

02 – Elaborazione – Iterazione 1

2.1 Introduzione

Conclusa la fase di ideazione si passa alla fase di elaborazione. Scopo delle iterazioni seguenti sarà quello di: raffinare la visione, implementare in maniera iterativa il nucleo dell'architettura del software, risolvere le problematiche relative ai rischi maggiori, identificare la maggior parte dei requisiti e la portata, fornire delle stime più realistiche del piano di lavoro e delle risorse complessive.

Durante questa prima iterazione i requisiti scelti sono i seguenti:

- Implementare lo scenario principale di successo del caso d'uso *UC3: Gestione ricovero*.
- Implementare un caso d'uso di Start Up necessario per gestire le esigenze di inizializzazione per questa iterazione.

2.2 Aggiornamento del caso d'uso UC3

Relativamente al caso d'uso UC3: Gestione ricovero, dopo aver stilato l'ideazione, sono state individuate alcune incongruenze e dei passi poco chiari. Al fine di semplificare l'implementazione e rendere la comprensione del progetto più chiara il caso d'uso è stato leggermente modificato. La descrizione è stata aggiornata come segue:

1. Il segretario, dopo aver ricevuto la richiesta del paziente, ricerca il servizio richiesto e il paziente tramite il software.
2. Il software restituirà la disponibilità per quel determinato servizio.
3. Verrà calcolato il preventivo ed eventualmente verrà associato uno sconto.

Nello specifico, è consigliata la visione del file *Ideazione_E1_aggiornamento*, per visualizzare lo scenario principale completo delle modifiche apportate.

2.3 Analisi Orientata agli Oggetti

L'analisi orientata agli oggetti si basa sulla costruzione del modello di dominio, delle SSD (Sequence System Diagram) e dei Contratti delle Operazioni.

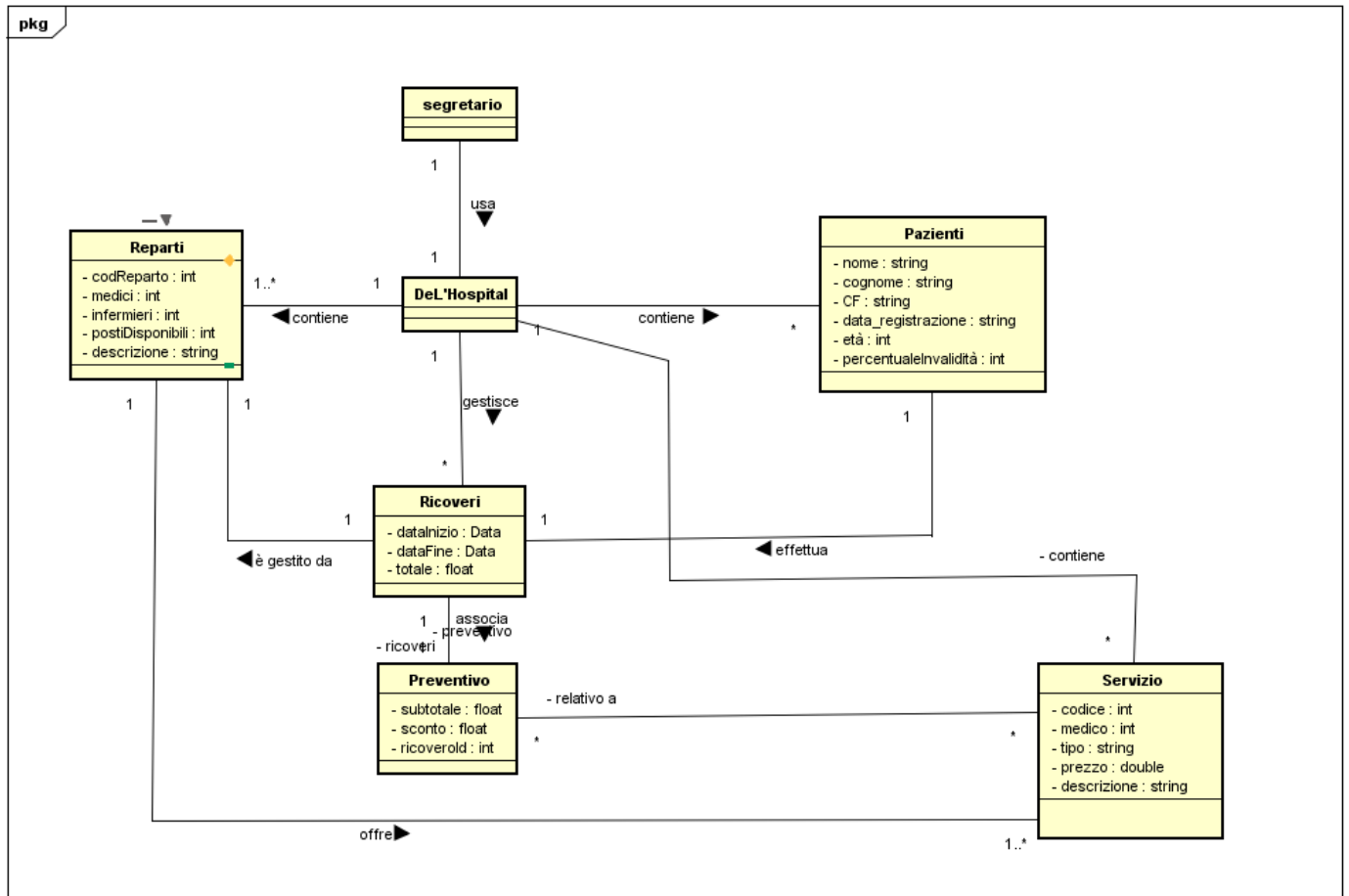
2.3.1 Modello di Dominio

Il Modello di Dominio viene utilizzato per descrivere le varie entità che fanno parte del sistema e le loro relazioni.

Relativamente al caso d'uso scelto (UC3) si evidenziano le seguenti classi concettuali:

- **Segretario:** rappresenta l'attore principale che interagisce col sistema;
- **DLH:** rappresenta il sistema;
- **Ricoveri:** rappresenta la lista di ricoveri che possono essere effettuati;
- **Preventivo:** contiene i dettagli relativi al pagamento di un particolare ricovero;
- **Servizio:** rappresenta il servizio relativo al preventivo;
- **Pazienti:** paziente che deve effettuare il ricovero;
- **Reperti:** dove il paziente effettua il ricovero;

Da cui è stato ricavato il seguente modello di dominio:

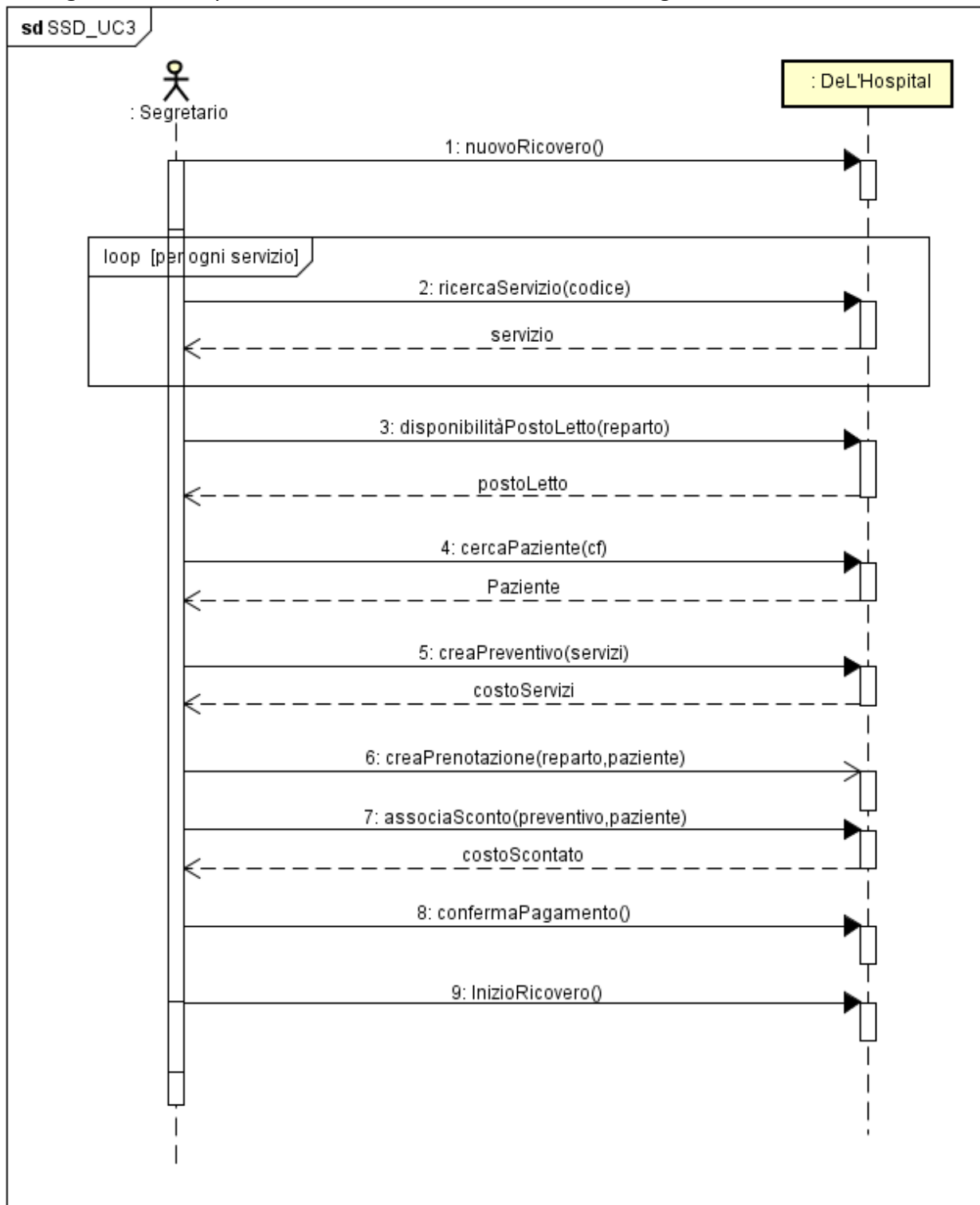


2.3.2 Diagramma di Sequenza di Sistema

Successivamente alla creazione del Modello di Dominio, si rende necessario descrivere il comportamento del sistema.

Un Diagramma di Sequenza di Sistema (SSD) descrive cosa fa il sistema, mostrando gli eventi di input e output del caso d'uso scelto, specificando gli attori che interagiscono sul sistema e le eventuali interazioni esterne con altri sistemi.

Il Diagramma di Sequenza di Sistema del caso d'uso UC3 è il seguente:



2.3.3 Contratti delle operazioni

Il passo successivo è la creazione dei contratti delle operazioni di sistema, che permettono di descrivere il comportamento del sistema in modo più dettagliato, nello specifico per gli eventi individuati nell'SSD di UC3:

Contratto CO1: nuovoRicovero()

Operazione: nuovoRicovero()
Riferimenti: caso d'uso: Gestisci ricovero
Pre-condizioni: - un paziente vuole usufruire di un servizio
Post-condizioni: -Viene creata una nuova istanza nuovoRicovero di ricovero
- gli attributi di nuovoRicovero sono stati inizializzati
- il nuovoRicovero viene aggiunto all'ArrayList ricoveri
- il ricovero nuovoRicovero è stato associato a del'Hospital tramite l'associazione "gestisce"

Contratto CO2: ricercaServizio(codice)

Operazione: ricercaServizio(codice : integer)
Riferimenti: Caso d'uso: Gestisci ricovero
Pre-condizioni: è in corso un Ricovero *nuovoRicovero*
Post-condizioni: - è noto se il *servizio* ricercato viene offerto dall'azienda ospedaliera

Contratto CO3: disponibilitàPostiLetto(reparto)

Operazione: verificaPostiDisponibili(reparto)
Riferimenti: Caso d'uso: Gestisci ricovero
Pre-condizioni: è in corso un ricovero *nuovoRicovero*
Post-condizioni: -è noto se sono presenti posti liberi

Contratto CO4: cercaPaziente(CF)

Operazione: ricercaPaziente(CF : string)
Riferimenti: Caso d'uso: Gestisci ricovero
Pre-condizioni: - è in corso un ricovero *nuovoRicovero*
Post-condizioni: - è noto se il paziente è già presente nella lista paziente

Contratto CO5: creaPreventivo(servizi)

Operazione: creaPreventivo (servizi)
Riferimenti: Caso d'uso: Gestisci ricovero
Pre-condizioni: è in corso un ricovero *nuovoRicovero*
Post-condizioni: è stato creato un nuovo preventivo che contiene il costo totale dei servizi

Contratto CO6: creaPrenotazione(reparto,paziente)

Operazione: creaPrenotazione(reparto,paziente)
Riferimenti: Caso d'uso: Gestisci ricovero
Pre-condizioni: è in corso un ricovero *nuovoRicovero*
Post-condizioni: stata inizializzata una nuova istanza *prenotazione* legata al *paziente* e al *reaprtio*

Contratto CO7: associaSconto(preventivo,paziente)

Operazione: associaSconto(preventivo,paziente)
Riferimenti: Caso d'uso: Gestisci ricovero
Pre-condizioni: è in corso un ricovero *nuovoRicovero*
Post-condizioni: è stato associato lo sconto da applicare al totale

Contratto CO8: iniziaRicovero()

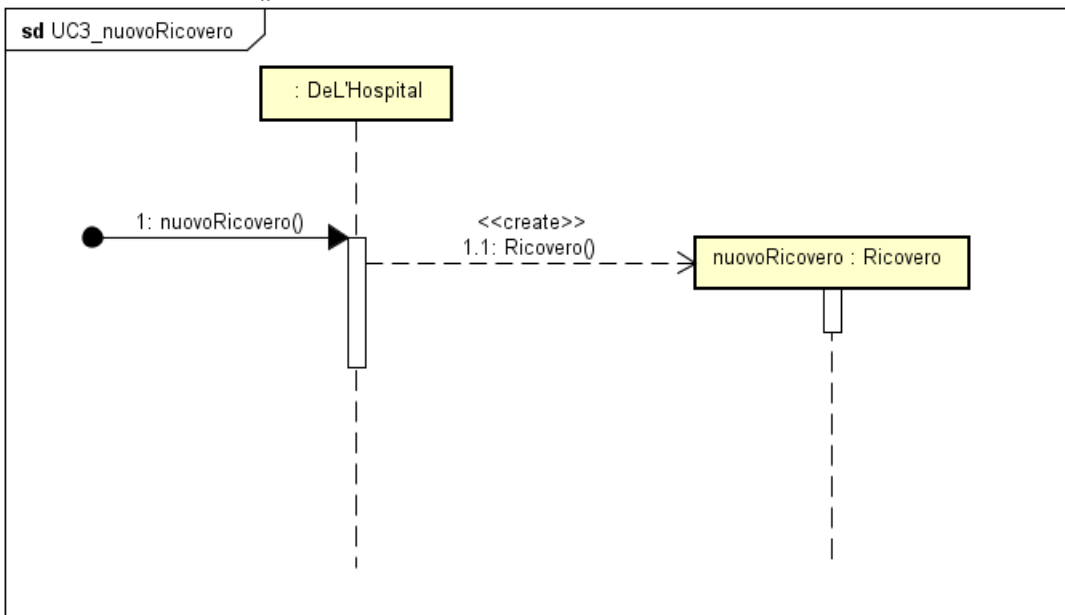
Operazione:	iniziaRicovero()
Riferimenti:	Caso d'uso: Gestisci ricovero
Pre-condizioni:	è in corso un ricovero <i>nuovoRicovero</i>
Post-condizioni:	è stato aggiunto un nuovo <i>ricovero</i> alla lista <i>listaRicoveri</i>

2.4 Progettazione

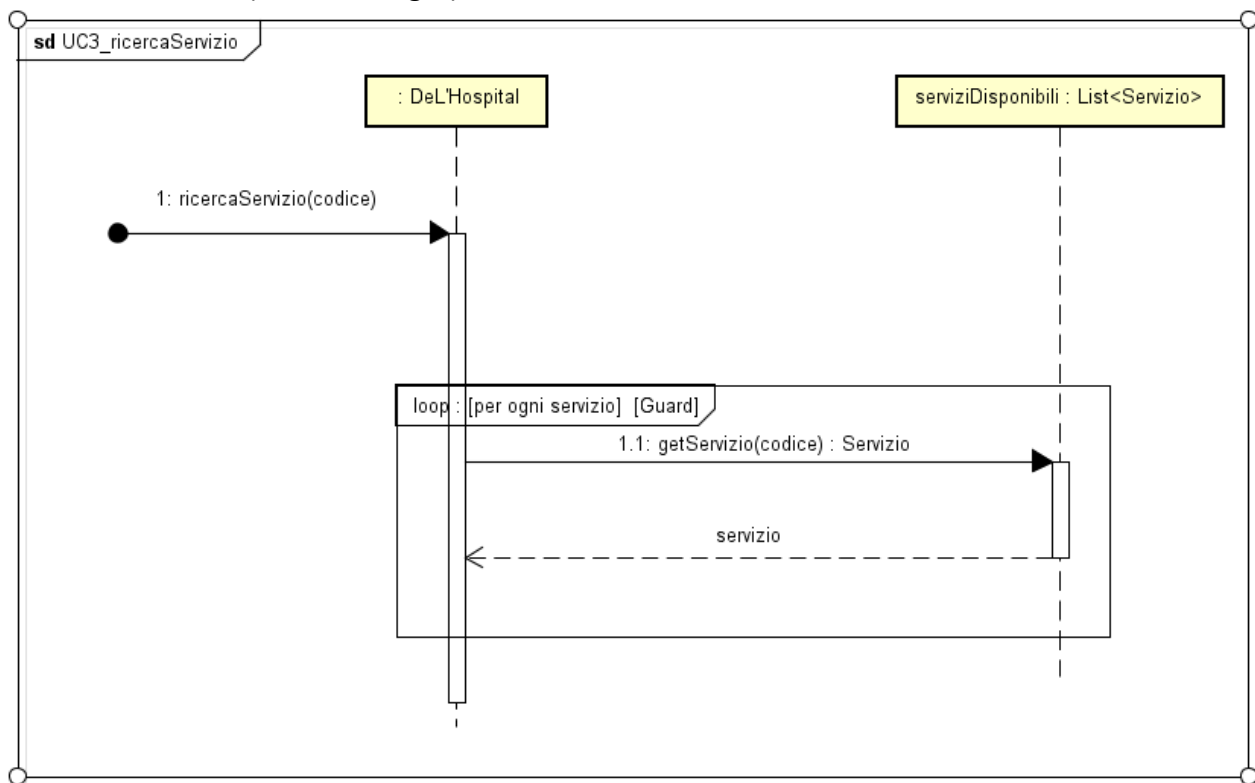
Il passo successivo consiste nella progettazione orientata agli oggetti (OOD), che è una disciplina che enfatizza la definizione e la caratterizzazione degli oggetti software e del modo in cui questi collaborano per soddisfare i requisiti. È di fondamentale importanza la scelta delle classi scelte durante l'OOD perché le classi scelte saranno poi implementate durante la programmazione orientata agli oggetti (OOP). Durante questa fase, viene creato il diagramma di sequenza, che mostra l'interazione, tra un insieme di oggetti, basata sullo scambio di messaggi. Esso infatti, documenta il comportamento di un singolo scenario.

2.4.1 Diagrammi di Sequenza UC3

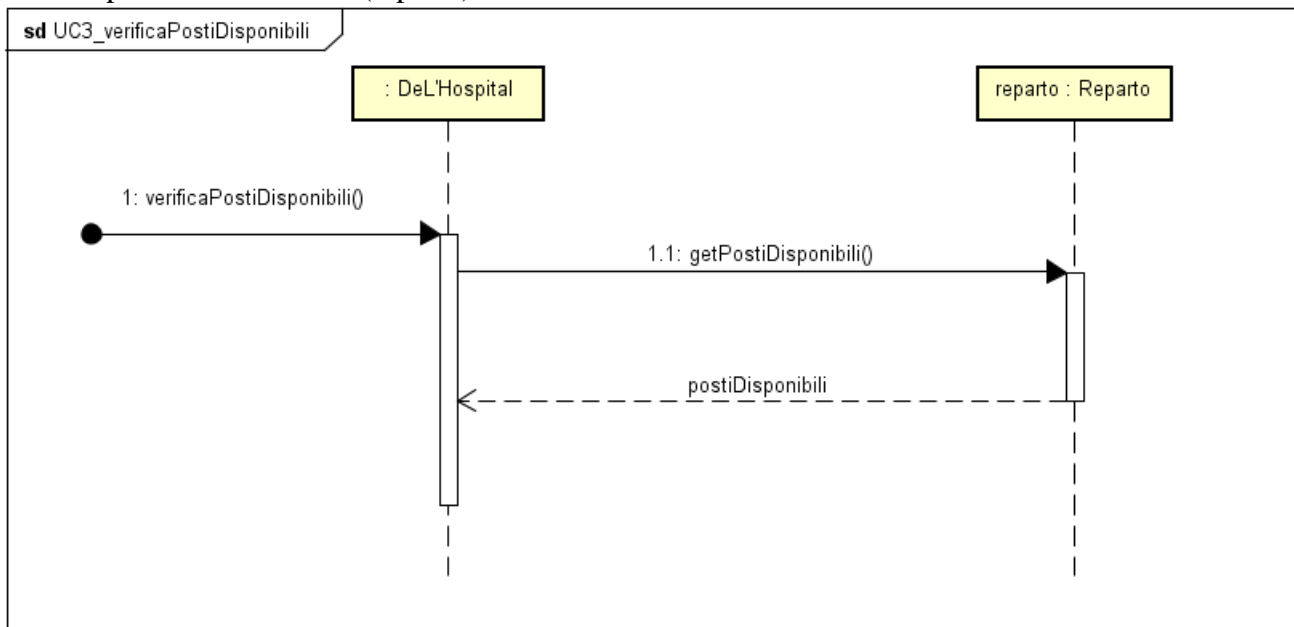
- nuovoRicovero()



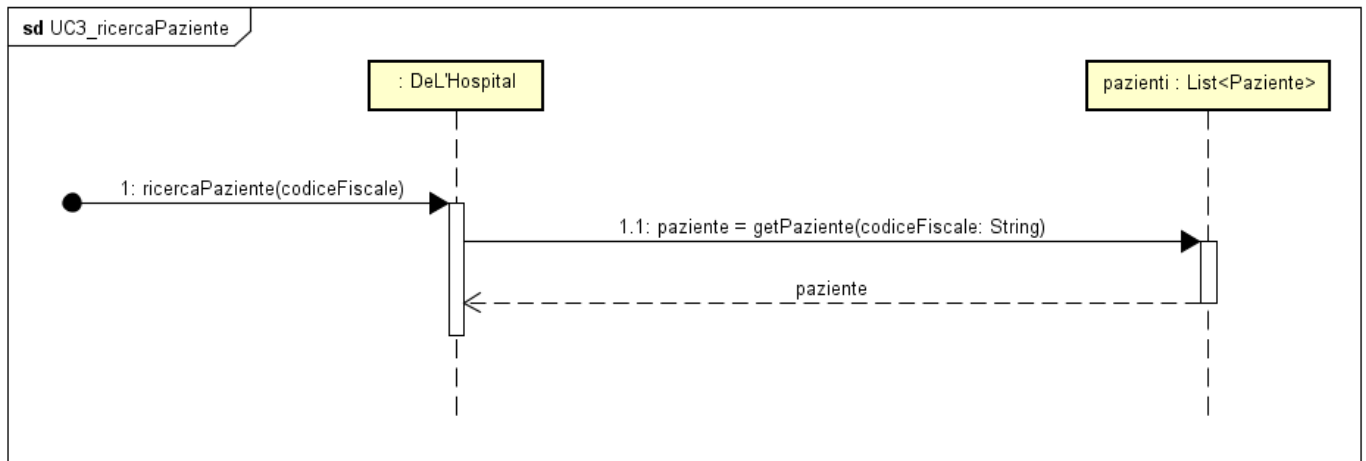
- ricercaServizio(codice: Integer)



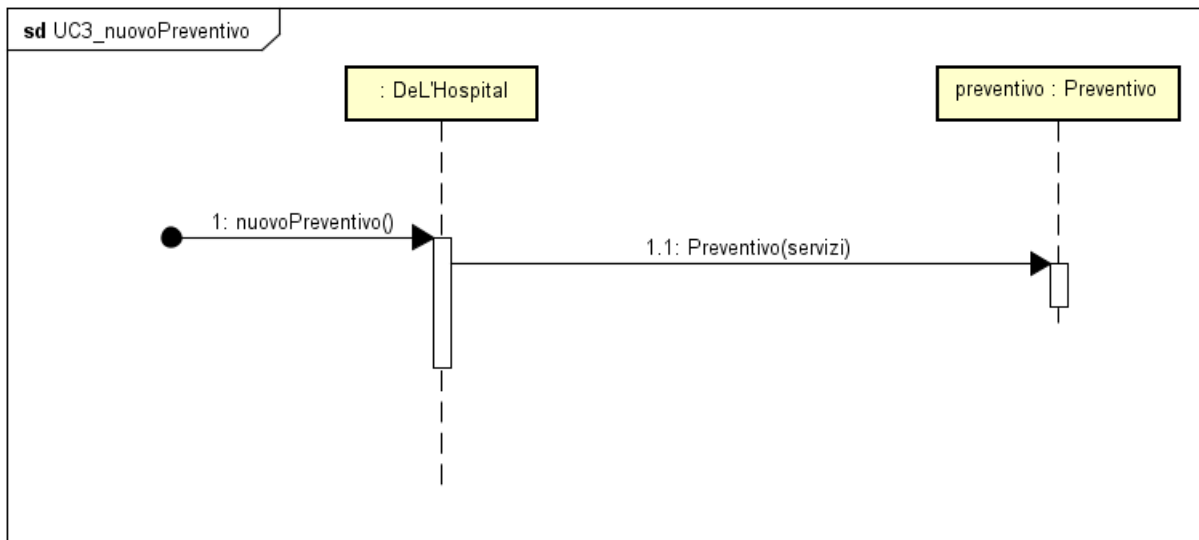
- disponibilitaPostiLetto(reparto)



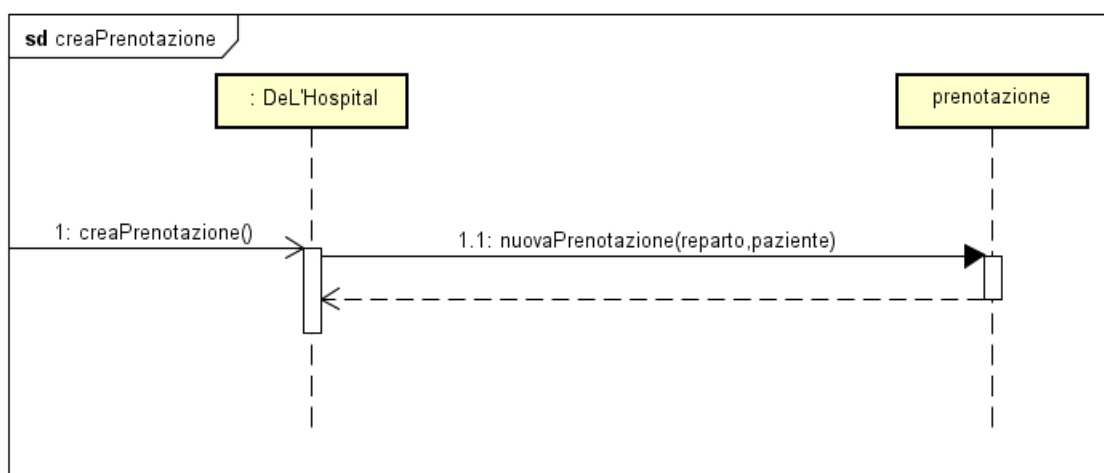
- ricercaPaziente(CF)



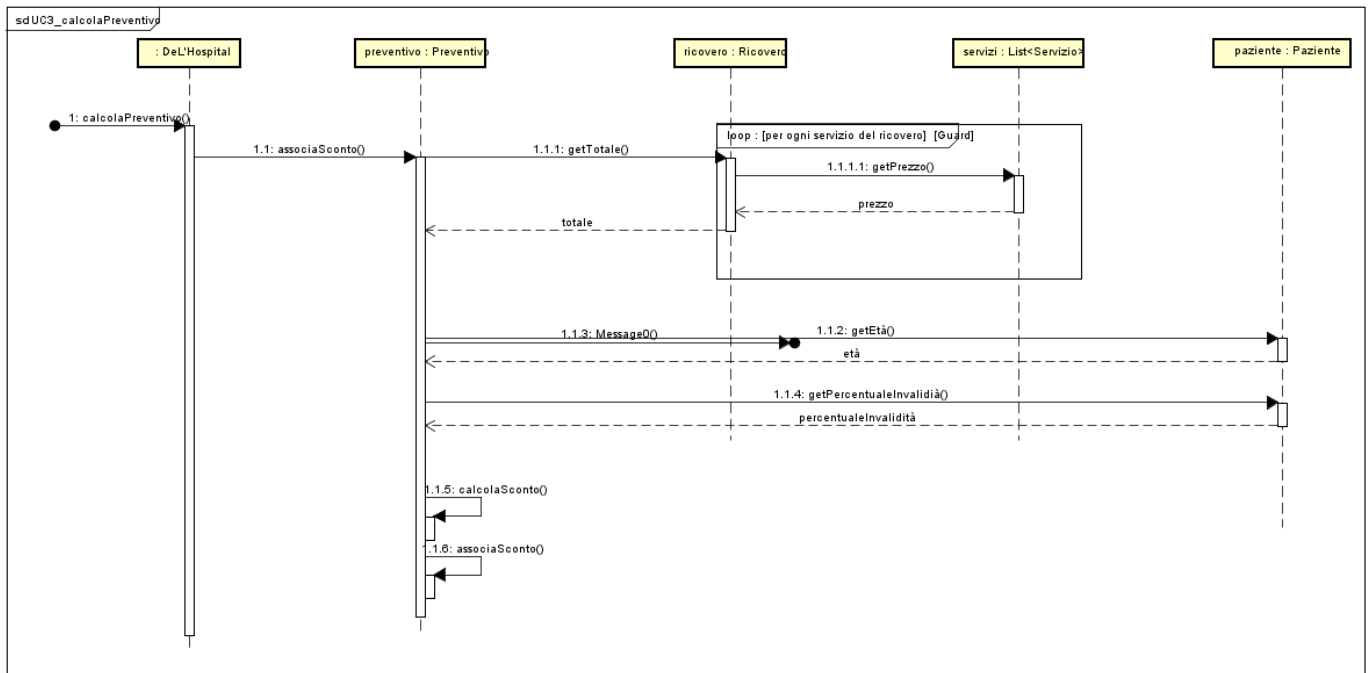
- creaPreventivo()



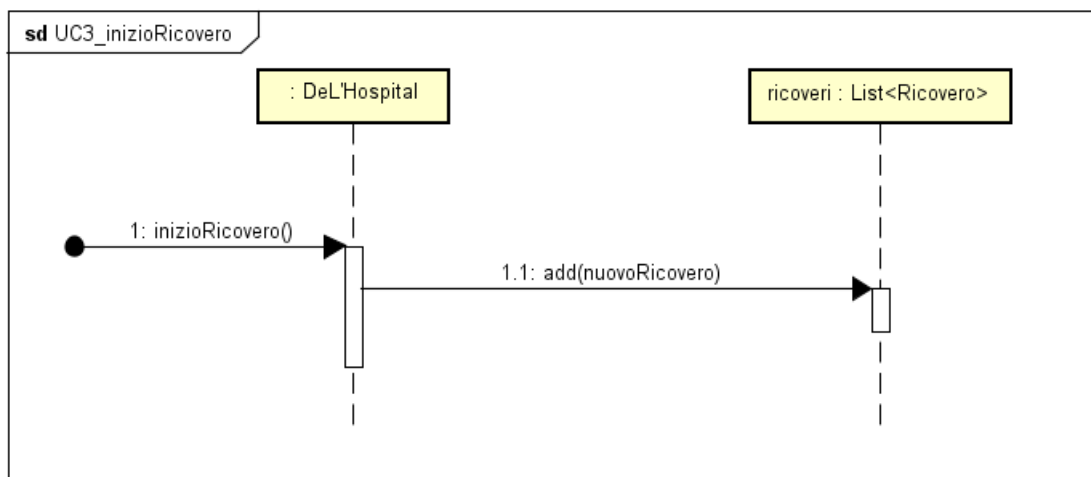
- creaPrenotazione(reparto,paziente)



- associaSconto()

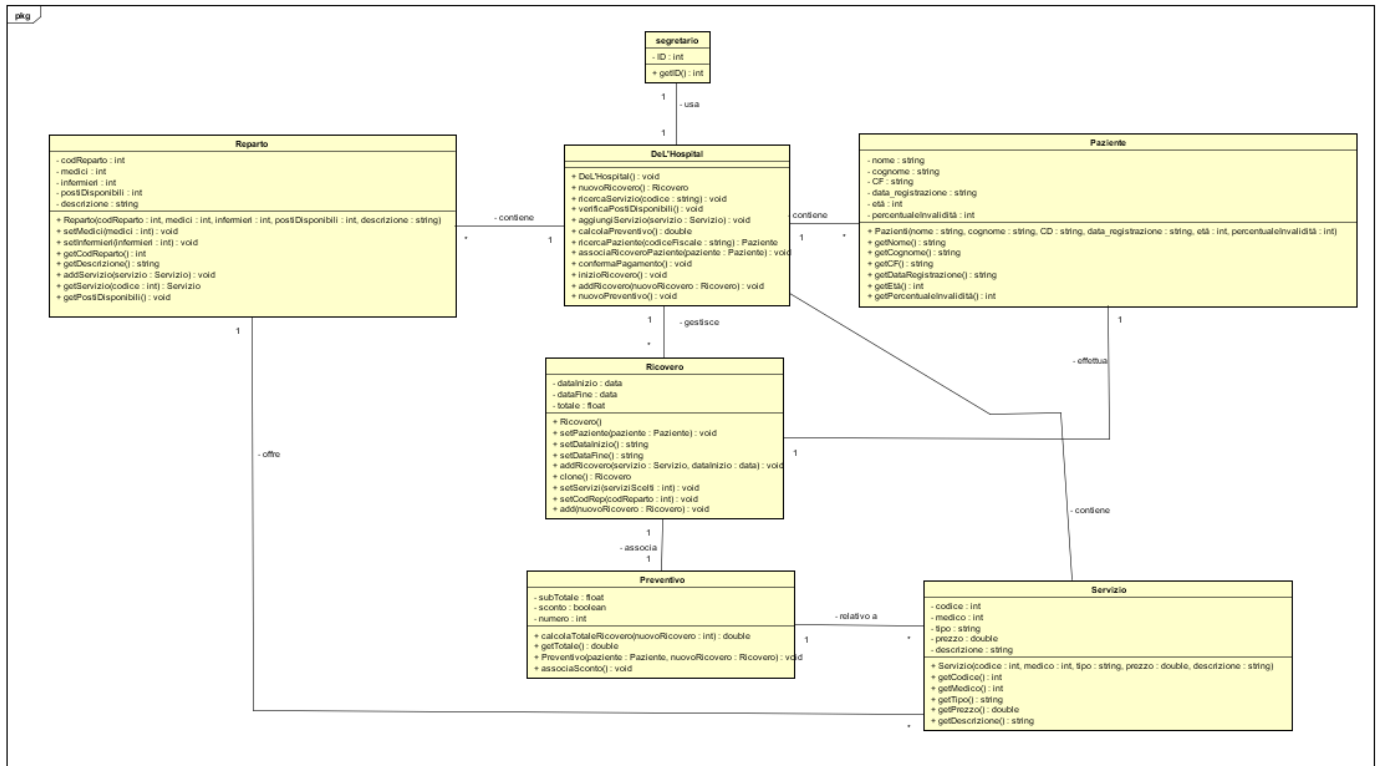


- inizioRicovero()



2.4.2 Diagramma delle classi

L'ultimo passo consiste nella creazione del diagramma delle classi, che descrive il tipo di oggetti che fanno parte di un sistema e le varie tipologie di relazioni statiche tra di essi, mostrando, nello specifico, le operazioni di una classe e i vincoli che si applicano ai collegamenti tra gli oggetti.



Per una maggiore leggibilità, vedere l'allegato *E1_DiagrammaDelleClassi.png*.