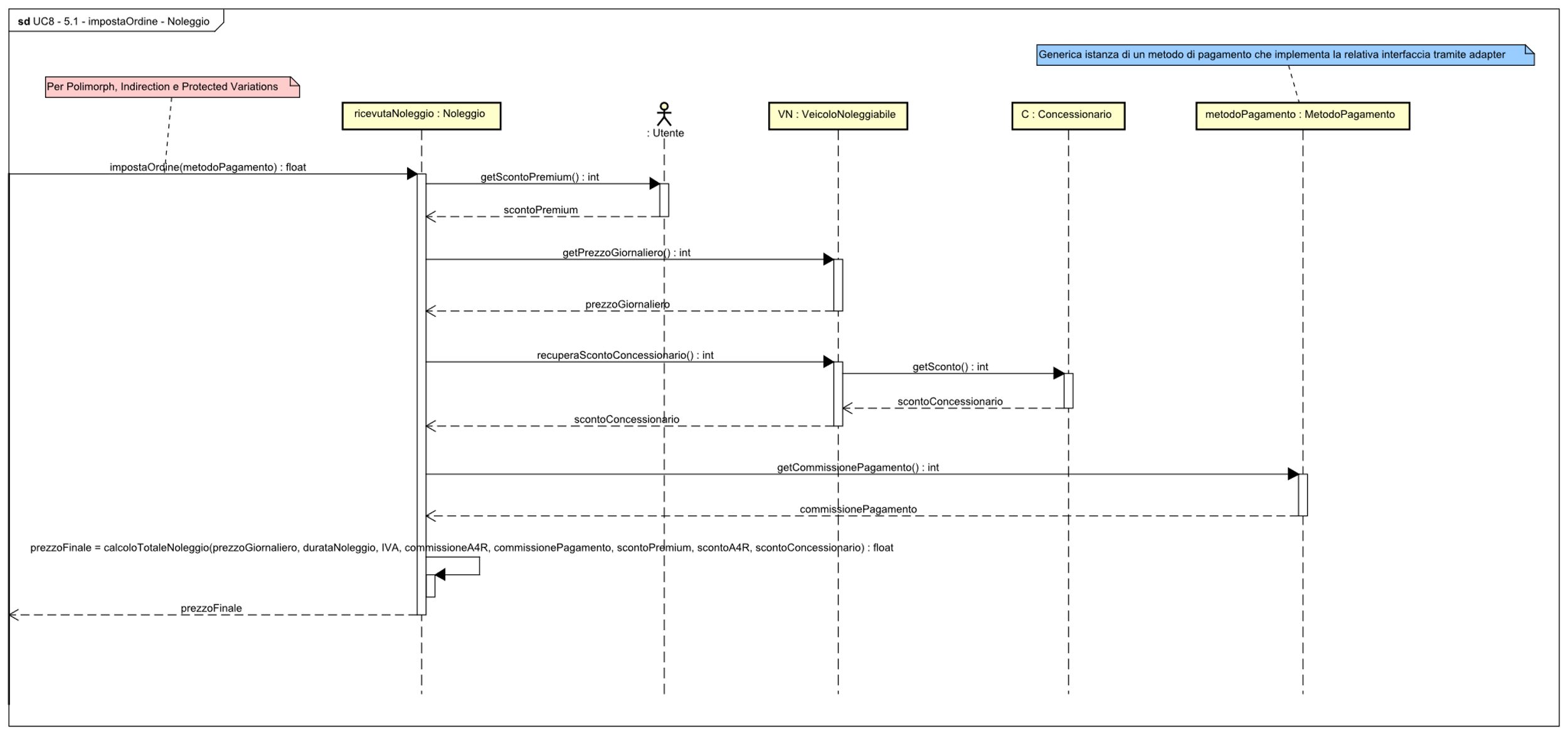


*Figura 5.4*

### 3.5 – scegliPagamento(codicePagamento : int)

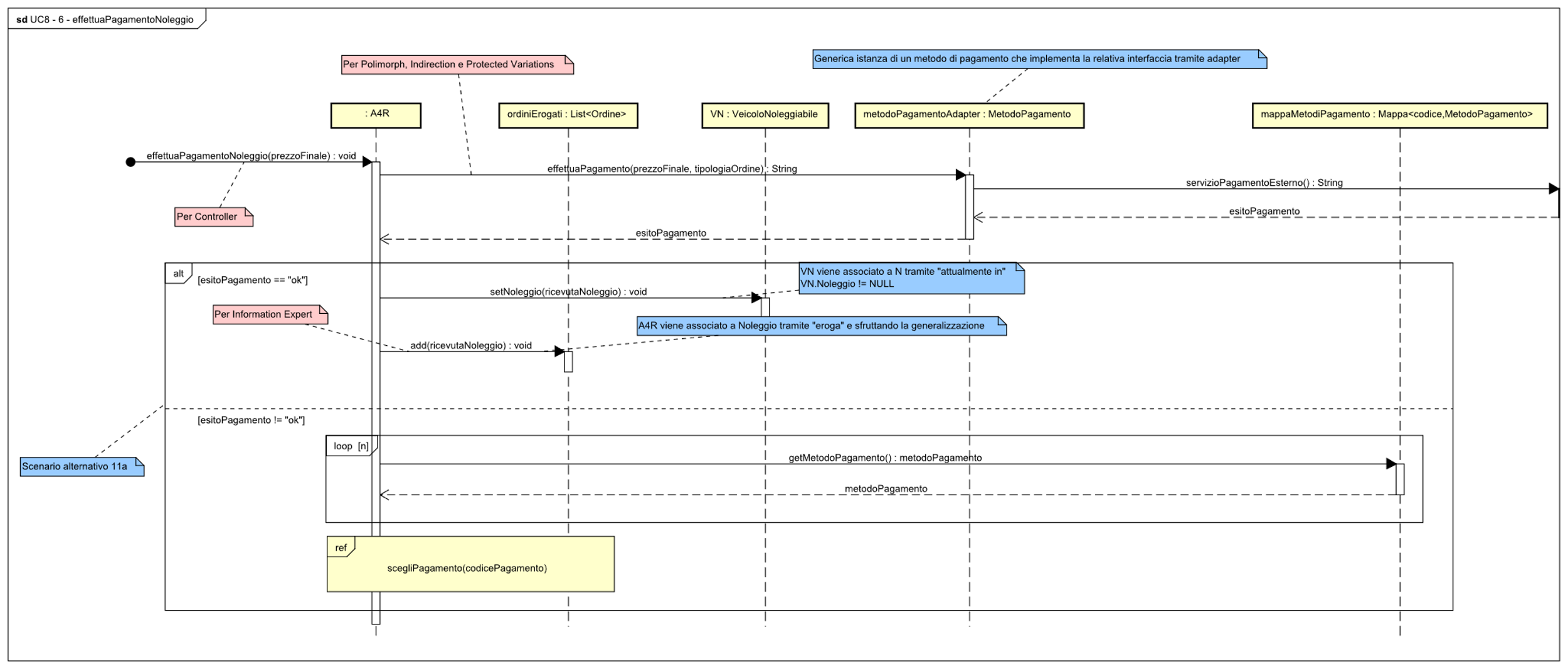
*(vedi Figura 3.6)*

### 3.6 – impostaOrdine(metodoPagamento : MetodoPagamento) – Noleggio



*Figura 5.5*

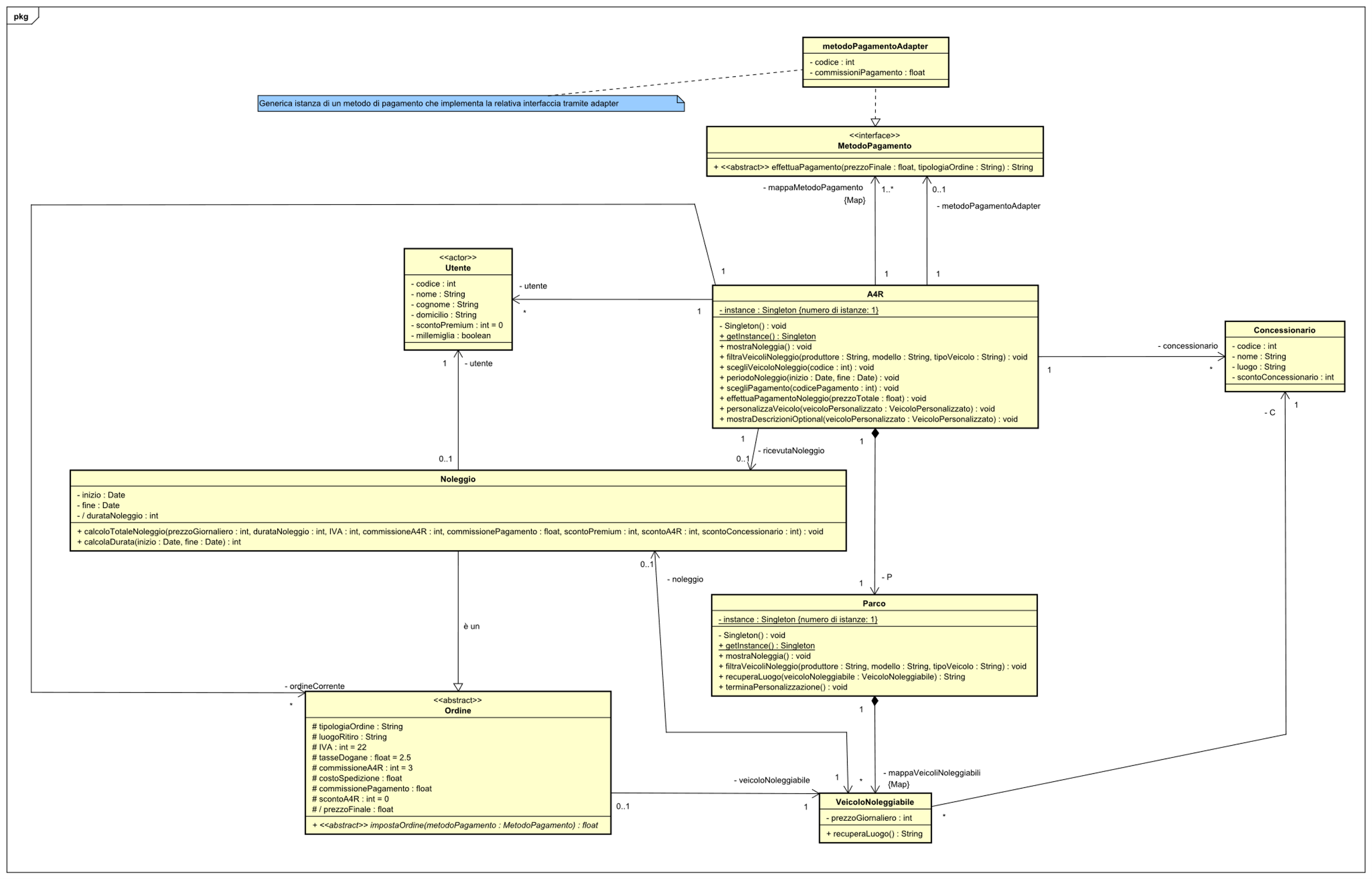
### 3.7 – effettuaPagamentoNoleggio(prezzoFinale : float)



*Figura 5.6*

## 4. Caso d’Uso 8 – Diagramma delle Classi

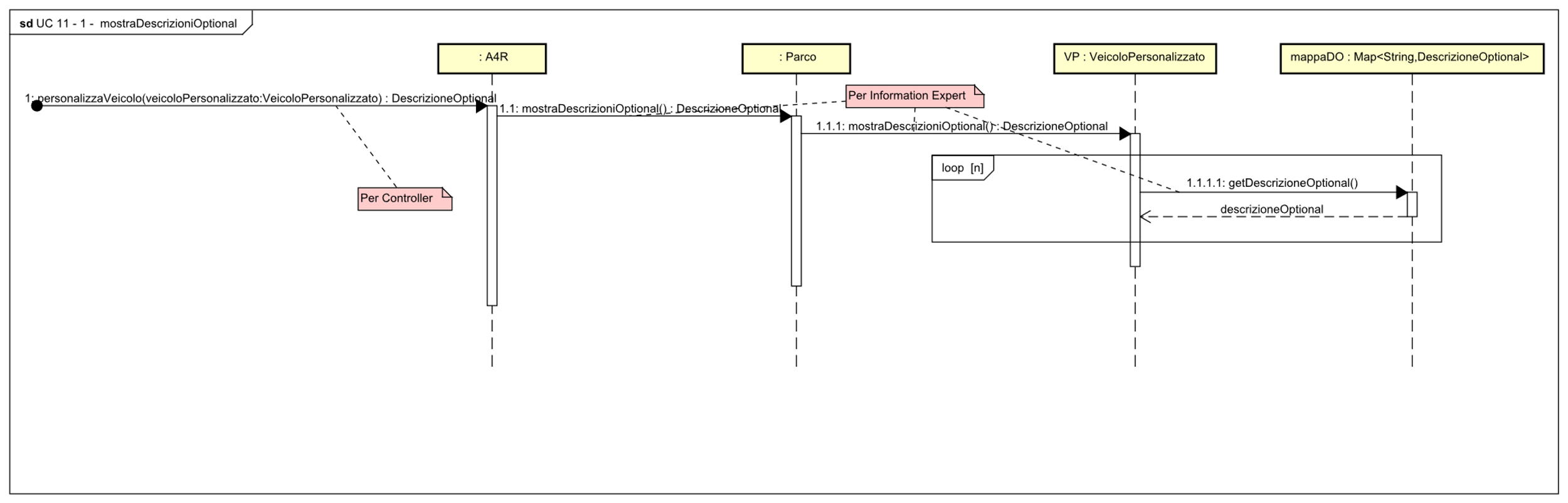
Presentiamo il *Diagramma delle Classi* software aggiornato, riferito al caso d’uso UC8.



*Figura 5.7*

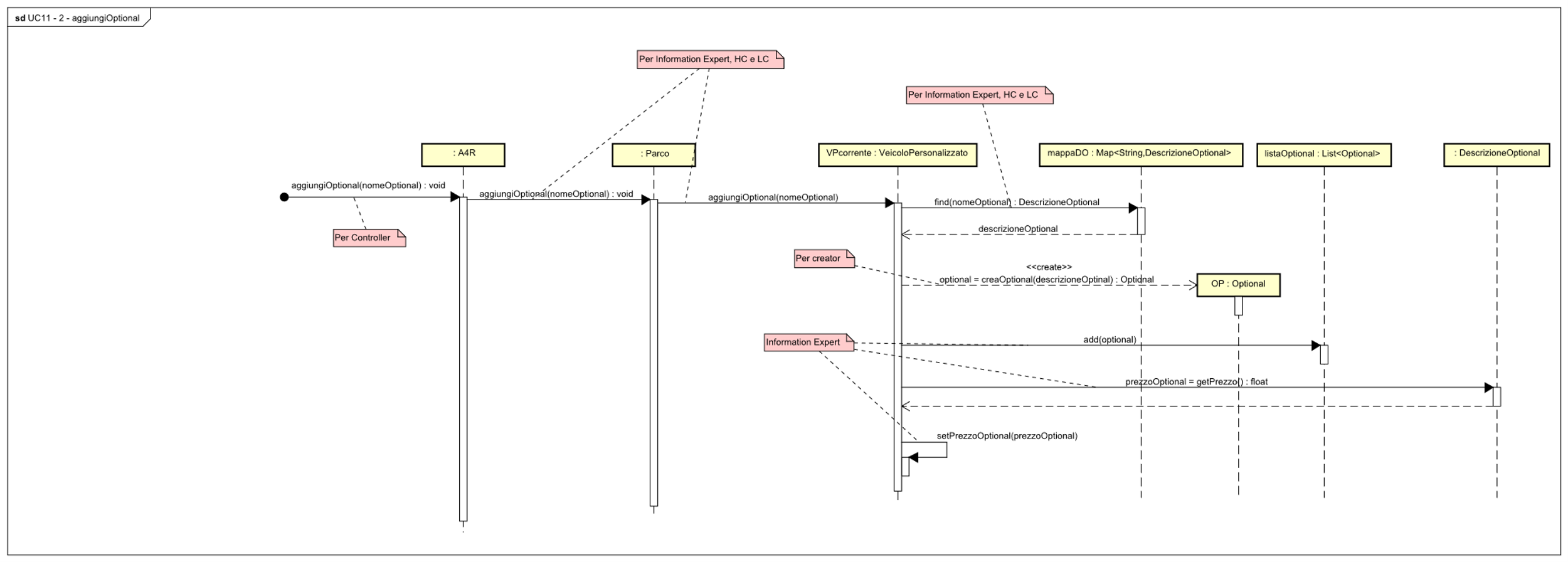
## 5. Caso d’Uso 11 – Diagrammi di Sequenza

### 5.1 – mostraDescrizioneOptional(descrizioneOptional : DescrizioneOptional)

**

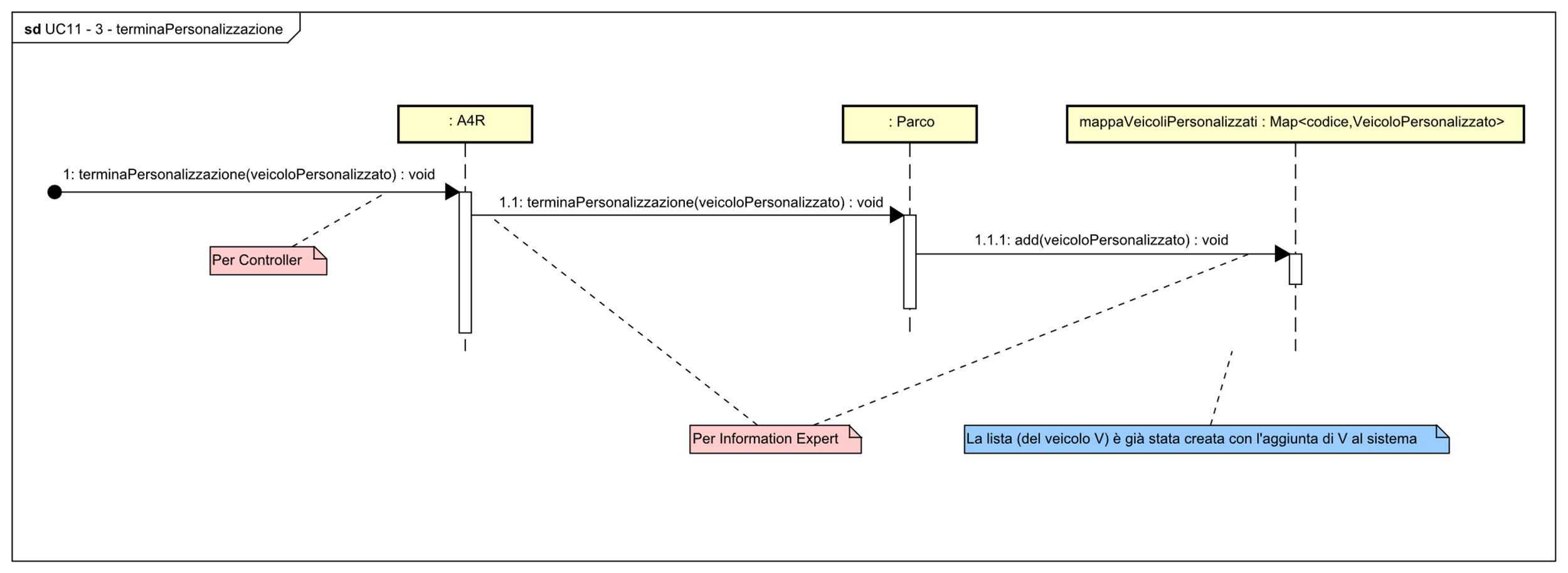
*Figura 5.8*

### 5.2 – aggiungiOptional(nomeOptional : String)

**

*Figura 5.9*

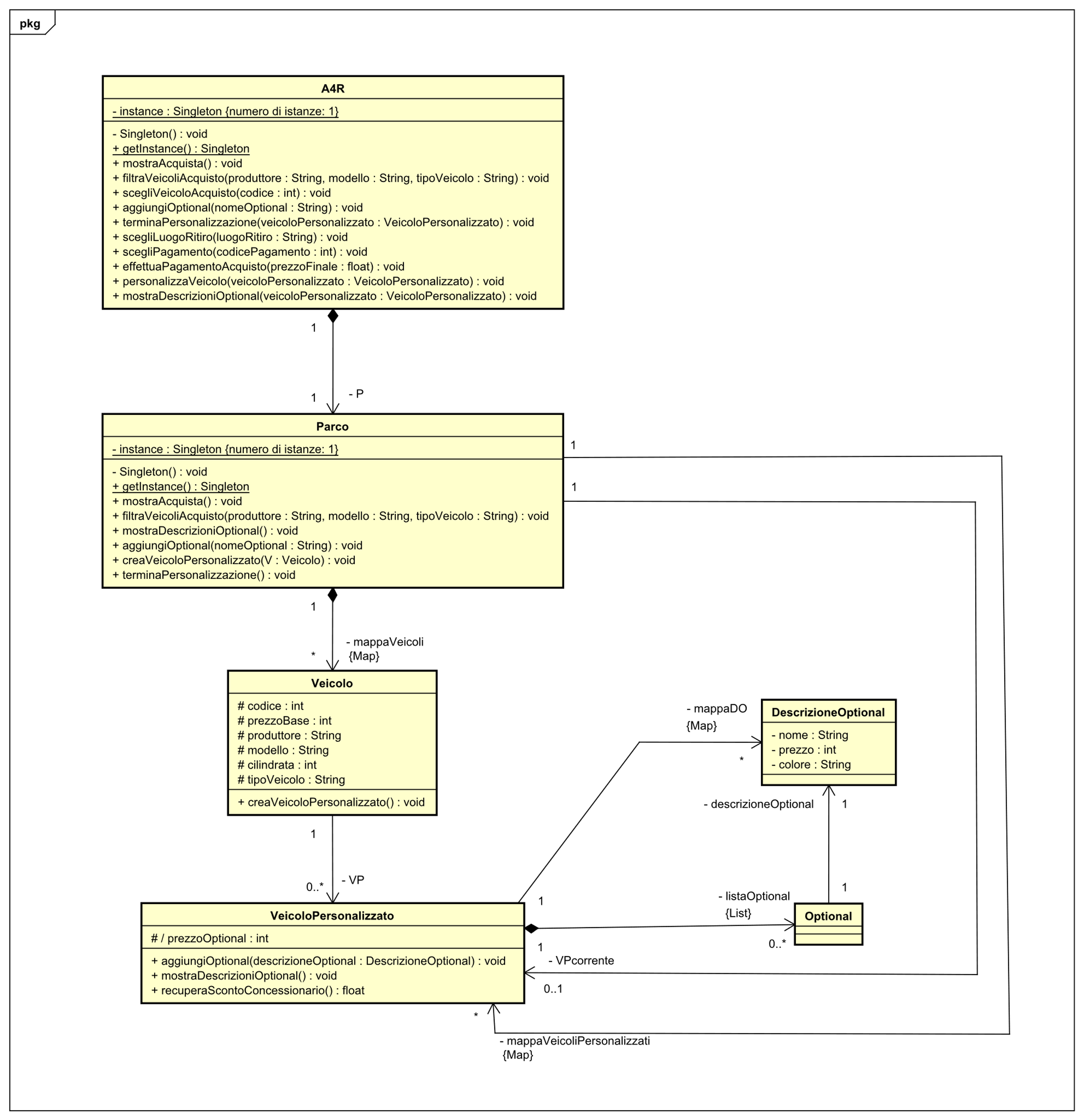
### 5.3 – terminaPersonalizzazione(veicoloPersonalizzato : VeicoloPersonalizzato)

**

*Figura 6.1*

## 6. Caso d’Uso 8 – Diagramma delle Classi

Presentiamo il *Diagramma delle Classi* software aggiornato, riferito al caso d’uso UC8.



*Figura 6.2*