

2	Elaborazione – Iterazione 1.....	3
2.1	Introduzione.....	3
2.2	Aggiornamento casi d’uso UC7, UC9 e UC10.....	3
	UC7: Effettua Pagamento	3
	UC9: Pulisci Camera	4
	UC10: Verifica disponibilità camera.....	5
2.3	Analisi orientata agli oggetti	6
2.3.1	Modello di dominio.....	6
	Caso d’uso UC1: Gestisci Camera.....	6
	Caso d’uso UC9: Pulisci Camera	7
	Caso d’uso UC10: Verifica disponibilità camera	8
2.3.2	Diagramma di sequenza di sistema	9
	Caso d’uso UC1: Gestisci Camera.....	9
	Caso d’uso UC9: Pulisci Camera	9
	Caso d’uso UC10: Verifica disponibilità camera	10
2.3.3	Contratti delle operazioni	10
	Caso d’uso UC1: Gestisci Camera.....	10
	Contratto CO1: inserisciNuovoTipoCamera.....	10
	Contratto CO2: definisciPrezzoCamera	10
	Contratto CO3: confermaTipoCamera	11
	Caso d’uso UC9: Pulisci Camera	11
	Contratto CO1: visualizzaCameraDaPulire.....	11
	Contratto CO2: selezionaCameraDaPulire.....	11
	Contratto CO3: registraFinePulizia	12
	Caso d’uso UC10: Verifica disponibilità camera	12
	Contratto CO1: verificaCheck-out.....	12

Contratto CO2: verificaFinePulizia	12
Contratto CO3: registraCheck-in.....	12
2.4 Progettazione	13
2.4.1 Diagramma di sequenza.....	13
Caso d'uso UC1: Gestisci Camera.....	13
Caso d'uso UC9: Pulisci Camera	15
Caso d'uso UC10: Verifica disponibilità camera	16
2.4.2 Diagramma delle classi.....	18
Caso d'uso UC1: Gestisci Camera.....	18
Caso d'uso UC9: Pulisci Camera	18
Caso d'uso UC10: Verifica disponibilità camera	19

2 Elaborazione – Iterazione 1

2.1 Introduzione

La fase di ideazione si è conclusa, si passerà adesso alla fase di elaborazione in cui si andrà a identificare la maggior parte dei requisiti e si andranno a risolvere le problematiche relative ai rischi maggiori. Per questa prima iterazione si è scelto di introdurre un'analisi più dettagliata di alcuni casi d'uso che necessitano una descrizione estesa, vista la loro importanza all'interno del dominio di interesse.

2.2 Aggiornamento casi d'uso UC7, UC9 e UC10

Vengono rappresentati in formato esteso i casi d'uso UC7, UC9, UC10, scoperti durante la fase di ideazione. Per le due successive fasi di Analisi orientata agli oggetti e Progettazione verranno trattati i casi d'uso UC1, UC9 ed UC10.

UC7: Effettua Pagamento

Nome caso d'uso	UC7: Effettua pagamento
Portata	Applicazione Amazing Hotel & Camper
Livello	Obiettivo utente
Attore primario	Cliente
Parti interessate e interessi	<ul style="list-style-type: none">- <i>Cliente</i>: Vuole effettuare il pagamento del soggiorno richiesto e fruire di un servizio rapido, nel modo più semplice possibile.- <i>Addetto del sistema</i>: Vuole che le procedure di pagamento siano veloci e semplificate; vuole che le informazioni relative ai pagamenti dei clienti siano aggiornate.
Pre-condizione	Il Cliente è identificato e autenticato.
Garanzie di successo	Il soggiorno richiesto viene pagato. Il cliente ha la garanzia di usufruire del soggiorno nel periodo richiesto.
Scenario principale di successo	<ol style="list-style-type: none">1. Il Cliente richiede al sistema il pagamento del soggiorno associato alla prenotazione effettuata.2. Il sistema informa il Cliente dell'importo da pagare e mostra le modalità di pagamento consentite.<ul style="list-style-type: none">• Carta di credito/debito• Bonifico bancario3. Il Cliente seleziona la modalità di pagamento desiderata.4. Il sistema richiede al Cliente le informazioni utili per effettuare il pagamento.<ol style="list-style-type: none">a) Numero carta di credito.b) Nome e Cognome del proprietario della carta.c) Codice CVV.d) Codice IBAN (per Bonifico bancario).5. Il Cliente inserisce i dati richiesti per effettuare il pagamento.

	6. Il sistema notifica il pagamento avvenuto con successo. 7. Il sistema genera un riepilogo con annessa fattura. 8. Il Cliente ha la possibilità di salvare o stampare la fattura.
Estensioni	<p>*a. In qualsiasi momento, il sistema fallisce e ha un arresto improvviso.</p> <ol style="list-style-type: none"> 1. Il Cliente riavvia il software. 2. Il sistema recupera lo stato precedente all'arresto improvviso. 3. Il cliente riprende le operazioni di pagamento. <p>1a. Il Cliente inserisce i dati relativi alla carta di credito/debito/bonifico bancario in maniera errata.</p> <ol style="list-style-type: none"> 1. Il sistema avverte il Cliente del campo relativo al dato inserito erroneamente e richiede nuovamente l'inserimento. 2. Il Cliente re-inserisce i dati e prosegue con le procedure mancanti. <p>2a. Il Cliente non dispone del credito sufficiente associato al metodo di pagamento scelto.</p> <ol style="list-style-type: none"> 1. Il sistema avverte il Cliente del credito insufficiente e richiede al Cliente di scegliere nuovamente la modalità di pagamento. <p>3a. Il Cliente decide di non proseguire con il pagamento o decide di pagare in seguito.</p> <ol style="list-style-type: none"> 1. Il cliente termina l'esecuzione del programma.
Requisiti speciali	
Elenco delle variabili tecnologiche e dei dati	
Frequenze di ripetizioni	Legata al numero di prenotazioni effettuate.
Varie	

UC9: Pulisci Camera

Nome caso d'uso	UC9: Pulisci Camera
Portata	Applicazione Amazing Hotel & Camper
Livello	Obiettivo utente
Attore primario	Addetto delle pulizie
Parti interessate e interessi	<ul style="list-style-type: none"> - <i>Addetto delle pulizie</i>: Vuole conoscere le camere da pulire della struttura per semplificare e rendere ordinato il suo lavoro, usufruendo di un servizio rapido e semplice. - <i>Addetto del sistema</i>: Vuole gestire l'intero processo di pulizia delle camere in maniera corretta e veloce; vuole che le informazioni relative alle pulizie delle camere siano aggiornate.
Pre-condizione	L'addetto delle pulizie è identificato e autenticato.

Garanzie di successo	La camera viene pulita. L'Addetto del sistema può far accomodare il nuovo Cliente. Il nuovo cliente può iniziare il soggiorno richiesto.
Scenario principale di successo	<ol style="list-style-type: none"> 1. L'addetto delle pulizie richiede al sistema quali camere devono essere pulite. 2. Il sistema restituisce la lista delle camere da pulire. 3. L'addetto prende visione della lista e seleziona una camera da pulire. 4. Al termine della pulizia l'addetto delle pulizie registra tramite il sistema la fine della pulizia. <p>I passi 3-4 vengono ripetuti fino all'esaurimento delle camere da pulire.</p>
Estensioni	<p>*a. In qualsiasi momento, il sistema fallisce e ha un arresto imprevisto.</p> <ol style="list-style-type: none"> 1. L'Addetto delle pulizie riavvia il software e richiede il ripristino dello stato precedente del sistema. 2. Il sistema ricostruisce lo stato precedente. <p>1a. La camera selezionata dall'Addetto delle pulizie risulta ancora occupata dagli ospiti del soggiorno precedente.</p> <ol style="list-style-type: none"> 1. L'addetto delle pulizie registra tramite il sistema il rinvio della pulizia. 2. L'addetto delle pulizie seleziona un'altra camera da pulire. 3. Al termine della pulizia l'addetto delle pulizie registra tramite il sistema la fine della pulizia. <p>I passi 1-2 si possono ripetere finché non viene trovata una camera libera da pulire.</p> <p>2a. Tutte le camere risultano pulite.</p> <ol style="list-style-type: none"> 1. L'Addetto delle pulizie termina l'esecuzione del programma. <p>3a. L'Addetto delle pulizie non vuole proseguire con la pulizia della camera, causa imprevisti.</p> <ol style="list-style-type: none"> 1. L'Addetto delle pulizie termina l'esecuzione del programma.
Requisiti speciali	
Elenco delle variabili tecnologiche e dei dati	
Frequenze di ripetizioni	Almeno una volta al giorno, se vi sono stati ospiti nella struttura il giorno precedente.
Varie	

UC10: Verifica disponibilità camera

Nome caso d'uso	UC10: Verifica disponibilità camera
Portata	Applicazione Amazing Hotel & Camper
Livello	Obiettivo utente
Attore primario	Addetto del sistema

Parti interessate e interessi	<ul style="list-style-type: none"> - <i>Addetto del sistema</i>: Vuole gestire l'intero processo di accoglienza dei clienti in maniera corretta e veloce; vuole che le informazioni relative alle disponibilità delle camere siano aggiornate. - <i>Cliente</i>: Vuole ricevere un'accoglienza adeguata dalla struttura per il servizio richiesto, in maniera rapida e semplice.
Pre-condizione	L'addetto è identificato e autenticato.
Garanzie di successo	La camera richiesta dal cliente risulta libera. La verifica della disponibilità è avvenuta con successo. L'Addetto può accogliere il nuovo cliente. Il cliente può usufruire del soggiorno prenotato.
Scenario principale di successo	<ol style="list-style-type: none"> 1. Il cliente arriva presso la struttura e richiede all'Addetto del sistema la camera prenotata. 2. L'addetto del sistema verifica l'uscita degli ospiti del soggiorno precedente (check-out). 3. L'addetto del sistema verifica la fine della pulizia della camera. 4. L'addetto del sistema registra il soggiorno occupato dal Cliente (check-in), consegna la chiave della camera al Cliente e accompagna i nuovi ospiti nella stessa.
Estensioni	<p>*a. In qualsiasi momento, il sistema fallisce e ha un arresto improvviso.</p> <ol style="list-style-type: none"> 1. L'Addetto del sistema riavvia il software e richiede il ripristino dello stato precedente del sistema. 2. Il sistema ricostruisce lo stato precedente. <p>1a. La camera non risulta disponibile per verifica negativa del passo 2 o 3 dello scenario principale di successo.</p> <ol style="list-style-type: none"> 1. L'addetto del sistema comunica la momentanea indisponibilità della camera al cliente, e lo invita ad accomodarsi nella sala d'attesa. 2. L'addetto del sistema provvede alle operazioni necessarie per liberare la camera.
Requisiti speciali	
Elenco delle variabili tecnologiche e dei dati	
Frequenze di ripetizioni	Legata al numero di prenotazioni associate ai clienti che dovranno soggiornare giornalmente.
Varie	

2.3 Analisi orientata agli oggetti

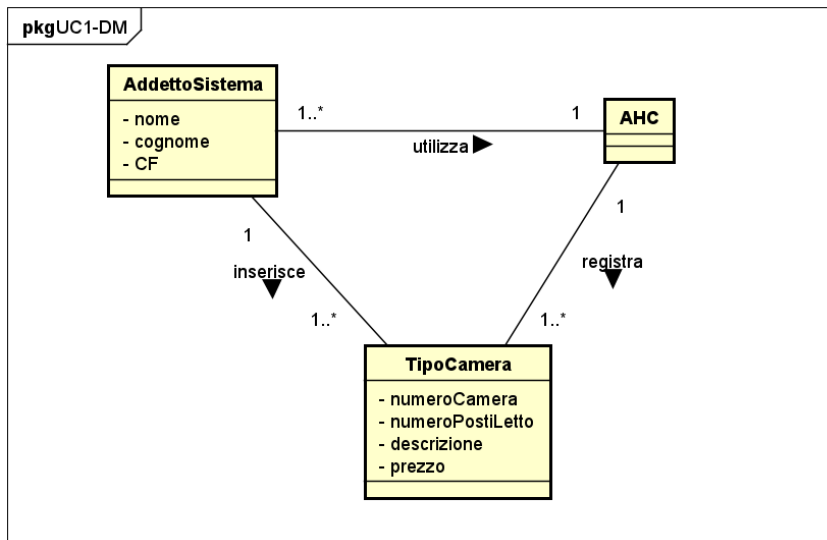
2.3.1 Modello di dominio

Caso d'uso UC1: Gestisci Camera

Valutato lo scenario principale di successo sono state identificate le seguenti classi concettuali:

- *AddettoSistema*: attore primario, che interagisce direttamente col sistema;
- *AHC*: rappresenta il sistema Amazing hotel & Camper;

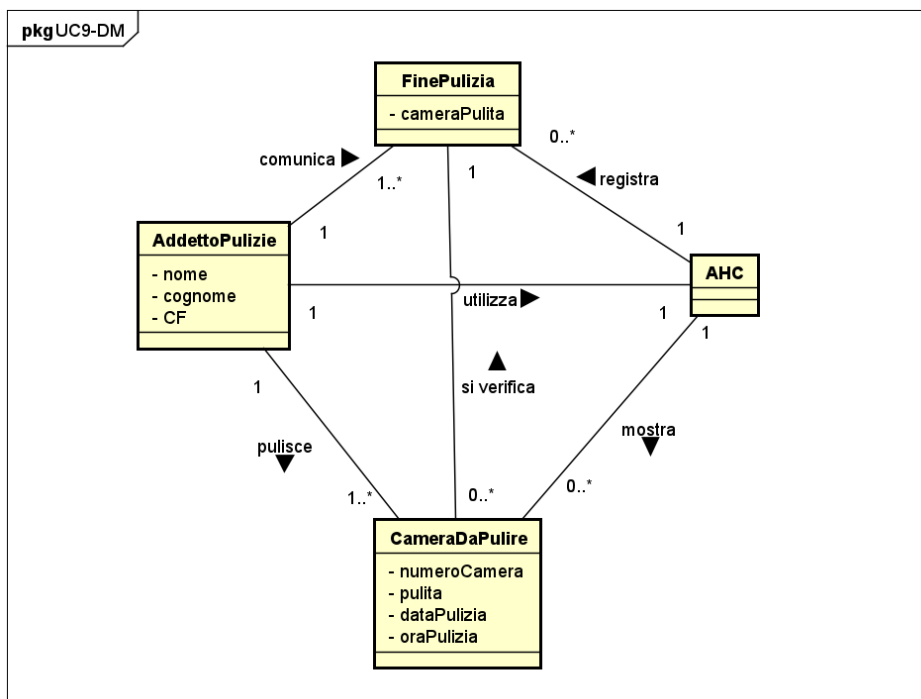
- *TipoCamera*: discrimina il tipo di camera tra quelle disponibili in struttura, e contiene una descrizione.



Caso d'uso UC9: Pulisci Camera

Valutato lo scenario principale di successo sono state identificate le seguenti classi concettuali:

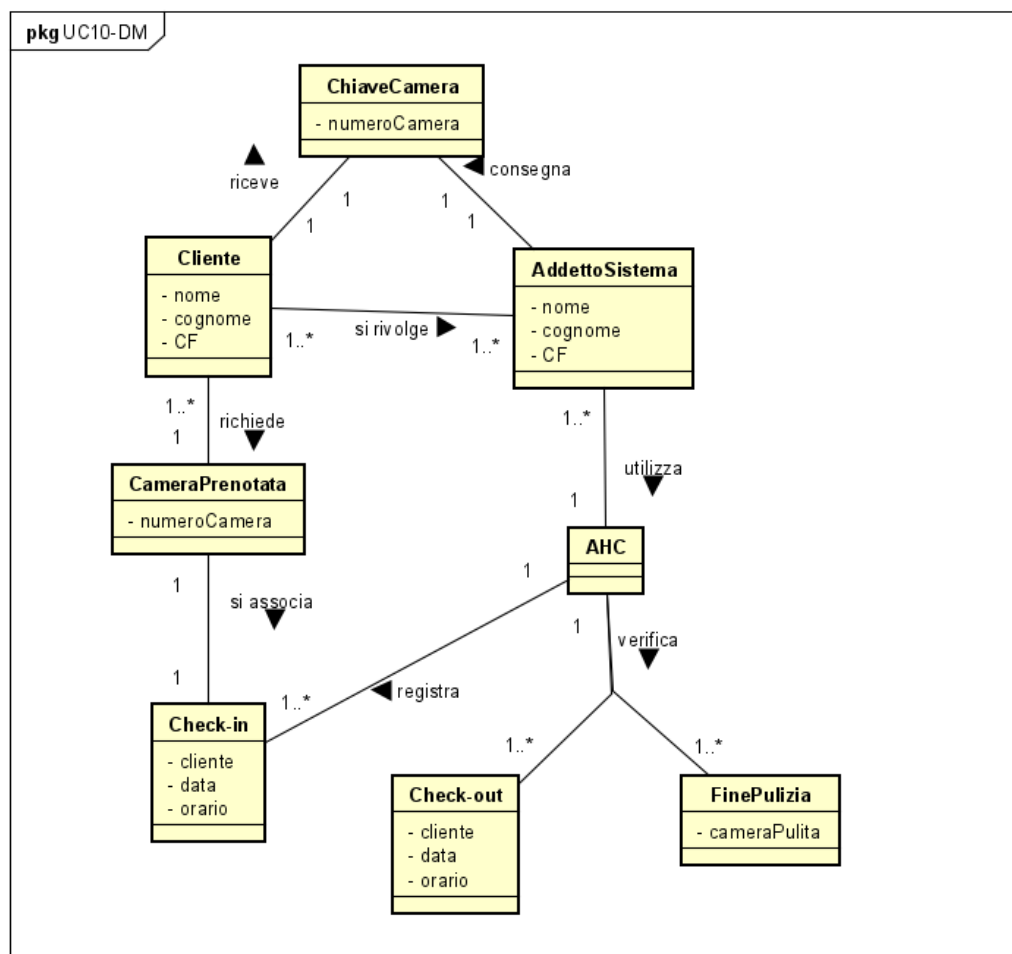
- *AddettoPulizie*: attore primario, che interagisce direttamente col sistema;
- *AHC*: rappresenta il sistema Amazing hotel & Camper;
- *CameraDaPulire*: rappresenta una camera per cui deve essere fatta la pulizia;
- *FinePulizia*: rappresenta la finalizzazione della pulizia di una camera.



Caso d'uso UC10: Verifica disponibilità camera

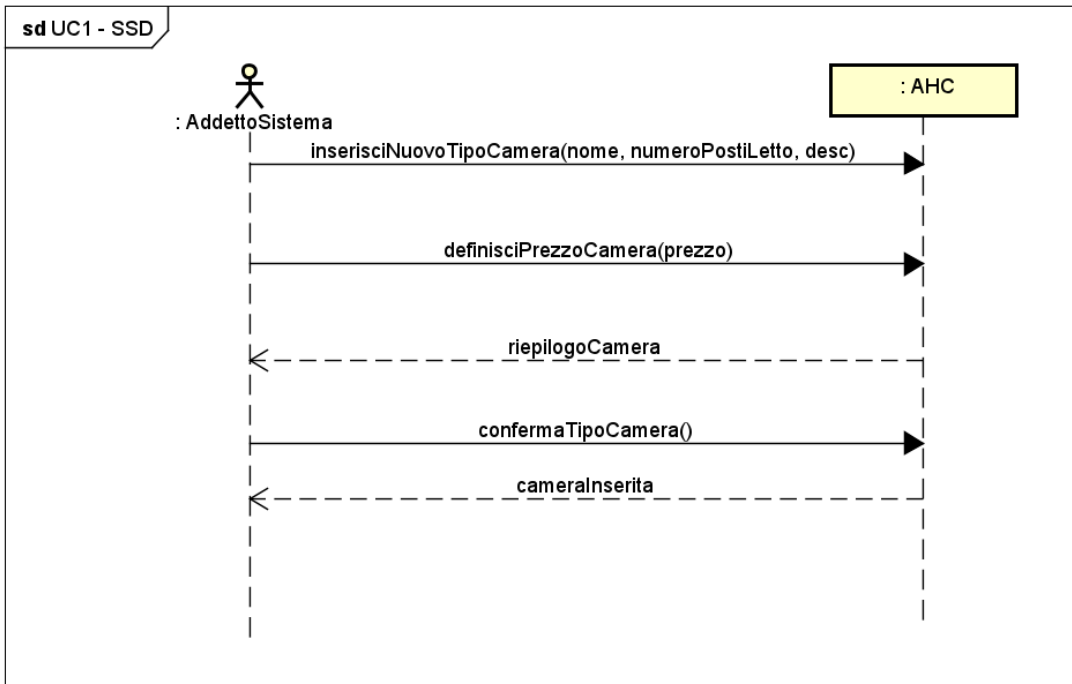
Valutato lo scenario principale di successo sono state identificate le seguenti classi concettuali:

- *AddettoSistema*: attore primario, che interagisce direttamente col sistema;
- *AHC*: rappresenta il sistema Amazing hotel & Camper;
- *CameraPrenotata*: rappresenta una camera che è stata bloccata tramite prenotazione;
- *Check-in*: rappresenta il momento di ingresso degli ospiti dalla camera, e di inizio del soggiorno;
- *Check-out*: rappresenta il momento di uscita degli ospiti dalla camera, e di terminazione del soggiorno;
- *ChiaveCamera*: rappresenta l'oggetto utile per avere accesso alla camera;
- *Cliente*: attore finale, utilizzatore dei servizi del SuD;
- *FinePulizia*: rappresenta la finalizzazione della pulizia di una camera.
- *Soggiorno*: indica la permanenza temporanea in un determinato luogo da parte del Cliente.

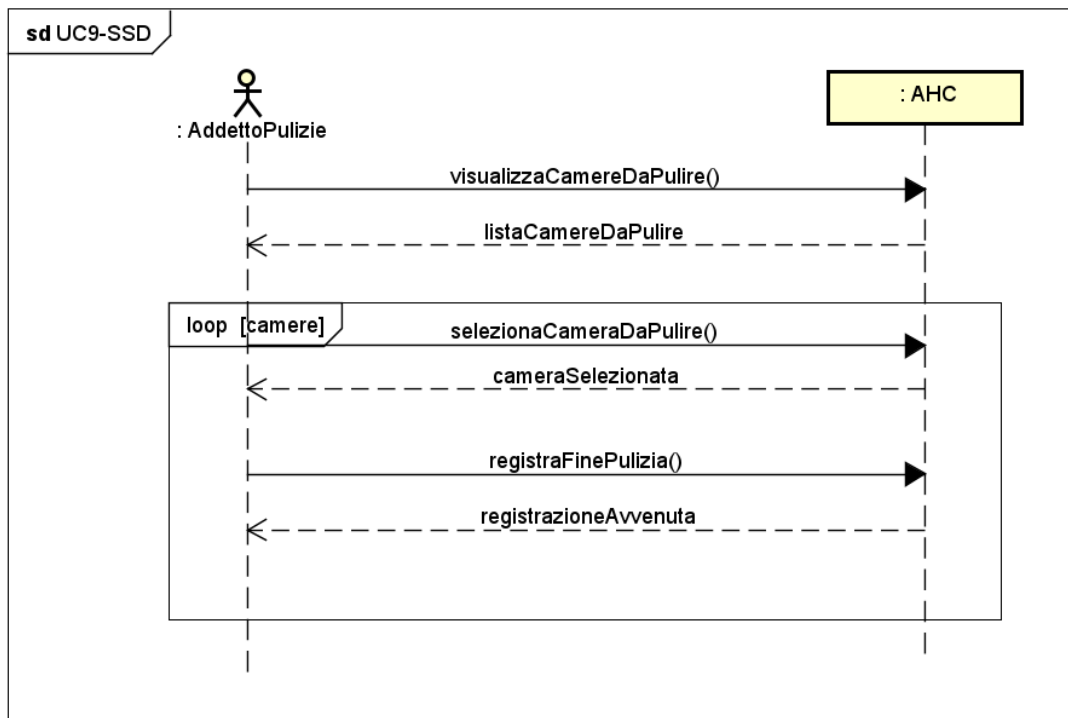


2.3.2 Diagramma di sequenza di sistema

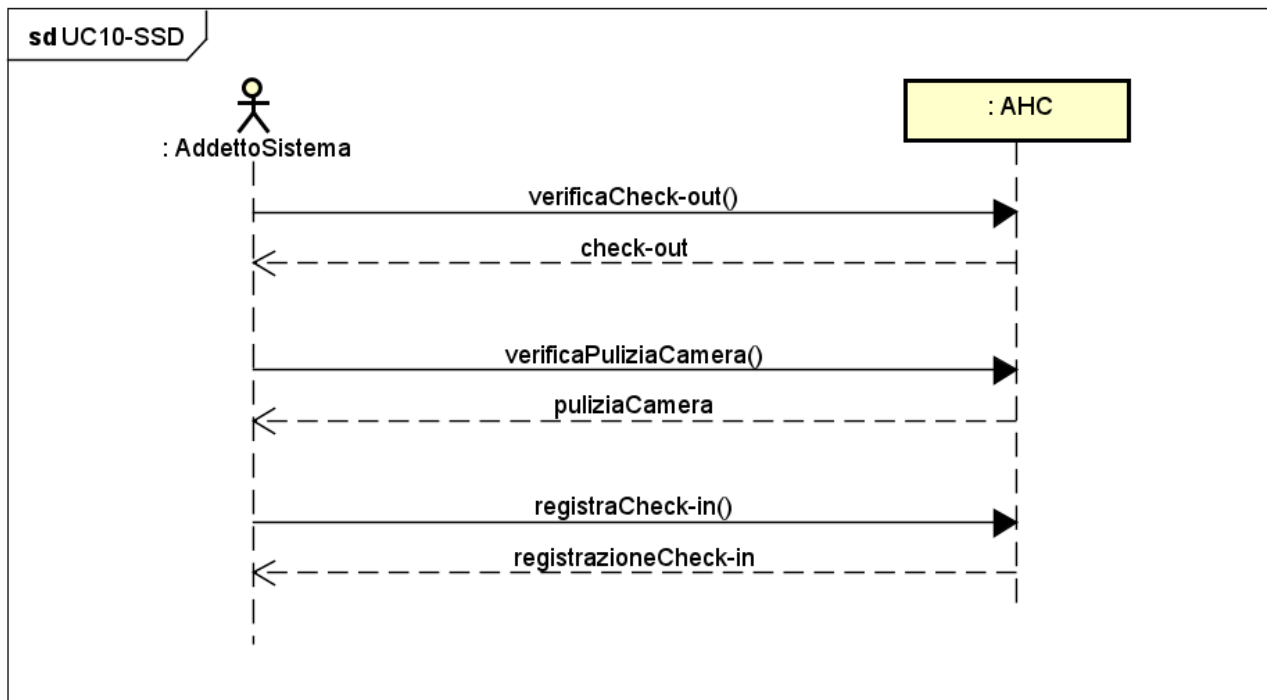
Caso d'uso UC1: Gestisci Camera



Caso d'uso UC9: Pulisci Camera



Caso d'uso UC10: Verifica disponibilità camera



2.3.3 Contratti delle operazioni

Caso d'uso UC1: Gestisci Camera

Contratto CO1: inserisciNuovoTipoCamera

Operazione	<code>inserisciNuovoTipoCamera(nome: String, numPostiLetto: int, descrizione: String)</code>
Riferimenti	Caso d'uso: Gestisci Camera
Pre-condizioni	nessuna
Post-condizioni	<ul style="list-style-type: none">- è stata creata una nuova istanza <i>tc</i> di <i>TipoCamera</i>;- gli attributi di <i>tc</i> sono stati inizializzati.

Contratto CO2: definisciPrezzoCamera

Operazione	<code>definisciPrezzoCamera(prezzo: Double)</code>
Riferimenti	Caso d'uso: Gestisci Camera
Pre-condizioni	<ul style="list-style-type: none">- è in corso la definizione del <i>TipoCamera tc</i>;

Post-condizioni	- è stata inizializzato l'attributo prezzo dell'istanza <i>tc</i> corrente di TipoCamera.
------------------------	---

Contratto CO3: confermaTipoCamera

Operazione	confermaTipoCamera()
Riferimenti	Caso d'uso: Gestisci Camera
Pre-condizioni	- è in corso la definizione del TipoCamera <i>tc</i> ;
Post-condizioni	- <i>tc</i> è stata associata al sistema AHC tramite l'associazione "registra".

Caso d'uso UC9: Pulisci Camera

Contratto CO1: visualizzaCameraDaPulire

Operazione	visualizzaCameraDaPulire()
Riferimenti	Caso d'uso: Pulisci Camera
Pre-condizioni	nessuna
Post-condizioni	<ul style="list-style-type: none"> - è stata creata una nuova istanza <i>cdp</i> di CameraDaPulire; - è stata associata <i>cdp</i> al sistema AHC tramite l'associazione "mostra".

Contratto CO2: selezionaCameraDaPulire

Operazione	selezionaCameraDaPulire(camera: CameraDaPulire)
Riferimenti	Caso d'uso: Pulisci Camera
Pre-condizioni	- è in corso la visualizzazione della CameraDaPulire;
Post-condizioni	<ul style="list-style-type: none"> - è stata creata una nuova istanza <i>cdp</i> di CameraDaPulire; - è stata creata una nuova istanza <i>ap</i> di AddettoPulizie; - è stata associata <i>cdp</i> ad <i>ap</i> tramite l'associazione "pulisce".

Contratto CO3: registraFinePulizia

Operazione	registraFinePulizia()
Riferimenti	Caso d'uso: Pulisci Camera
Pre-condizioni	- è in corso la selezione della CameraDaPulire;
Post-condizioni	- è stata creata una nuova istanza <i>fp</i> di FinePulizia; - è stata creata una nuova istanza <i>ap</i> di AddettoPulizie; - è stata associata <i>ap</i> a <i>fp</i> tramite l'associazione "comunica"; - è stata associata <i>fp</i> al sistema AHC tramite l'associazione "registra".

Caso d'uso UC10: Verifica disponibilità camera

Contratto CO1: verificaCheck-out

Operazione	verificaCheck-out()
Riferimenti	Caso d'uso: Verifica disponibilità camera
Pre-condizioni	nessuna
Post-condizioni	- è stata creata una nuova istanza <i>co</i> di Check-out;

Contratto CO2: verificaFinePulizia

Operazione	verificaPuliziaCamera()
Riferimenti	Caso d'uso: Verifica disponibilità camera
Pre-condizioni	- è stato verificato Check-out.
Post-condizioni	- è stata creata una nuova istanza <i>pc</i> di FinePulizia;

Contratto CO3: registraCheck-in

Operazione	registraCheck-in()
Riferimenti	Caso d'uso: Verifica disponibilità camera

Pre-condizioni	<ul style="list-style-type: none"> - è stato verificato Check-out; - è stato verificato FinePulizia;
Post-condizioni	<ul style="list-style-type: none"> - è stata creata una nuova istanza <i>ci</i> di Check-in; - è stata associata <i>ci</i> al sistema AHC tramite l'associazione "registra".

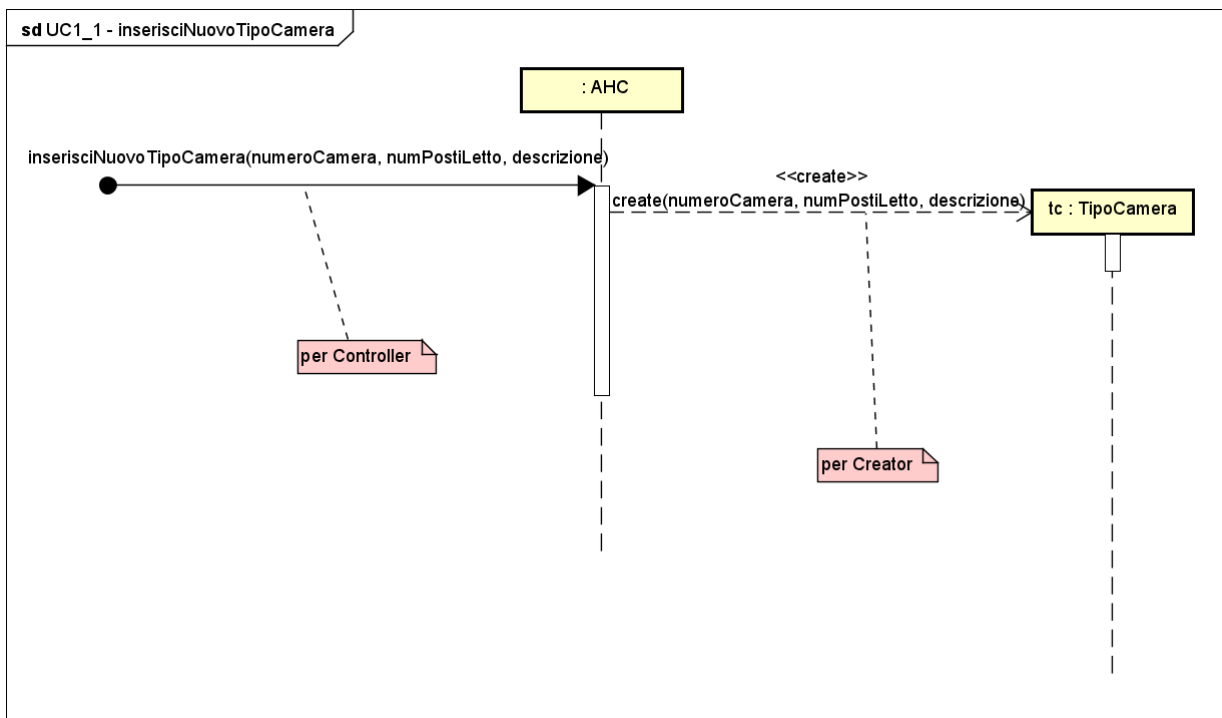
2.4 Progettazione

2.4.1 Diagramma di sequenza

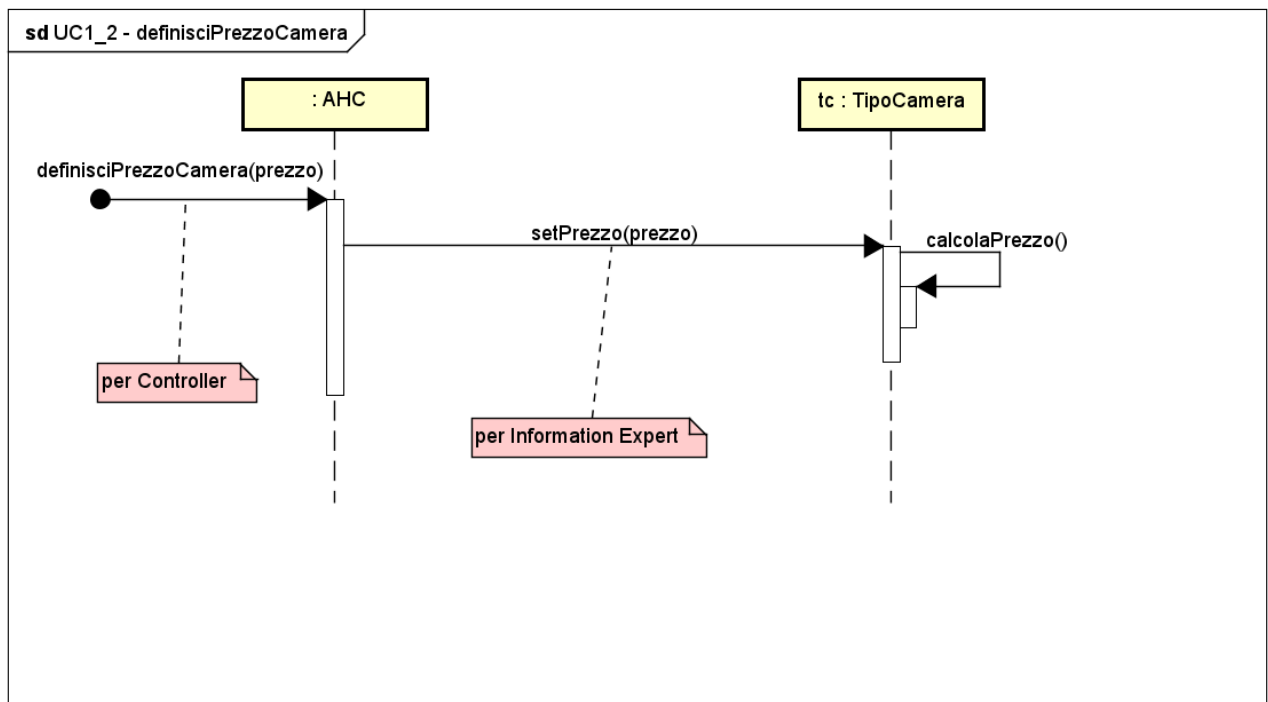
Rappresentano la progettazione logica da un punto di vista dinamico.

Caso d'uso UC1: Gestisci Camera

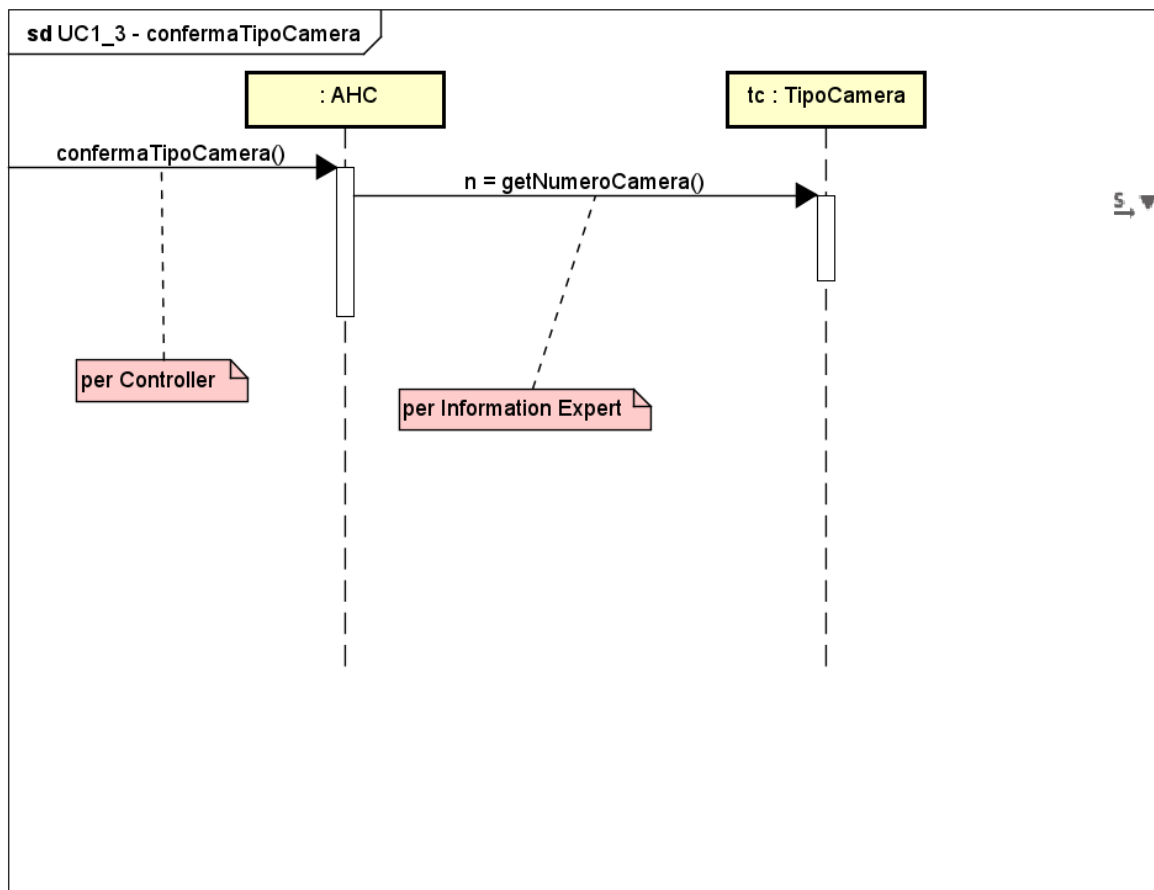
- inserisciNuovoTipoCamera



- **definisciPrezzoCamera**

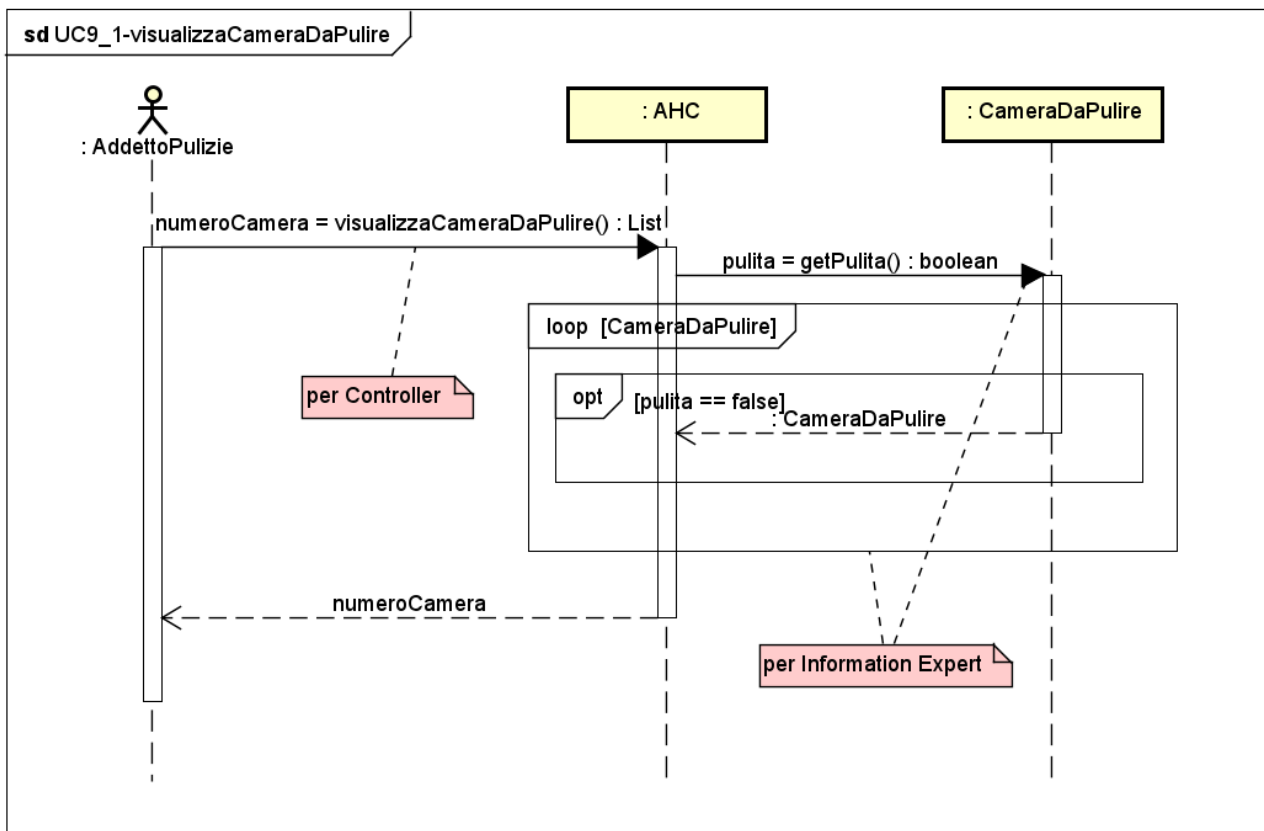


- **confermaTipoCamera**

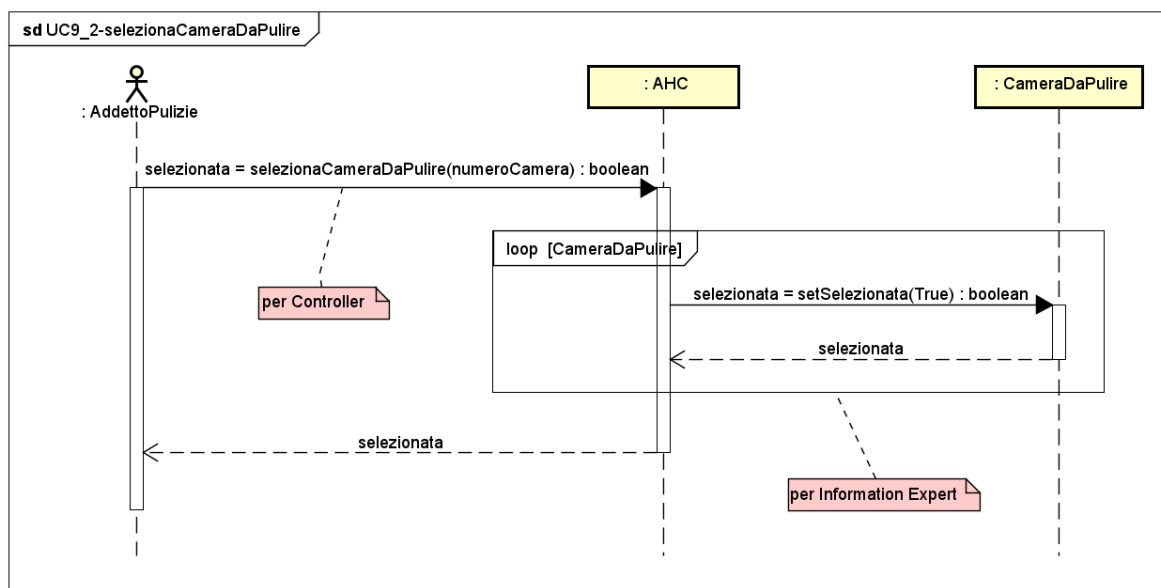


Caso d'uso UC9: Pulisci Camera

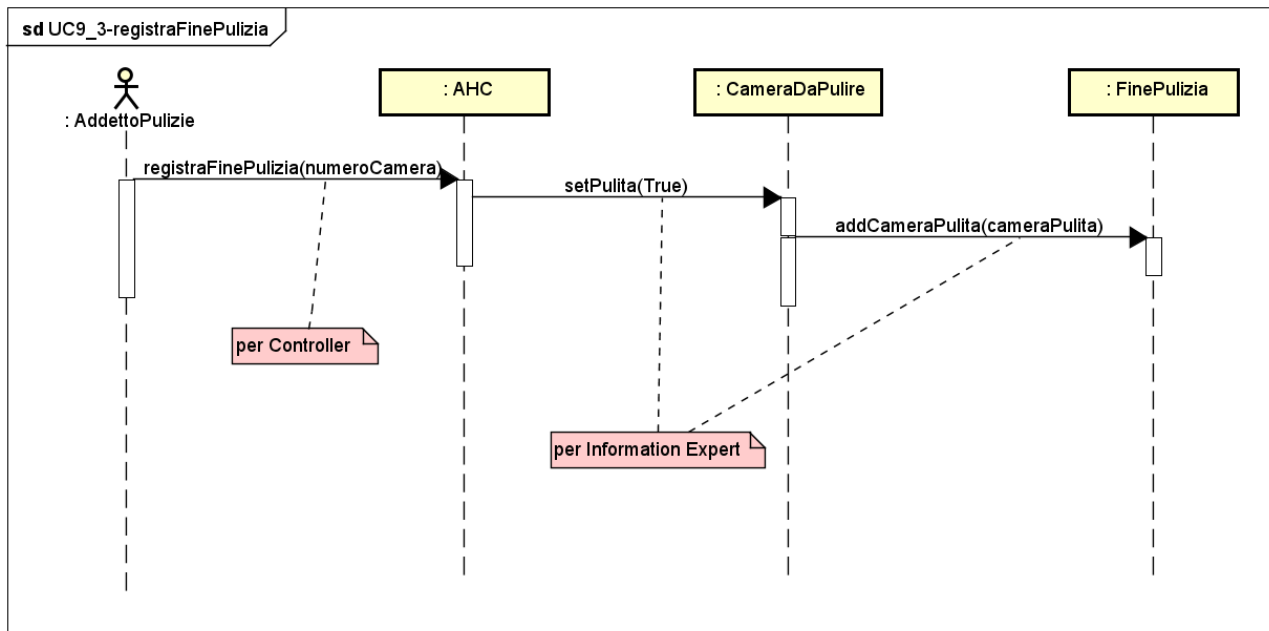
- **visualizzaCameraDaPulire**



- **selezionaCameraDaPulire**

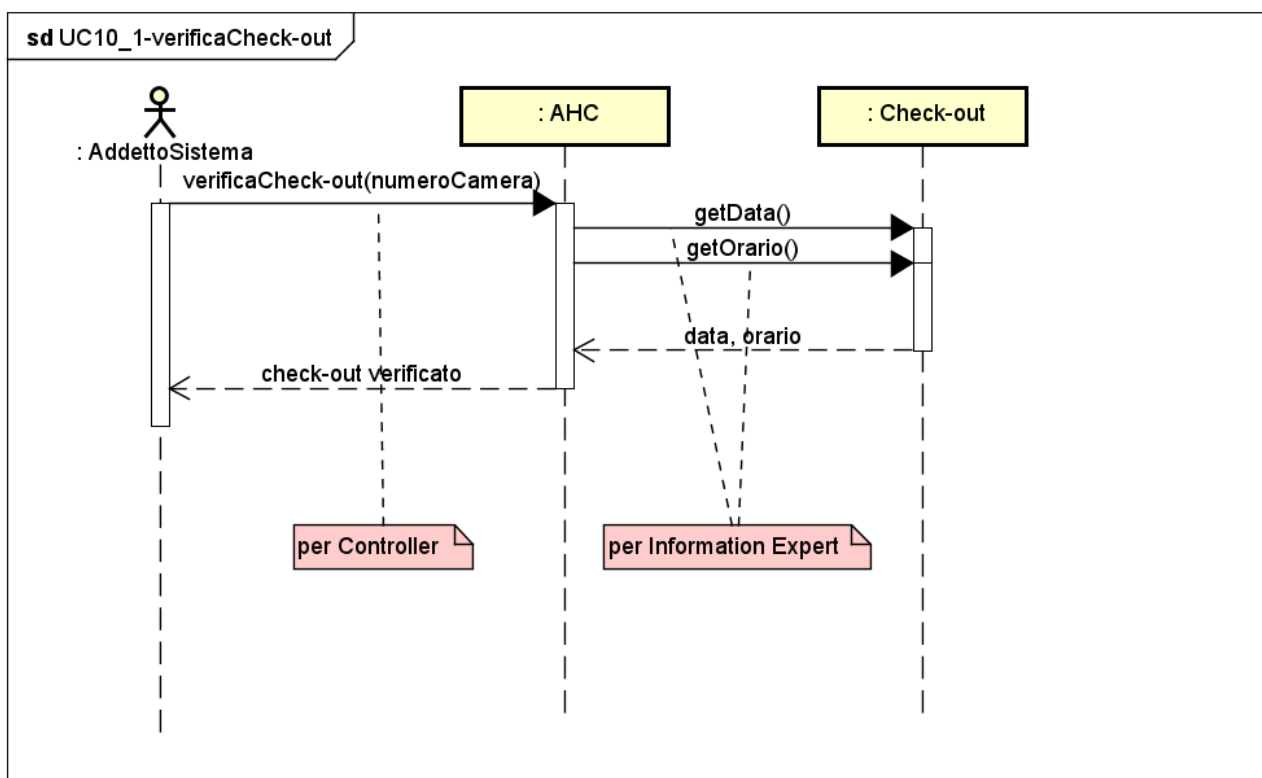


- registraFinePulizia

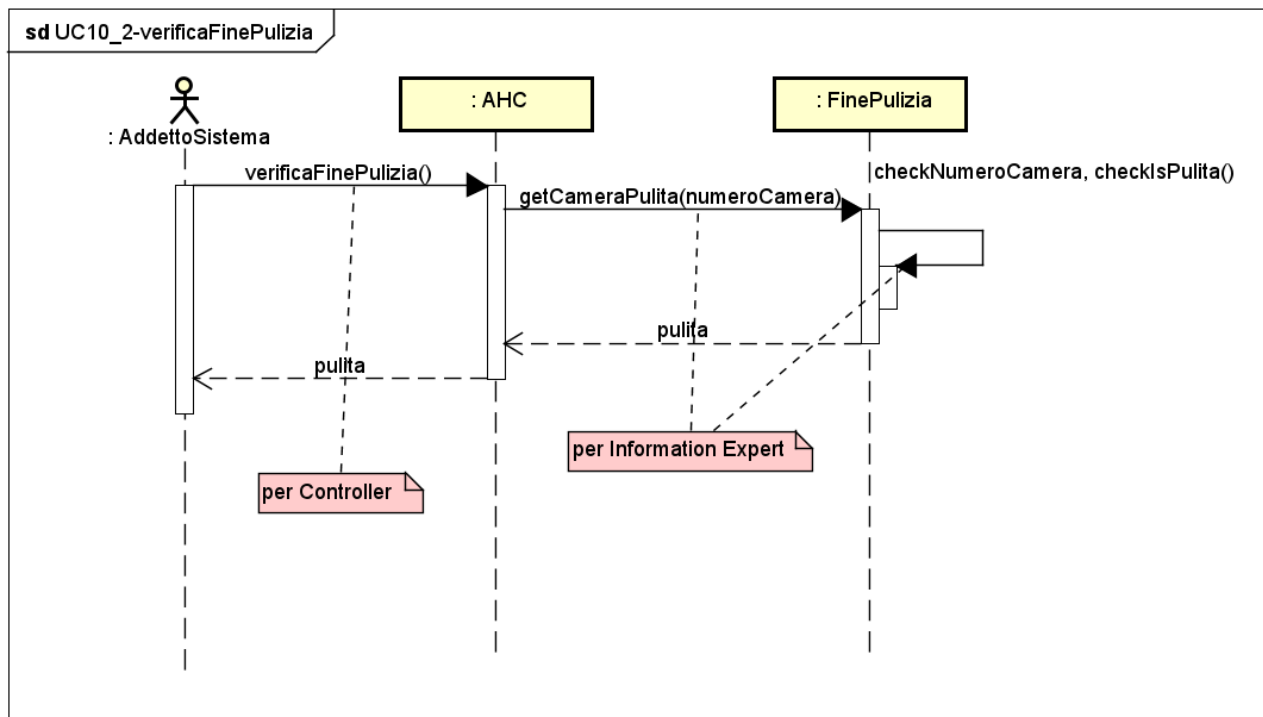


Caso d'uso UC10: Verifica disponibilità camera

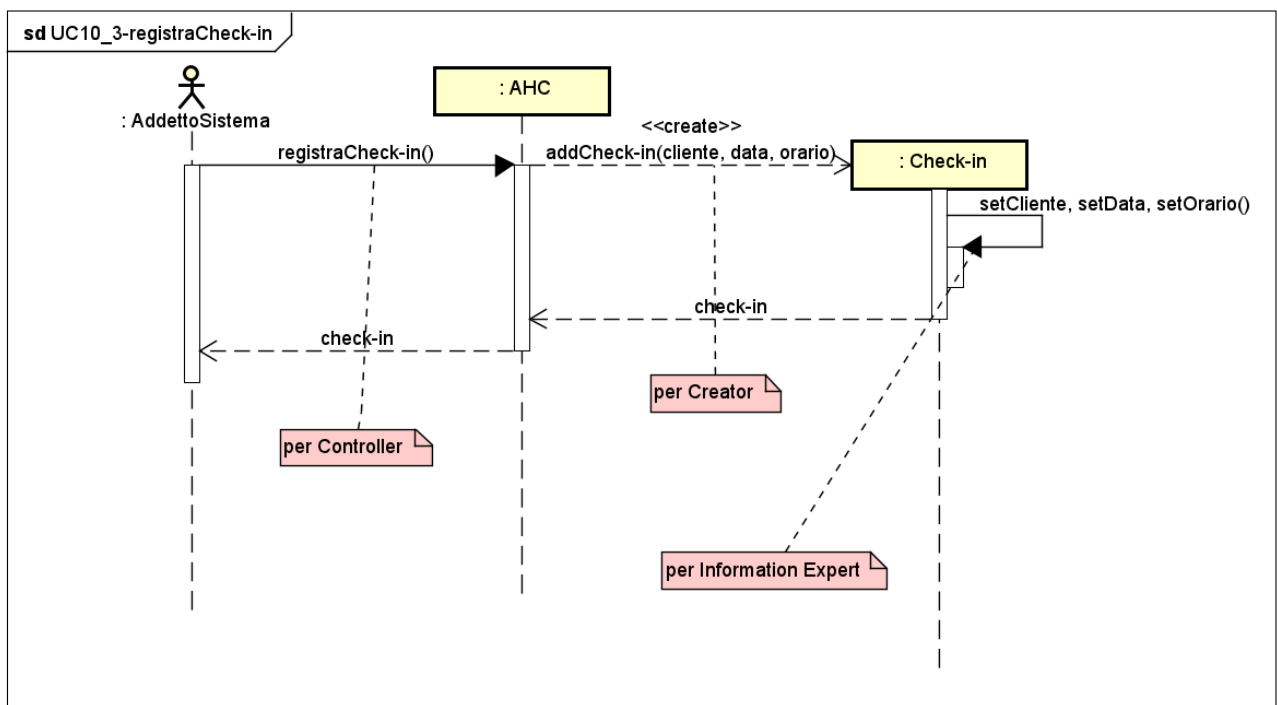
- verificaCheck-out



- **verificaFinePulizia**



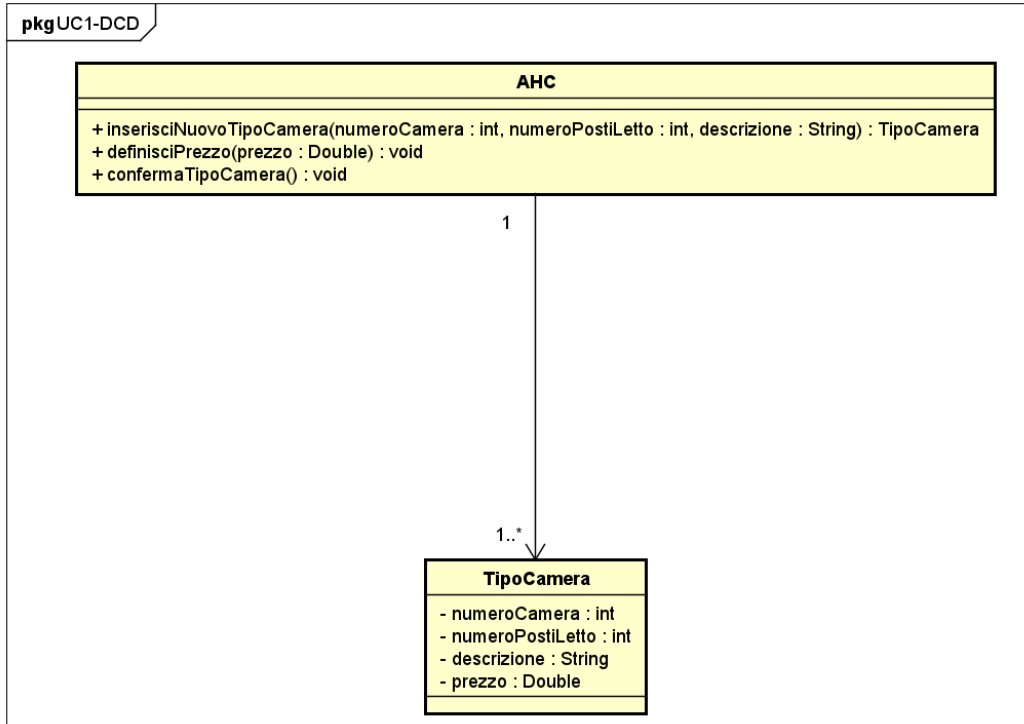
- **registraCheck-in**



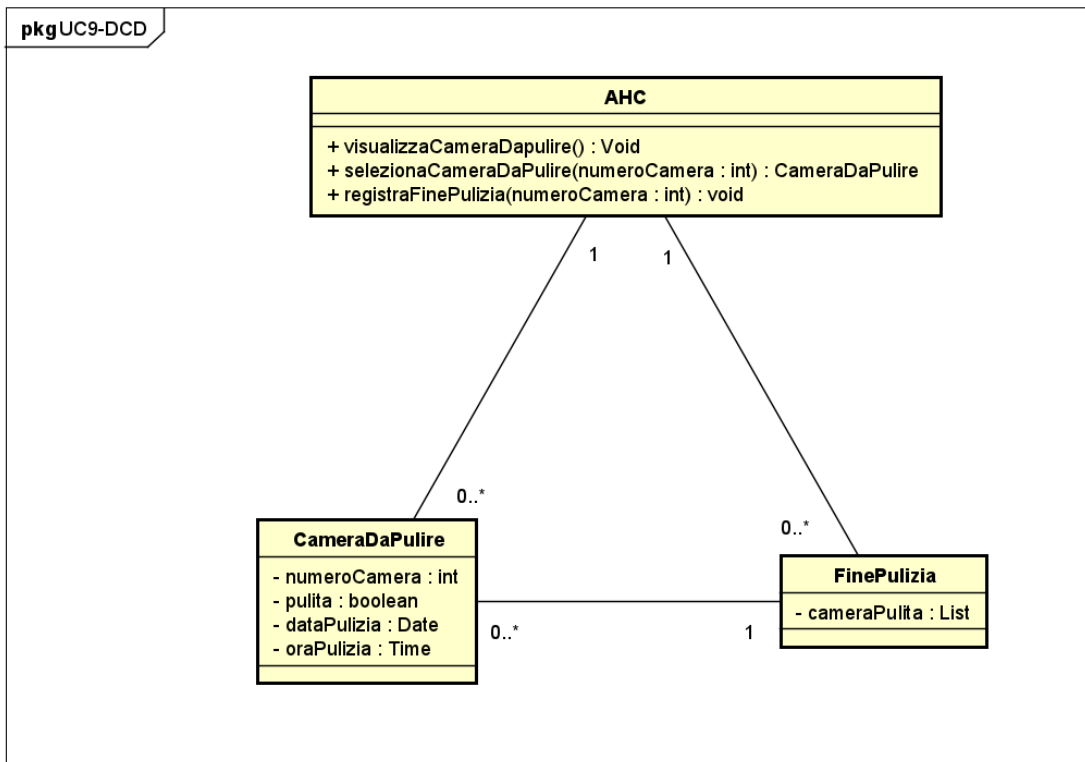
2.4.2 Diagramma delle classi

Rappresentano la progettazione logica da un punto di vista statico.

Caso d'uso UC1: Gestisci Camera



Caso d'uso UC9: Pulisci Camera



Caso d'uso UC10: Verifica disponibilità camera

