[3 Elaborazione – Iterazione 2 3](#_Toc77248212)

[3.1 Introduzione 3](#_Toc77248213)

[3.2 Aggiornamento casi d’uso UC2, UC6 e UC11 3](#_Toc77248214)

[UC2: Gestisci Campeggio 3](#_Toc77248215)

[UC6: Gestisci Cliente 4](#_Toc77248216)

[UC11: Registra Check-in 5](#_Toc77248217)

[3.3 Analisi orientata agli oggetti 6](#_Toc77248218)

[3.3.1 Modello di dominio 6](#_Toc77248219)

[Caso d’uso UC3: Effettua Prenotazione 6](#_Toc77248220)

[Caso d’uso UC7: Effettua Pagamento 7](#_Toc77248221)

[3.3.2 Diagramma di sequenza di sistema 9](#_Toc77248222)

[Caso d’uso UC3: Effettua Prenotazione 9](#_Toc77248223)

[Caso d’uso UC7: Effettua Pagamento 9](#_Toc77248224)

[3.3.3 Contratti delle operazioni 10](#_Toc77248225)

[Caso d’uso UC3: Effettua Prenotazione 10](#_Toc77248226)

[Contratto CO1: richiestaPrenotazione 10](#_Toc77248227)

[Contratto CO2: inserisciDatiSoggiorno 10](#_Toc77248228)

[Caso d’uso UC7: Effettua Pagamento 10](#_Toc77248229)

[Contratto CO1: richiestaPagamento 10](#_Toc77248230)

[Contratto CO2: scegliModalitaPagamento 11](#_Toc77248231)

[Contratto CO3: inserisciDatiCarta 11](#_Toc77248232)

[Contratto CO4: inserisciDatiBonifico 11](#_Toc77248233)

[3.4 Progettazione 12](#_Toc77248234)

[3.4.1 Diagramma di sequenza 12](#_Toc77248235)

[Caso d’uso UC3: Effettua Prenotazione 12](#_Toc77248236)

[Caso d’uso UC7: Effettua Pagamento 13](#_Toc77248237)

[3.4.2 Diagramma delle classi 16](#_Toc77248238)

[Caso d’uso UC3: Effettua Prenotazione 16](#_Toc77248239)

[Caso d’uso UC7: Effettua Pagamento 17](#_Toc77248240)

# 3 Elaborazione – Iterazione 2

## 3.1 Introduzione

La prima iterazione si è conclusa, si passerà adesso alla seconda fase di elaborazione in cui si andrà ad approfondire la conoscenza di altri casi d’uso non ancora analizzati a fondo e si andranno a risolvere le problematiche relative ai rischi maggiori.

Per questa seconda 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.

## 3.2 Aggiornamento casi d’uso UC2, UC6 e UC11

Vengono rappresentati in formato esteso i casi d’uso UC2, UC6, UC11, scoperti durante la fase di ideazione. Per le due successive fasi di Analisi orientata agli oggetti e Progettazione verranno trattati  
i casi d’uso UC3, UC7 ed UC11.

### UC2: Gestisci Campeggio

|  |  |
| --- | --- |
| **Nome caso d’uso** | UC2: Gestisci Campeggio |
| **Portata** | Applicazione Amazing Hotel & Camper |
| **Livello** | Obiettivo utente |
| **Attore primario** | Addetto del sistema |
| **Parti interessate e interessi** | * *Addetto del sistema*: Vuole gestire l’intero processo di creazione dei campeggi in maniera corretta e veloce; vuole che le informazioni relative alle prenotazioni dei clienti siano aggiornate. * *Cliente*: Vuole conoscere i campeggi presenti nella struttura per effettuare le prenotazioni degli stessi, usufruendo di un servizio rapido e semplice. |
| **Pre-condizione** | L’Addetto è identificato e autenticato. |
| **Garanzie di successo** | Il campeggio viene inserito. Il cliente può procedere alla sua prenotazione. |
| **Scenario principale di successo** | 1. L’Addetto richiede al sistema l’inserimento di un nuovo tipo di campeggio. 2. Il sistema richiede all’Addetto le informazioni relative al campeggio. 3. L’Addetto inserisce i dati del nuovo campeggio (numero del campeggio, numero dei posti sosta, descrizione). 4. L’Addetto inserisce il prezzo base per il nuovo tipo di campeggio. 5. Il sistema mostra i dati relativi al nuovo tipo di campeggio immesso. 6. L’Addetto conferma i dati immessi. Il sistema registra i dati notifica l’inserimento avvenuto con successo. |
| **Estensioni** | \*a. In qualsiasi momento, il sistema fallisce e ha un arresto improvviso.   1. L’Addetto del sistema riavvia il software e richiede il ripristino dello stato precedente del sistema. 2. Il sistema ricostruisce lo stato precedente.   1a. L’Addetto richiede al sistema di modificare un campeggio esistente.   1. L’Addetto richiede al sistema la ricerca di un campeggio tramite l’inserimento di un parametro (numero identificativo). 2. Il sistema mostra l’elenco dei risultati della ricerca. 3. L’addetto seleziona il campeggio e inserisce i dati da modificare. 4. Il sistema notifica la modifica avvenuta.   2a. L’Addetto richiede al sistema di eliminare un campeggio esistente   1. L’Addetto richiede al sistema la ricerca di un campeggio tramite l’inserimento di un parametro (numero identificativo). 2. Il sistema mostra l’elenco dei risultati della ricerca. 3. L’addetto seleziona il campeggio da eliminare. 4. Il sistema notifica l’eliminazione avvenuta.   3a. L’Addetto non vuole proseguire con l’inserimento/modifica/eliminazione del campeggio.   1. L’Addetto termina l’esecuzione del programma. |
| **Requisiti speciali** |  |
| **Elenco delle variabili tecnologiche e dei dati** |  |
| **Frequenze di ripetizioni** | Legata al numero di campeggi presenti in struttura e alla loro possibile modifica, eliminazione. |
| **Varie** |  |

### UC6: Gestisci Cliente

|  |  |
| --- | --- |
| **Nome caso d’uso** | UC6: Gestisci Cliente |
| **Portata** | Applicazione Amazing Hotel & Camper |
| **Livello** | Obiettivo utente |
| **Attore primario** | Addetto del sistema |
| **Parti interessate e interessi** | * *Addetto del sistema*: Vuole gestire l’intero processo di gestione dei clienti in maniera corretta e veloce; vuole che le informazioni relative alle informazioni dei clienti siano aggiornate. |
| **Pre-condizione** | L’addetto del sistema è identificato e autenticato. |
| **Garanzie di successo** | Il cliente viene inserito. Possono essere effettuate le operazioni di check-in/ check-out. |
| **Scenario principale di successo** | 1. L’Addetto richiede al sistema l’inserimento di un nuovo Cliente. 2. Il sistema richiede all’Addetto le informazioni relative al Cliente. 3. L’Addetto inserisce i dati del nuovo cliente (nome, cognome, CF). 4. Il sistema mostra i dati relativi al nuovo cliente immessi. 5. L’Addetto conferma i dati immessi. 6. Il sistema registra i dati e notifica l’inserimento avvenuto con successo. |
| **Estensioni** | \*a. In qualsiasi momento, il sistema fallisce e ha un arresto improvviso.   1. L’Addetto delle pulizie riavvia il software e richiede il ripristino dello stato precedente del sistema. 2. Il sistema ricostruisce lo stato precedente.   1a. L’Addetto richiede al sistema di modificare un cliente esistente.   1. L’Addetto richiede al sistema la ricerca di un cliente tramite l’inserimento di un parametro (CF). 2. Il sistema mostra l’elenco dei risultati della ricerca. 3. L’addetto seleziona il cliente e inserisce i dati da modificare. 4. Il sistema notifica la modifica avvenuta.   2a. L’Addetto richiede al sistema di eliminare un cliente esistente.   1. L’Addetto richiede al sistema la ricerca di un cliente tramite l’inserimento di un parametro (CF). 2. Il sistema mostra l’elenco dei risultati della ricerca. 3. L’addetto seleziona il cliente da eliminare. 4. Il sistema notifica l’eliminazione avvenuta.   3a. L’Addetto non vuole proseguire con l’inserimento/modifica/eliminazione del cliente.   1. L’Addetto termina l’esecuzione del programma. |
| **Requisiti speciali** |  |
| **Elenco delle variabili tecnologiche e dei dati** |  |
| **Frequenze di ripetizioni** | Legata al numero di clienti che vogliono usufruire dei servizi offerti dalla struttura e alla loro possibile modifica, eliminazione. |
| **Varie** |  |

### UC11: Registra Check-in

|  |  |
| --- | --- |
| **Nome caso d’uso** | UC11: Registra Check-in |
| **Portata** | Applicazione Amazing Hotel & Camper |
| **Livello** | Obiettivo utente |
| **Attore primario** | Addetto del sistema |
| **Parti interessate e interessi** | * *Addetto del sistema*: Vuole gestire l’intero processo di accoglienza (tramite la registrazione del check-in) dei clienti in maniera corretta e veloce; vuole che le informazioni relative alle disponibilità delle camere siano aggiornate. * *Cliente*: Vuole ricevere un’accoglienza adeguata dalla struttura per il servizio richiesto, in maniera rapida e semplice. Vuole essere registrato per poter usufruire del servizio richiesto. |
| **Pre-condizione** | L’addetto è identificato e autenticato. |
| **Garanzie di successo** | Il check-in viene registrato. Il cliente può usufruire del soggiorno prenotato. |
| **Scenario principale di successo** | 1. Il Cliente si reca presso l’accettazione della struttura e si rivolge all’Addetto del sistema richiedendo l’ingresso alla camera (già prenotata), fornendo il suo nominativo (nome, cognome e CF). 2. L’Addetto verifica la presenza della prenotazione associata al nominativo del cliente che ha fatto la richiesta. 3. L’Addetto inserisce i dati del Cliente e la data e l’orario dell’ingresso (check-in) tramite il sistema. 4. Il sistema notifica la registrazione del check-in avvenuta con successo. 5. L’Addetto informa il Cliente della registrazione avvenuta, fornisce la chiave della camera al cliente. 6. Il cliente viene accompagnato nella sua camera e ne prende il possesso. |
| **Estensioni** | \*a. In qualsiasi momento, il sistema fallisce e ha un arresto improvviso.   1. L’Addetto del sistema riavvia il software e richiede il ripristino dello stato precedente del sistema. 2. Il sistema ricostruisce lo stato precedente.   1a. Il nominativo del cliente non corrisponde con nessuna prenotazione registrata nel sistema.   1. L’addetto del sistema comunica il disguido al cliente. 2. L’addetto del sistema provvede alle operazioni necessarie per capire le motivazioni del problema. |
| **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** |  |

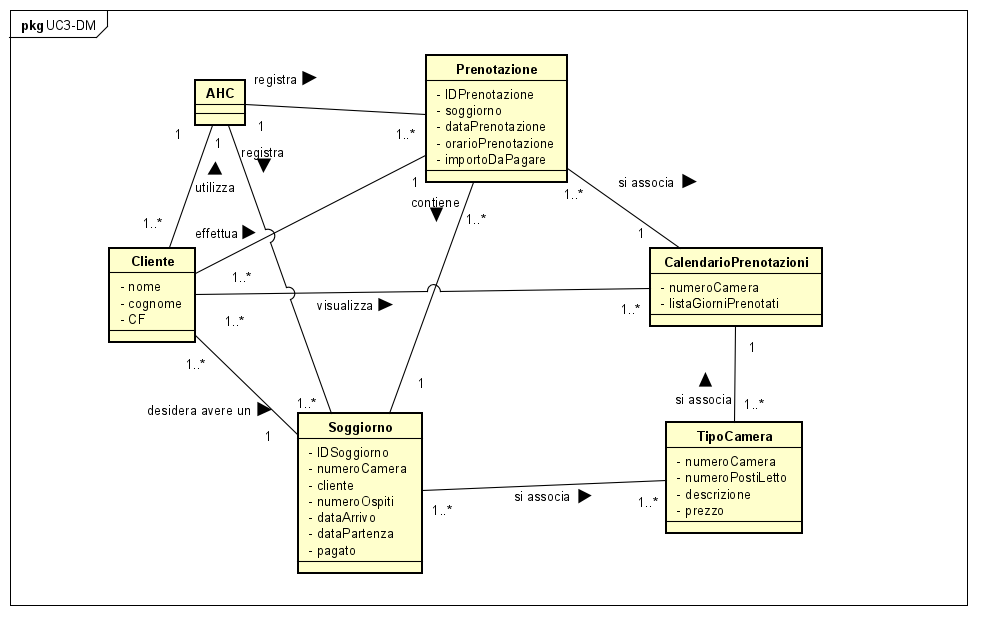
## 3.3 Analisi orientata agli oggetti

### 3.3.1 Modello di dominio

#### Caso d’uso UC3: Effettua Prenotazione

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

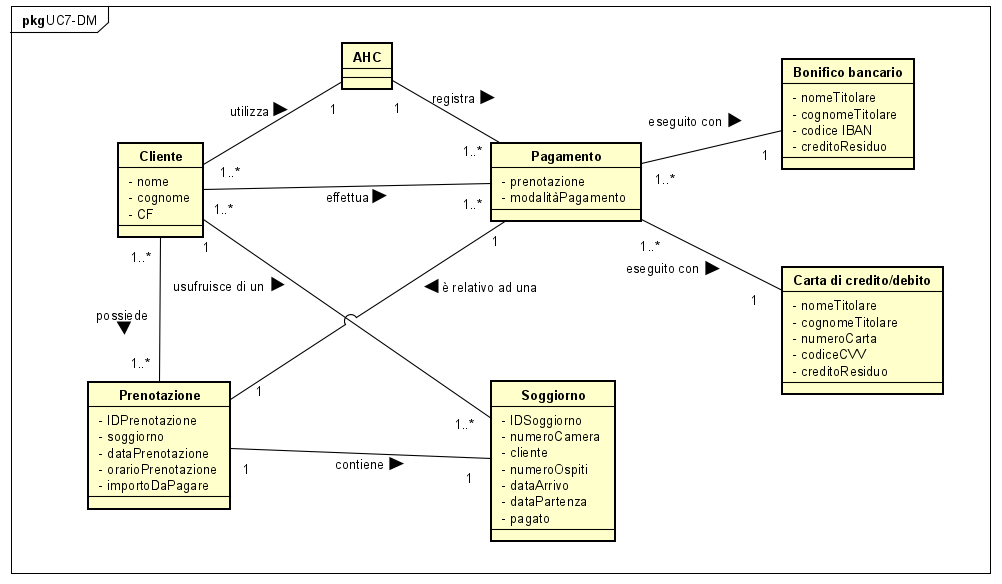
* *Cliente*: attore primario, che interagisce direttamente col sistema;
* *AHC*: rappresenta il sistema Amazing hotel & Camper;
* *Tipo Camera*: discrimina il tipo di camera grazie al numero di posti letto tra quelle disponibili in struttura, e contiene una descrizione;
* *Soggiorno*: Indica la permanenza per un periodo di tempo ben preciso in un determinato luogo della struttura da parte del Cliente;
* *Prenotazione*: Indica l’atto di richiesta di un Cliente di aver riservato un posto e un servizio nella struttura ricettiva;
* *Calendario Prenotazioni*: indica la lista delle prenotazioni effettuate per una specifica camera.

**

#### Caso d’uso UC7: Effettua Pagamento

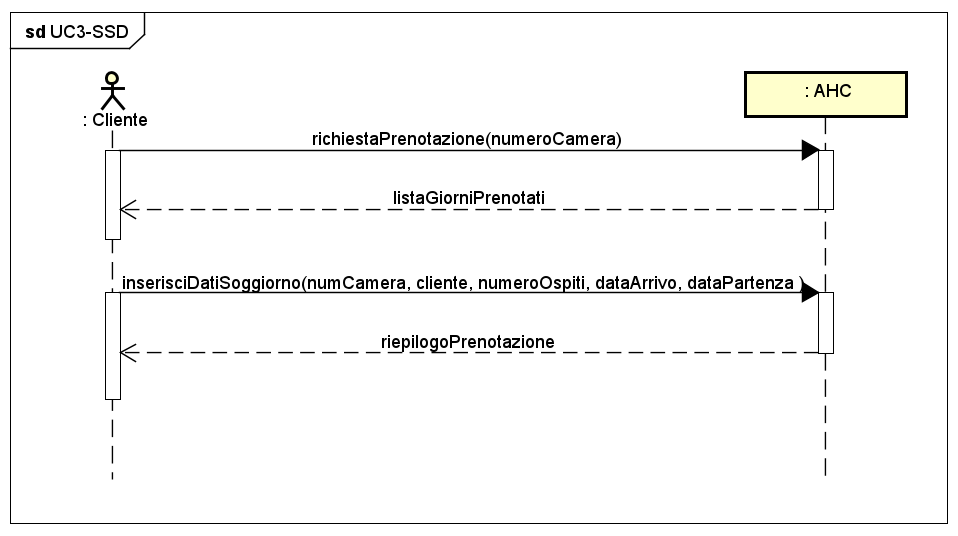
Valutato lo scenario principale di successo sono state identificate le seguenti classi concettuali:

* *Cliente*: attore primario, che interagisce direttamente col sistema;
* *AHC*: rappresenta il sistema Amazing hotel & Camper;
* *Soggiorno*: Indica la permanenza per un periodo di tempo ben preciso in un determinato luogo della struttura da parte del Cliente;
* *Prenotazione*: Indica l’atto di richiesta di un Cliente di aver riservato un posto e un servizio nella struttura ricettiva;
* *Pagamento:* Indica il trasferimento di denaro, dal Cliente alla struttura, come corrispettivo di un servizio ricevuto;
* *Carta di credito/debito, Bonifico bancario:* indicano delle modalità di pagamento previste per il trasferimento di denaro dal cliente alla struttura.

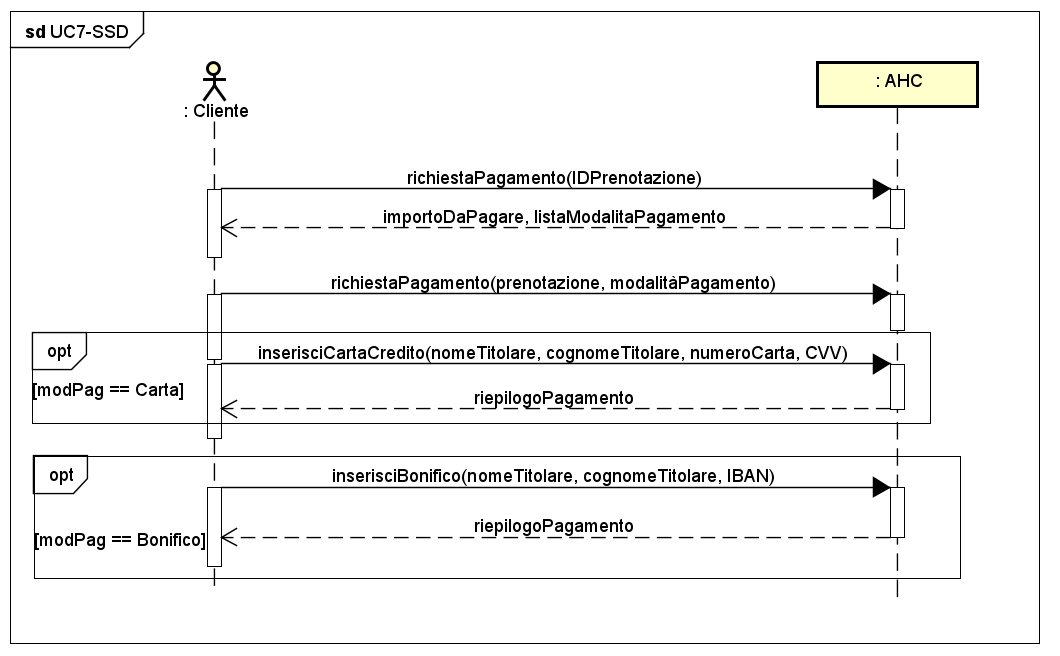


### 3.3.2 Diagramma di sequenza di sistema

#### Caso d’uso UC3: Effettua Prenotazione



#### Caso d’uso UC7: Effettua Pagamento



### 3.3.3 Contratti delle operazioni

#### Caso d’uso UC3: Effettua Prenotazione

##### Contratto CO1: richiestaPrenotazione

|  |  |
| --- | --- |
| **Operazione** | richiestaPrenotazione (numeroCamera: int) |
| **Riferimenti** | Caso d’uso: Effettua Prenotazione |
| **Pre-condizioni** | nessuna |
| **Post-condizioni** | è stato visualizzato il calendario dei giorni prenotati *CalendarioPrenotazioni* *cp.* |

##### Contratto CO2: inserisciDatiSoggiorno

|  |  |
| --- | --- |
| **Operazione** | inserisciDatiSoggiorno ( int: numeroCamera, cliente: Cliente, numOspiti: int, dataArrivo: Date, dataPartenza: Date) |
| **Riferimenti** | Caso d’uso: Effettua Prenotazione |
| **Pre-condizioni** | è in corso una richiesta prenotazione p. |
| **Post-condizioni** | * è stata creata una nuova istanza *s* di Soggiorno (creazione di oggetto); * è stata creata una nuova istanza *p* di Prenotazione (creazione di oggetto); * è stata associata *p* a *s* tramite l’associazione “contiene” (formazione di collegamento); * s.pagato è diventato true (modifica di attributo). * è stata creata una nuova istanza *cp* di CalendarioPrenotazioni (creazione di oggetto); * è stata associata *p* a *cp* tramite l’associazione “si associa” (formazione di collegamento). |

#### Caso d’uso UC7: Effettua Pagamento

##### Contratto CO1: richiestaPagamento

|  |  |
| --- | --- |
| **Operazione** | richiestaPagamento ( int: idPrenotazione) |
| **Riferimenti** | Caso d’uso: Effettua Pagamento |
| **Pre-condizioni** | nessuna |
| **Post-condizioni** | * è stato visualizzato l’importo da pagare relativo all’istanza *Prenotazione* p creata nell’operazione CO2 del caso d’uso precedente; * è stato visualizzata la lista dei giorni prenotati relativi all’istanza *CalendarioPrenotazioni* *cp* creata nell’operazione CO2 del caso d’uso precedente. |

##### Contratto CO2: scegliModalitaPagamento

|  |  |
| --- | --- |
| **Operazione** | scegliModalitaPagamento (Prenotazione: prenotazione, String: modalitaPagamento, CartaCredito\_Debito: carta, BonificoBancario: bonifico) |
| **Riferimenti** | Caso d’uso: Effettua Pagamento |
| **Pre-condizioni** | * è in corso la richiesta di un Pagamento; |
| **Post-condizioni** | * è stata creata una nuova istanza *p* di *Pagamento* (creazione di oggetto). |

##### Contratto CO3: inserisciDatiCarta

|  |  |
| --- | --- |
| **Operazione** | inserisciDatiCarta (int: numeroCarta, String: nome, String: cognome, int: codiceCVV) |
| **Riferimenti** | Caso d’uso: Effettua Pagamento |
| **Pre-condizioni** | * è in corso un *Pagamento* *p*; |
| **Post-condizioni** | * è stata creata una nuova istanza *cc* di *CartaCredito* (creazione di oggetto); * è stata associata *p* di Pagamento a *cc* tramite l’associazione “eseguito con” (formazione di collegamento). |

##### Contratto CO4: inserisciDatiBonifico

|  |  |
| --- | --- |
| **Operazione** | *inserisciDatiBonifico* ( String: nome, String: cognome, String : IBAN) |
| **Riferimenti** | Caso d’uso: Effettua Pagamento |
| **Pre-condizioni** | * è in corso un *Pagamento* *p*; |
| **Post-condizioni** | * è stata creata una nuova istanza *bb* di *BonificoBancario (*creazione di oggetto); * è stata associata *p* di *Pagamento* a *bb* tramite l’associazione “eseguito con” (formazione di collegamento). |

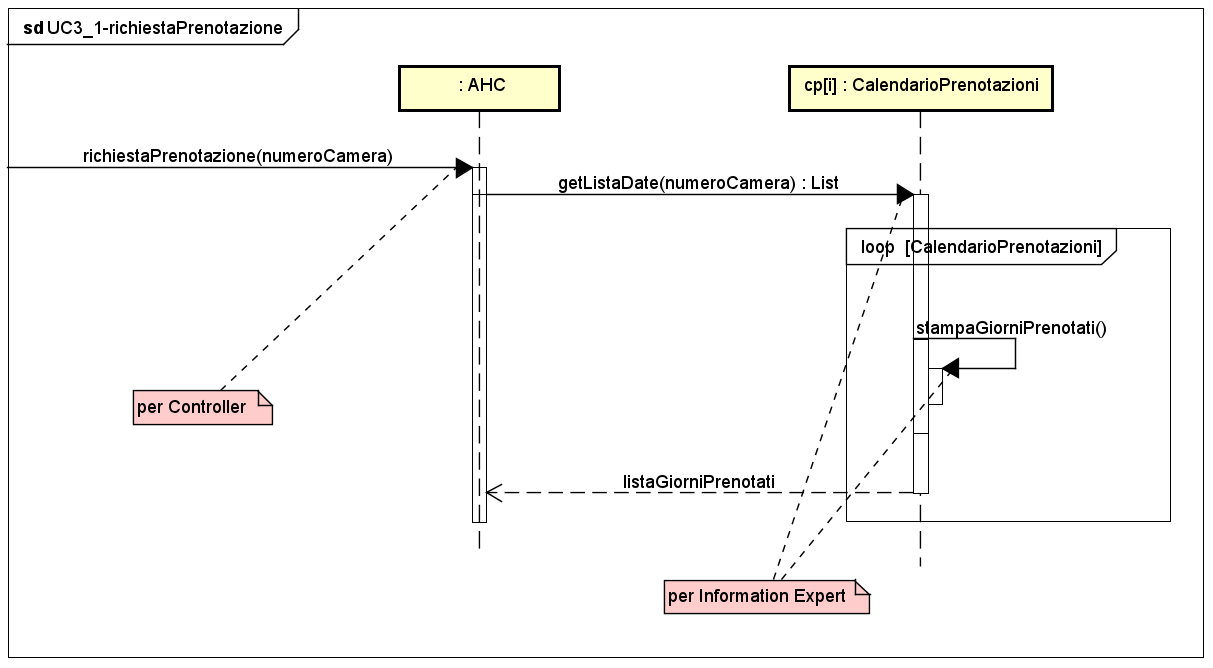
## 3.4 Progettazione

### 3.4.1 Diagramma di sequenza

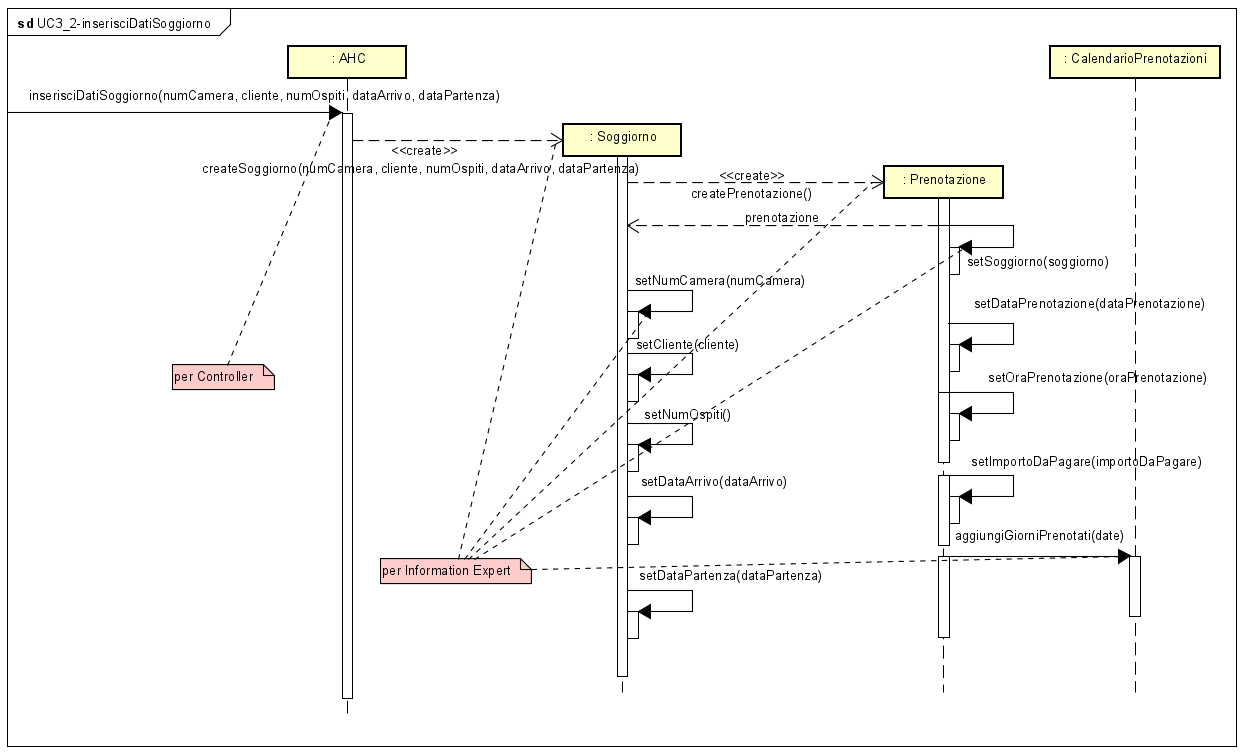
Si rappresenta, in questa fase, la progettazione logica da un punto di vista dinamico.

#### Caso d’uso UC3: Effettua Prenotazione

* **richiestaPrenotazione**

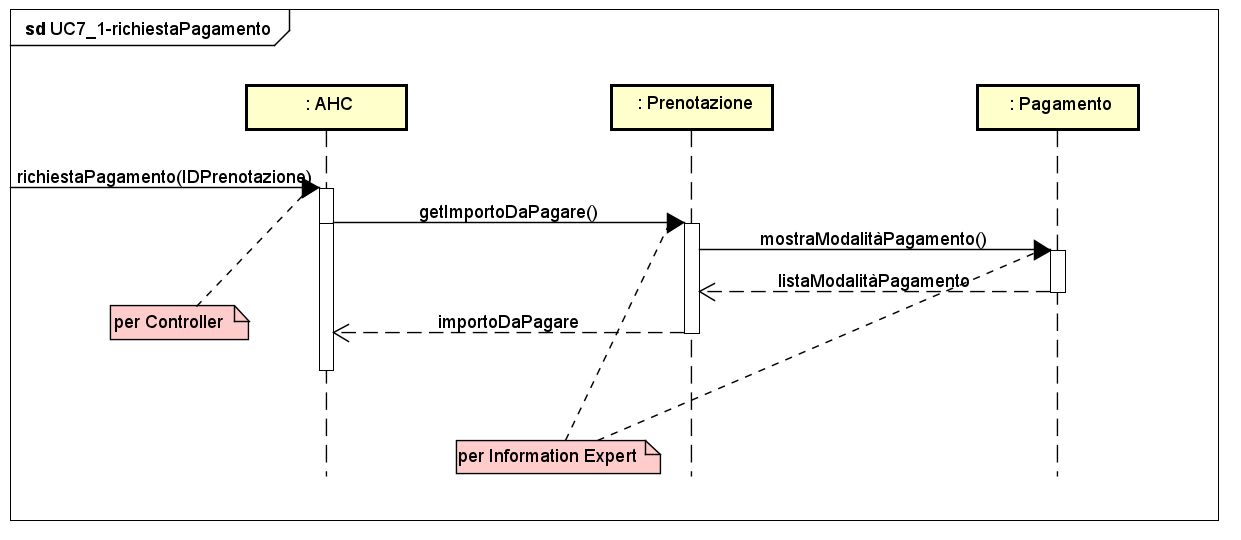
****

* **inserisciDatiSoggiorno**

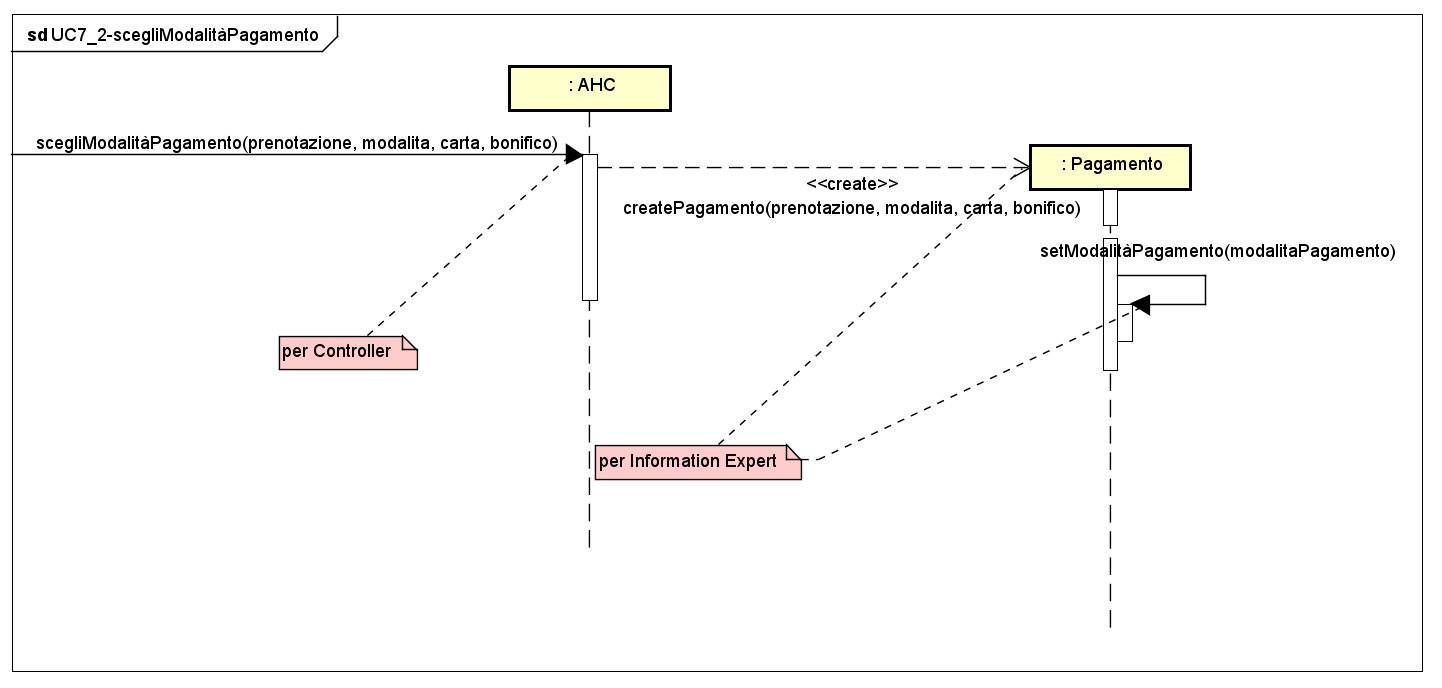
****

#### Caso d’uso UC7: Effettua Pagamento

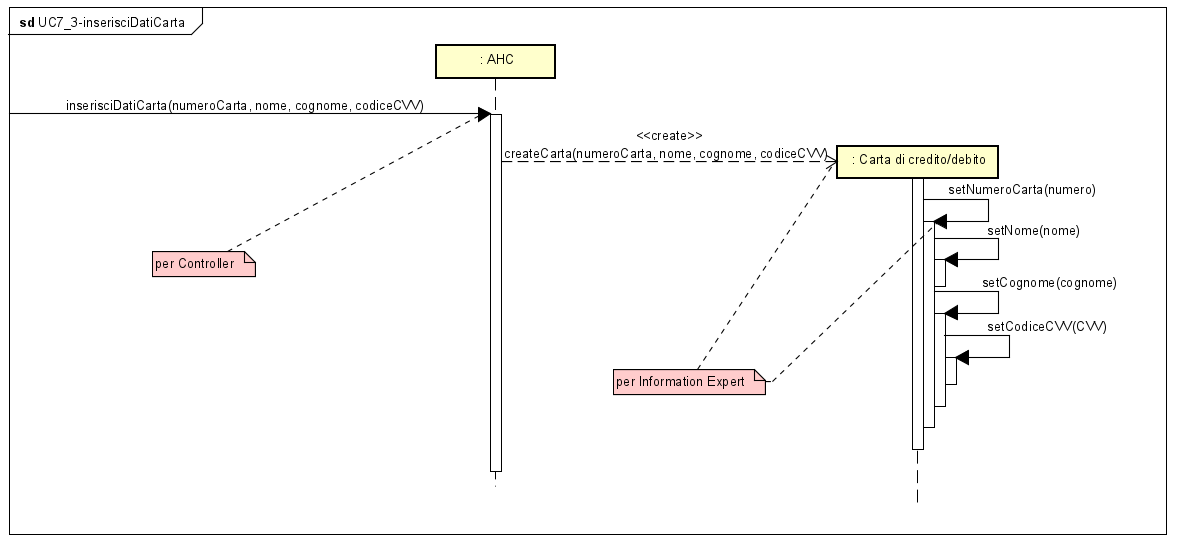
* **richiestaPagamento**



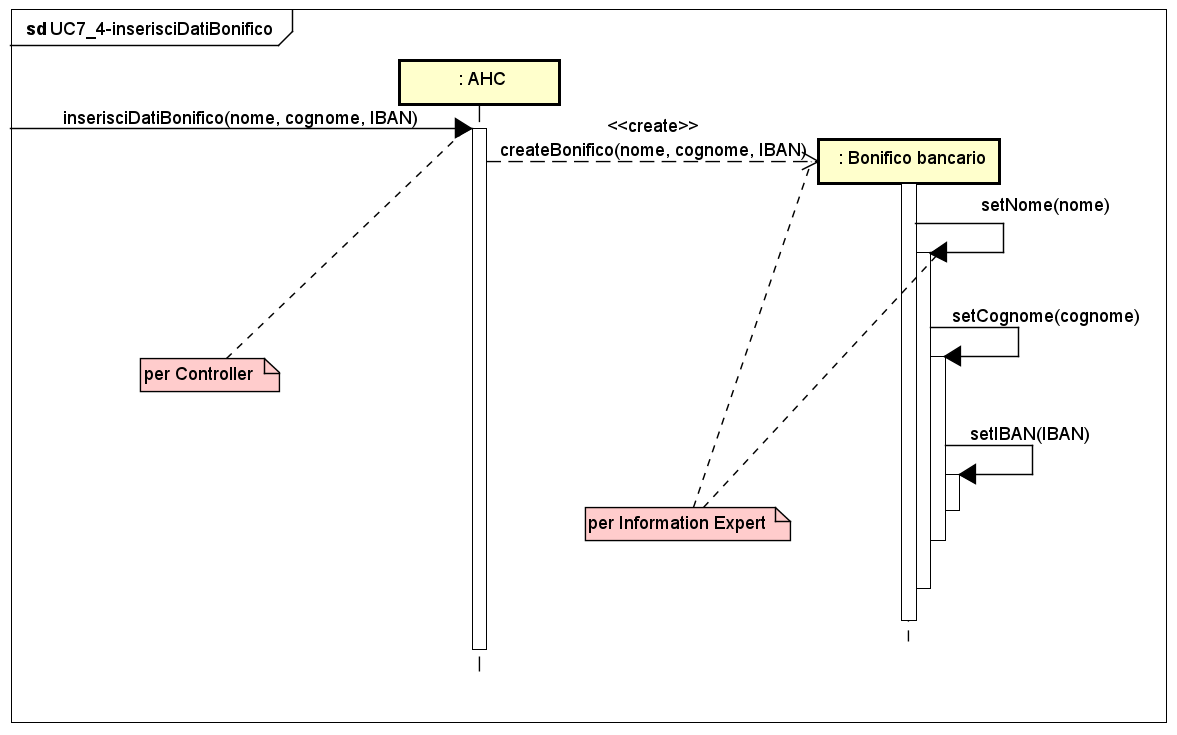
* **scegliModalitàPagamento**



* **inserisciDatiCarta**



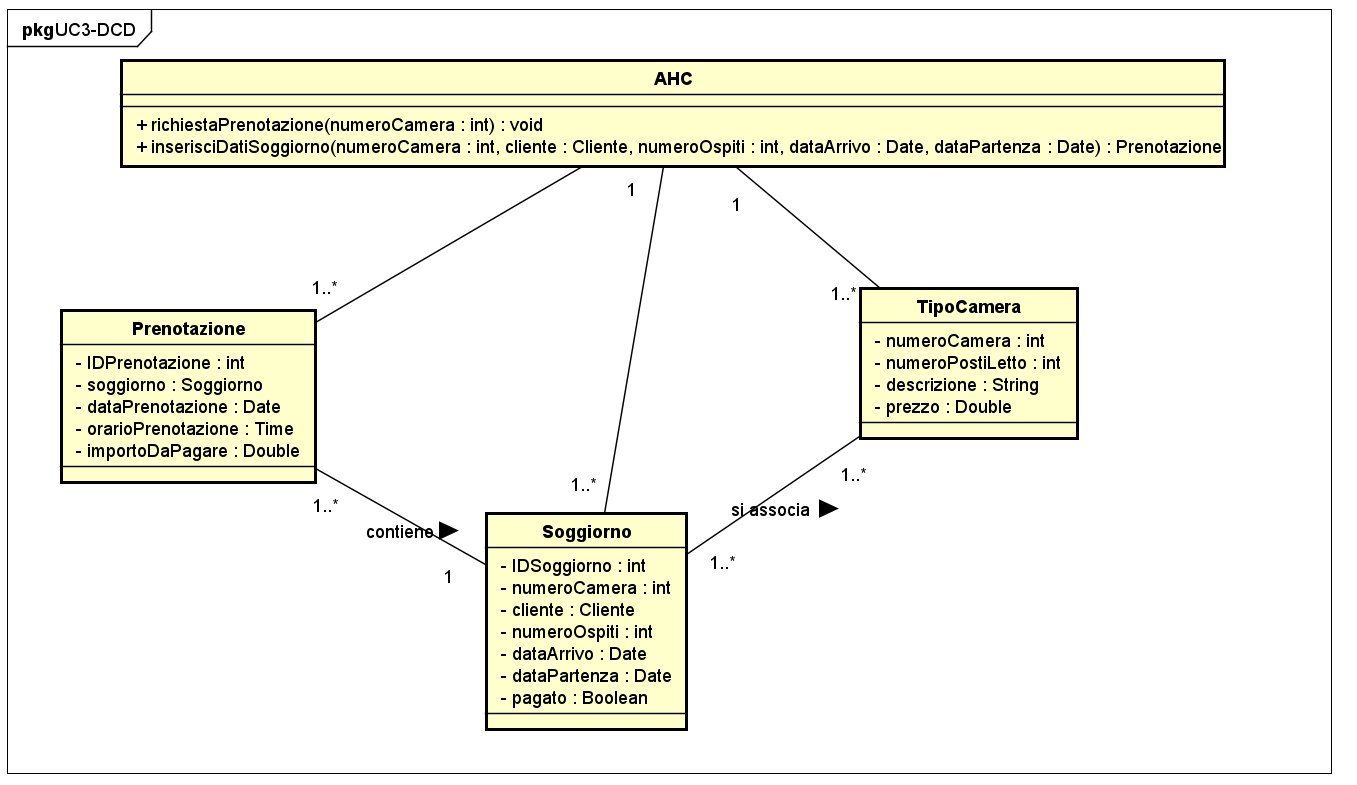
* **inserisciDatiBonifico**



### 3.4.2 Diagramma delle classi

Si rappresenta, in questa fase, la progettazione logica da un punto di vista statico.

#### Caso d’uso UC3: Effettua Prenotazione



#### Caso d’uso UC7: Effettua Pagamento

