

MyClinic



Studenti:

Luca Antonio Orlando

Antonio Gabriele Reina

Prefazione

Il seguente documento presenta una descrizione relativa allo sviluppo dell'applicazione MyClinic implementata durante il corso di Ingegneria del Software. Il software è stato implementato nel linguaggio Java sfruttando l'ambiente di sviluppo IntelliJ IDEA, mentre la progettazione in UML è stata realizzata mediante il software Astah. Verranno qui presentate solamente le versioni finali di ciascun elaborato, ottenute al termine di tutte le fasi di progettazione (per visionare le versioni intermedie si vedano i documenti nelle corrispondenti sottocartelle della documentazione). Infine, vengono descritte le fasi conclusive di testing e di refactoring del programma, con cui sostanzialmente sono stati apportati dei miglioramenti al codice ed è stato aggiunto un Database per gestire la persistenza dei dati.

INDICE

<i>Corso di Laurea Magistrale in Ingegneria Informatica – A. A. 2020/2021</i>	1
<i>Ingegneria del Software</i>	1
<i>MyClinic</i>	1
1 Ideazione e analisi dei requisiti	3
1.1 Introduzione	3
1.2 Requisiti	4
1.3 Obiettivi e casi d’uso	5
1.4 Modello dei casi d’uso.....	6
1.5 Documento di visione.....	13
1.6 Specifiche supplementari	13
1.7 Glossario	13
2 Analisi Orientata agli Oggetti	14
2.1 Introduzione	14
2.2 Modello di Dominio.....	15
2.3 SSD e Contratti.....	16
3 Progettazione	21
3.1 Diagramma delle Classi	21
3.2 Diagramma di Sequenza.....	22
4 Testing	27
4.1 Introduzione	27
4.2 Individuazione dei casi di test e Testing Unitario	28
4.3 Test di Sistema.....	28
5 Refactoring e Conclusioni	29
5.1 Database e Refactoring	29
5.2 Test di Accettazione	29

1 Ideazione e analisi dei requisiti

1.1 Introduzione

L'obiettivo principale della fase di ideazione è quello di condurre un'analisi con lo scopo finale di portare alla realizzazione di un progetto. Dunque, in questa fase, è fondamentale ricavare delle considerazioni per verificare che l'idea iniziale sia realizzabile, andando a prendere in considerazione sia i tempi che le risorse necessarie.

Per un'analisi corretta, verranno considerati i seguenti documenti: Modello dei Casi d'Uso, Documento di Visione, Regole di Business, Specifiche Supplementari e Glossario.

1.2 Requisiti

Il dottore proprietario di una clinica dentistica commissiona la realizzazione di un software al fine di migliorare la gestione degli interventi di odontoiatria, dei materiali e strumenti necessari e dei dati dei pazienti. Questo software consente al medico (o a chi viene assegnato il compito di amministrare questo ambito ad es. : segretaria) di studio di accedere come amministratore ed effettuare diverse operazioni. In particolare:

- Il medico deve poter gestire l'anagrafica dei propri pazienti e tenere traccia degli interventi e operazioni;
- Il medico deve poter creare una nuova sala operatoria in base alla tipologia di operazione da effettuare;
- Il medico deve poter consultare una lista di tutti i materiali, utensili e strumenti di lavoro disponibili;
- Il medico deve poter consultare e modificare un catalogo di tutti i materiali, utensili di lavoro presso un fornitore;
- In clinica, i pazienti devono indicare la tipologia di problema da risolvere. In base alla sala operatoria a cui è destinato il paziente (in base al tipo di intervento da fare), il sistema dovrà verificare l'effettiva disponibilità. Si immagina infatti che ci si trovi all'interno di una clinica specializzata, ed in ogni sala operatoria ci siano diverse postazioni in cui i medici dentisti possono operare. L'esito della ricerca comporta due possibili scenari:
 1. Se c'è almeno 1 posto libero nella sala dedicata, si procede con l'individuazione dell'operazione da effettuare;
 2. Se non ci sono posti disponibili nella sala dedicata, si procederà con la prenotazione dell'intervento odontoiatrico.
- Nel caso 1, in base al tipo di intervento da effettuare, il sistema dovrà verificare l'effettiva disponibilità degli strumenti e materiali necessari in scorta. L'esito della ricerca comporta tre scenari:
 - a. Se i materiali sono presenti in scorta, procedere con l'intervento odontoiatrico;
 - b. Se qualche materiale non è presente in scorta, procedere con l'ordine presso il fornitore;
 - c. Il paziente si rifiuta di procedere con l'intervento.
- Nei casi a. e b. viene creata una nuova scheda relativa all'intervento necessario al paziente. Inoltre, il sistema calcolerà il totale da pagare e lo mostra a schermo. Successivamente, il sistema passerà alla cassa tali informazioni e si procederà con il pagamento da parte del paziente;

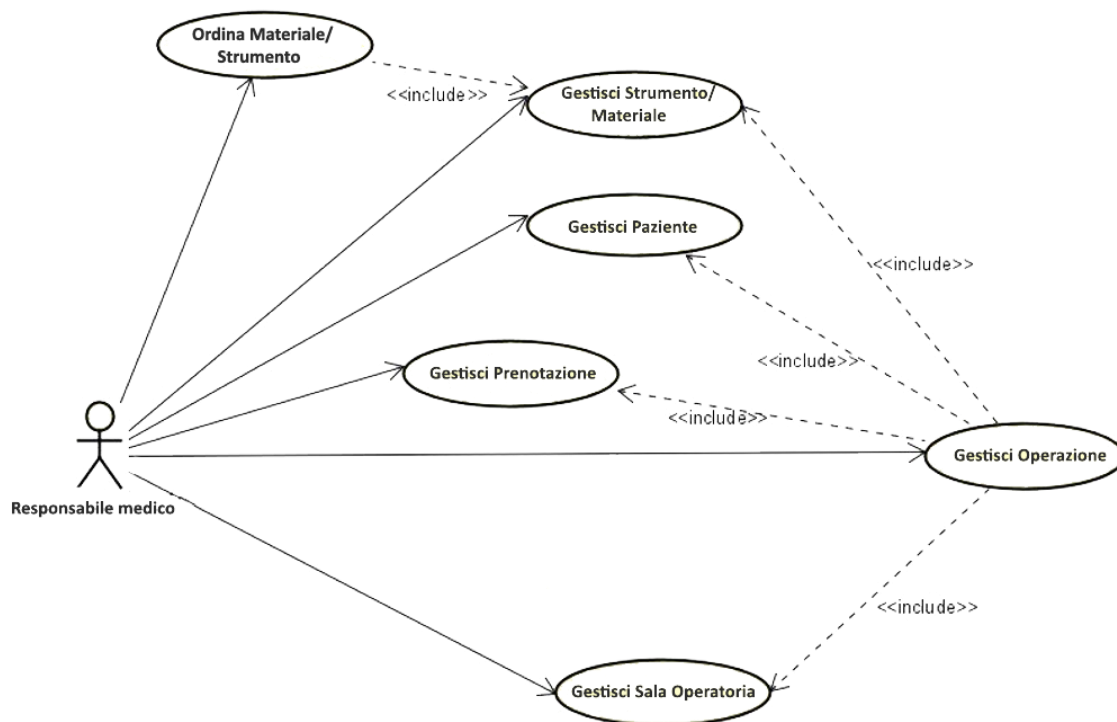
- Il responsabile medico deve poter ricercare un paziente all'interno del sistema e modificare le informazioni riguardanti lo stato dell'intervento effettuato;
- Il responsabile medico deve poter creare degli ordini verso i fornitori dei materiali da lavoro, tenendo principalmente conto dei materiali e strumenti necessari all'intervento del paziente non presenti in inventario, ma anche scegliendoli da un catalogo (ad esempio, i materiali, strumenti e medicinali più utilizzati);
- All'arrivo dei materiali e strumenti da lavoro, il responsabile medico deve poter aggiornare le giacenze;

1.3 Obiettivi e casi d'uso

Analizzando i requisiti precedentemente riportati, è stato individuato l'attore principale a cui è destinato il sistema e, inoltre, sono stati ricavati i casi d'uso principali.

Attore	Obiettivo	Casi d'uso
Responsabile medico	Gestire la creazione di un nuovo intervento odontoiatrico, comprensivo di un'eventuale prenotazione dei materiali e strumenti mancanti	UC1: Gestisci Operazione
Responsabile medico	Inserire, rimuovere, ricercare o modificare un materiale o strumento nel catalogo	UC2: Gestisci Strumento e/o Materiale (CRUD)
Responsabile medico	Creare, modificare, rimuovere o ricercare una sala operatoria	UC3: Gestisci Sala Operatoria (CRUD)
Responsabile medico	Creare, modificare, rimuovere o ricercare una prenotazione per effettuare un'operazione in base alla disponibilità di posti liberi nella sala operatoria dedicata e al numero di sedute che occorre per finalizzare l'operazione.	UC4: Gestisci Prenotazione (CRUD)
Responsabile medico	Gestire i dati relativi a ciascun paziente (inserisci, modifica, rimuovi, ricerca)	UC5: Gestisci Paziente(CRUD)
Responsabile medico	Creare un ordine verso un fornitore per ordinare i materiali richiesti dal medico e destinati alla scorta	UC6: Ordina materiale

I casi d'uso appena descritti si traducono graficamente nel seguente diagramma:



1.4 Modello dei casi d'uso

Forniamo una descrizione in dettaglio per i casi d'uso *UC1: Gestisci Operazione* e *UC5: Gestisci Paziente*.

Gli altri casi d'uso vengono descritti nel formato breve o informale.

UC1: Gestisci Operazione

Nome del caso d'uso	UC1: Gestisci Operazione
Portata	Applicazione: MyClinic
Livello	Obiettivo paziente
Attore primario	Responsabile medico
Parti interessate e interessi	<ul style="list-style-type: none"> Responsabile medico: vuole gestire l'intera scheda relativa all'operazione dentistica da effettuare; Paziente: vuole risolvere un problema ai denti.
Pre-condizioni	La sala operatoria per effettuare un determinato intervento ha almeno un posto disponibile
Garanzia di successo	L'operazione si conclude con successo e il paziente effettua il pagamento alla cassa. Viene aggiornata la scorta degli strumenti e materiali.
Scenario principale di successo	1. Un paziente arriva in clinica e chiede una visita dentistica.

	<ol style="list-style-type: none"> 2. Il responsabile medico utilizza il software MyClinic per verificare la disponibilità di posti liberi all'interno della sala operatoria <i>[vedi UC3]</i>. Il sistema mostra al responsabile medico l'esito positivo della ricerca, il quale lo comunica al paziente. 3. Il paziente viene sottoposto a visita da parte dei medici per capire che tipologia di operazione effettuare. Al termine di questa fase, vengono comunicati al responsabile medico i materiali e gli strumenti necessari per effettuare l'operazione. 4. Il responsabile medico utilizza il software MyClinic per verificare la disponibilità degli strumenti e materiali, necessari all'operazione, all'interno delle giacenze <i>[vedi UC2]</i>. Il sistema mostra al responsabile medico l'esito positivo della ricerca. 5. L'operazione dentistica viene eseguita con successo. 6. Il responsabile medico, utilizzando il software MyClinic, crea una nuova scheda relativa all'operazione effettuata. Il paziente viene ricercato all'interno del sistema <i>[vedi UC5]</i>, il software restituisce il risultato richiesto e viene associato all'intervento. 7. Il responsabile medico conferma l'operazione. 8. Il sistema aggiorna le giacenze . 9. Se l'operazione richiede più sedute si procederà alla prenotazione della seduta successiva, fornendo l'id operazione ottenuto alla creazione della prima prenotazione. 10. Al termine dell'ultima seduta di operazione sistema calcola l'ammontare da pagare <i>[vedi UC7]</i> e mostra a video il totale. 11. Il paziente si reca alla cassa per effettuare il pagamento.
Estensioni	<p>*a. In qualsiasi momento il sistema va in fault e ha un arresto improvviso:</p> <ol style="list-style-type: none"> 1. Il responsabile medico riavvia il software MyClinic; 2. Il sistema ritorna allo stato precedente. <p>2a. Non ci sono posti disponibili nella sala operatoria:</p> <ol style="list-style-type: none"> 1. Il responsabile medico comunica al paziente che non potrà essere servito momentaneamente; 2. Il paziente decide di prenotarsi per la disponibilità futura di un posto nella sala operatoria dedicata; 3. Il sistema registra la prenotazione del paziente <i>[vedi UC4]</i>; 4. Il paziente se ne va; 5. Il responsabile medico termina l'esecuzione del programma. <p>4a. Un materiale o strumento di lavoro non è presente nelle giacenze:</p> <ol style="list-style-type: none"> 1. Il sistema chiede al responsabile medico di effettuare un ordine presso il fornitore competente; 2. Il responsabile medico seleziona lo strumento o materiale desiderato dal catalogo e conferma l'ordine <i>[vedi UC8]</i>. <p>5a. Il paziente non è presente all'interno del sistema:</p> <ol style="list-style-type: none"> 1. Il sistema chiede al responsabile medico di inserire il nuovo paziente; 2. Il responsabile medico inserisce i dati e conferma <i>[vedi UC5]</i>; 3. Il sistema associa il nuovo paziente all'intervento da effettuare.

Requisiti speciali	Non specificati
Elenco delle varianti tecnologiche e dei dati	PC
Frequenza di ripetizioni	Legata all'affluenza dei pazienti

UC2: Gestisci Strumento e/o Materiale

1. Il responsabile medico richiede al sistema l'**inserimento** in catalogo di un nuovo materiale o strumento di lavoro.
2. Il sistema chiede al responsabile medico se si tratta di un pezzo singolo oppure di una serie di pezzi più complessi (ad esempio, le varie parti di un apparecchio dentistico).
3. Il responsabile medico sceglie l'opzione "*singolo*".
4. Il sistema richiede al responsabile medico le informazioni relative allo strumento o materiale.
5. Il responsabile medico inserisce codice identificativo, nome, fornitore, marca, disponibilità, prezzo e singolo.

Possibili Scenari

1a. Il responsabile medico richiede al sistema la **modifica** di uno strumento o materiale di lavoro:

1. Il sistema richiede al responsabile medico i dati aggiornati.
2. Il responsabile medico aggiorna i campi di interesse e conferma.

1b. Il responsabile medico richiede al sistema di **ricercare** uno strumento o materiale di lavoro:

1. Il sistema richiede al responsabile medico di inserire le informazioni necessarie nei campi di ricerca.
2. Il responsabile medico inserisce le informazioni che possiede.
3. Il sistema mostra l'elenco degli strumenti che soddisfano i criteri di ricerca.

1c. Il responsabile medico richiede al sistema di **eliminare** uno strumento o materiale di lavoro:

1. Il sistema chiede al responsabile medico di confermare l'eliminazione.
2. Il responsabile medico conferma.
3. Il sistema elimina l'oggetto.

3a. Il responsabile medico sceglie l'opzione "*complesso*":

1. Il sistema richiede al responsabile medico di inserire le componenti che costituiscono lo strumento o materiale completo.
2. Il responsabile medico inserisce le componenti richieste e conferma.

UC3: Gestisci Sala Operatoria

1. Il responsabile medico richiede al sistema l'**inserimento** di una nuova sala operatoria.

2. Il sistema richiede al responsabile medico le informazioni relative alla sala.
3. Il responsabile medico inserisce id_sala, nome e posti_disponibili.

Scenari alternativi

1a. Il responsabile medico richiede al sistema la **modifica** di una sala operatoria:

1. Il sistema richiede al responsabile medico i dati aggiornati.
2. Il responsabile medico aggiorna i campi di interesse e conferma.

1b. Il responsabile medico richiede al sistema di **ricercare** una sala operatoria:

1. Il sistema richiede al responsabile medico di inserire le informazioni necessarie nei campi di ricerca.
2. Il responsabile medico inserisce le informazioni che ha a disposizione.
3. Il sistema mostra l'elenco delle sale che soddisfano i criteri di ricerca.

1c. Il responsabile medico richiede al sistema di **eliminare** una sala operatoria:

1. Il sistema chiede al responsabile medico di confermare l'eliminazione.
2. Il responsabile medico conferma.
3. Il sistema elimina la sala operatoria.

UC4: Gestisci Prenotazione

1. Il responsabile medico richiede al sistema l'**inserimento** di una nuova prenotazione.
2. Il sistema richiede al responsabile medico le informazioni relative alla prenotazione.
3. Il responsabile medico inserisce id_prenotazione, id_operazione, sala_operatoria, paziente e prenotato.

Scenari alternativi

1a. Il responsabile medico richiede al sistema la **modifica** di una prenotazione per un errore commesso:

1. Il sistema richiede al responsabile medico i dati aggiornati;
2. Il responsabile medico aggiorna i campi di interesse e conferma.

1b. Il responsabile medico richiede al sistema di **ricercare** una prenotazione:

1. Il sistema richiede al responsabile medico di inserire le informazioni necessarie nei campi di ricerca;
2. Il responsabile medico inserisce le informazioni a sua disposizione;
3. Il sistema mostra l'elenco delle prenotazioni che soddisfano i criteri di ricerca.

1c. Il responsabile medico richiede al sistema di **eliminare** una prenotazione:

1. Il sistema chiede al responsabile medico di confermare l'eliminazione;
2. Il responsabile medico conferma;
3. Il sistema elimina la prenotazione.

1d. Il responsabile medico richiede al sistema di **inserire** una prenotazione **associata** ad un'operazione esistente che si svolgerà in più sedute:

1. Il responsabile medico richiede al sistema l'**inserimento** di una nuova prenotazione.
2. Il sistema richiede al responsabile medico le informazioni relative alla prenotazione.

3. Il responsabile medico inserisce id_prenotazione, id_operazione , sala_operatoria, paziente e prenotato. In questo caso **id_operazione** sarà un codice fornito dal paziente, che gli verrà inviato via mail contestualmente alla prima prenotazione.
4. I tre step precedenti dovranno essere ripetuti al termine di ciascuna seduta, fino a quella conclusiva.

UC5: Gestisci Paziente

Nome del caso d'uso	UC5: Gestisci Paziente
Portata	Applicazione: MyClinic
Livello	Obiettivo paziente
Attore primario	Responsabile medico
Parti interessate e interessi	<ul style="list-style-type: none"> • Responsabile medico: vuole gestire l'intero processo di registrazione dei dati relativi ai pazienti.
Pre-condizioni	La sala operatoria in cui effettuare l'operazione ha almeno un posto disponibile
Garanzia di successo	La registrazione si conclude se i dati inseriti sono corretti. La lista dei pazienti viene aggiornata.
Scenario principale di successo	<ol style="list-style-type: none"> 1. Il responsabile medico utilizza il software MyClinic per inserire il paziente nel sistema. 2. Il paziente fornisce i dati necessari. 3. Il responsabile medico inserisce CF, nome, cognome, email, cellulare, operazioni_passate. 4. Il sistema memorizza le informazioni sul cliente. 5. Il responsabile medico comunica al paziente di aver concluso la registrazione.
Estensioni	<p>*a. In qualsiasi momento il sistema fallisce e ha un arresto improvviso:</p> <ol style="list-style-type: none"> 1. Il responsabile medico riavvia il software; 2. Il sistema ritorna allo stato precedente. <p>1a. Il responsabile medico richiede al sistema la modifica dei dati di un paziente:</p> <ol style="list-style-type: none"> 1. Il sistema richiede al responsabile medico di inserire i dati aggiornati. 2. Il responsabile medico aggiorna i campi di interesse e conferma. <p>1b. Il responsabile medico richiede al sistema la ricerca di un paziente:</p> <ol style="list-style-type: none"> 1. Il sistema richiede al responsabile medico di inserire il codice fiscale del paziente da cercare. 2. Il responsabile medico inserisce il codice fiscale e conferma. 3. Il sistema restituisce il paziente corrispondente. <p>1c. Il responsabile medico richiede al sistema l'eliminazione di un paziente:</p> <ol style="list-style-type: none"> 1. Il sistema richiede al responsabile medico di inserire il codice fiscale del paziente da eliminare. 2. Il responsabile medico inserisce il codice fiscale e conferma. 3. Il sistema elimina il paziente corrispondente.

	3a. Il responsabile medico inserisce informazioni sbagliate: <ol style="list-style-type: none"> 1. Il responsabile medico controlla i dati inseriti. 2. Il responsabile medico modifica i dati errati e conferma. 3. Il sistema memorizza le informazioni sul paziente.
Requisiti speciali	Non specificati
Elenco delle varianti tecnologiche e dei dati	PC
Frequenza di ripetizioni	Legata all'affluenza dei pazienti

UC6: Ordina Materiale

1. Il responsabile medico richiede al sistema la creazione di un nuovo ordine di fornitura.
2. Il sistema chiede l'inserimento del fornitore.
3. Il responsabile medico inserisce l'informazione e conferma.
4. Il sistema chiede se si tratta di un materiale singolo o un materiale / strumento più complesso.
5. Il responsabile medico inserisce l'informazione e conferma.
6. Il sistema chiede l'inserimento della marca e nome dell'oggetto desiderato.
7. Il responsabile medico inserisce l'informazione e conferma.
8. Il sistema mostra il catalogo degli strumenti che soddisfano i precedenti criteri.
9. Il responsabile medico seleziona ed inserisce lo strumento o materiale desiderato nel nuovo ordine.
10. Il sistema chiede al responsabile medico se vuole aggiungere ulteriori oggetti all'ordine.
11. Il responsabile medico rifiuta e conferma la creazione del nuovo ordine.

Scenari alternativi

13a. Il responsabile medico accetta, aggiunge un ulteriore oggetto all'ordine e conferma la creazione del medesimo.

1.5 Documento di visione

Il Documento di Visione di questo capitolo è presente nell'*appendice A*. Questo documento è stato redatto nel corso dello sviluppo della fase di ideazione e verranno apportate modifiche nelle successive iterazioni.

1.6 Specifiche supplementari

Usabilità:

- L'interazione col sistema deve essere semplice;
- Dare priorità alla velocità di elaborazione e l'assenza di errori.

Affidabilità:

- È necessario garantire una forma, di ritenzione dei dati in caso di fault o altre problematiche;
- Sono necessari backup periodici dei dati nel database.

Aspetti implementativi:

- Il software è basato su linguaggio Java;
- Viene sfruttato un database SQL per la persistenza dei dati.

1.7 Glossario

- **Fornitore:** azienda realizzatrice e distributrice di materiali e strumenti presso cui la clinica dentistica si rifornisce.
- **Giacenza:** strumenti, materiali e medicinali che sono conservati nelle scorte in clinica.
- **Intervento:** si riferisce all'operazione da effettuare ad un determinato paziente.

2 Analisi Orientata agli Oggetti

2.1 Introduzione

La realizzazione dell'applicazione è stata articolata su quattro iterazioni. In questo modo, è stato possibile implementare in maniera iterativa il nucleo dell'architettura del software, sono state risolte problematiche relative ai rischi maggiori ed è stata affrontata un'analisi dei requisiti graduale, in modo tale da limitare al minimo il danno causato da eventuali errori di progettazione e implementazione.

In particolare, per ciascuna iterazione ci si è occupati di gestire le seguenti problematiche:

➤ Iterazione 1

- Implementazione dello scenario principale di successo del caso d'uso *UC1: Gestisci operazione*;
- Implementazione di un caso d'uso di Start Up necessario per gestire le esigenze di inizializzazione per questa iterazione.

➤ Iterazione 2

- Implementazione dello scenario alternativo **2a** del caso d'uso *UC1: Gestisci Operazione*, in cui si prevede che non sono disponibili posti nella Sala Operatoria per poter effettuare un'Operazione, ma nonostante ciò il paziente effettua una prenotazione;
- Implementazione del caso d'uso *UC4: Gestisci Prenotazione*, in quanto completa lo scenario alternativo **2a**. In questo modo, sarà possibile creare una prenotazione se non sono disponibili posti nella sala operatoria e sarà aggiornata quando i posti saranno disponibili.

➤ Iterazione 3

- Ci concentreremo sullo sviluppo del caso d'uso *UC6: Ordina Materiale*, il cui obiettivo è quello di consentire al responsabile medico di poter effettuare un ordine, verso un fornitore, per ordinare i materiali richiesti dal medico e destinati alla scorta.
- In particolare, andremo ad analizzare l'estensione **4a** del caso d'uso *UC1: Gestisci Operazione*, rappresentante la situazione in cui un materiale, o strumento di lavoro, non è presente nelle giacenze e, dunque, nasce l'esigenza di dover effettuare un ordine. Da qui, lo studio del caso d'uso *UC6*.

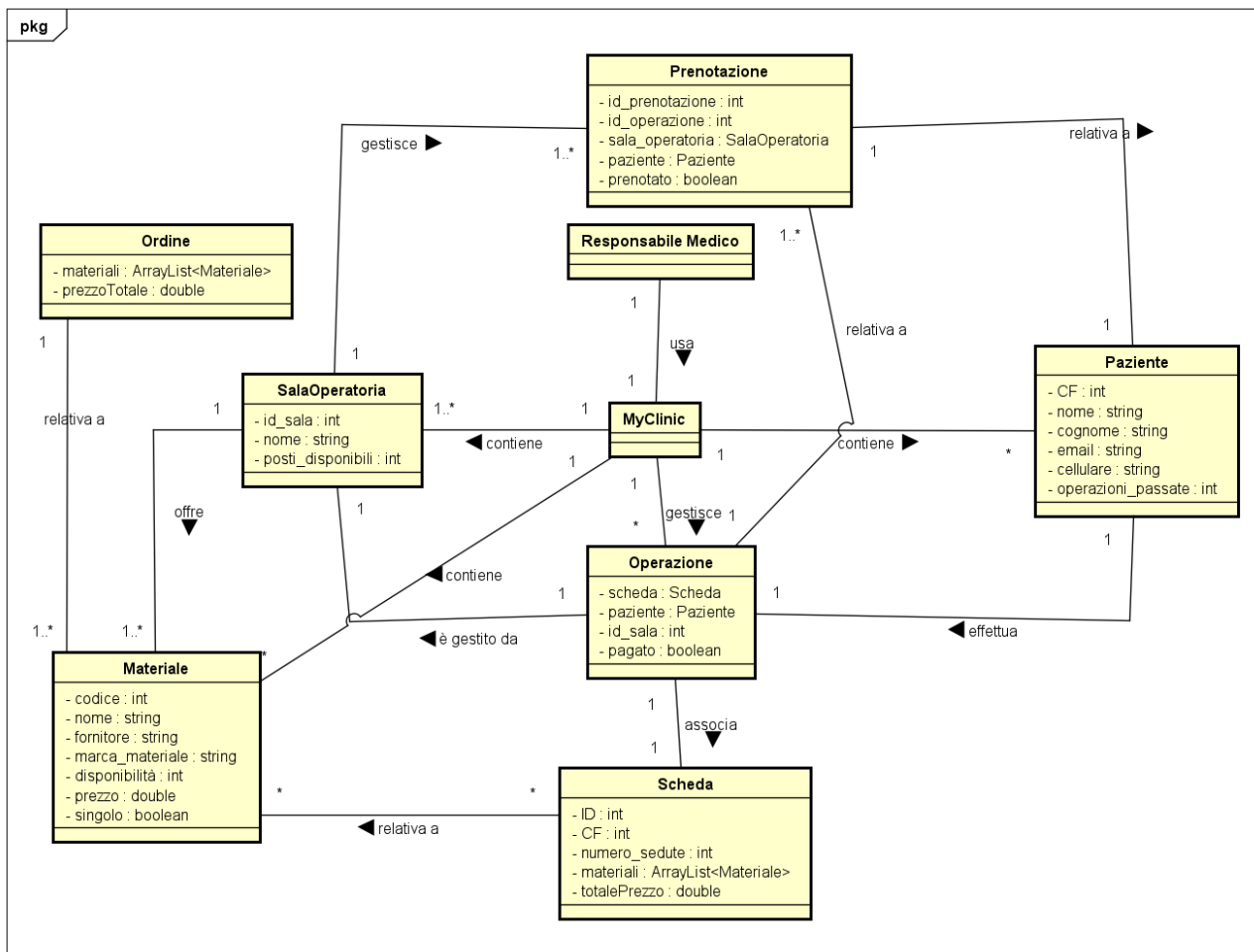
➤ Iterazione 4

- Implementazione dello scenario alternativo **2a** del caso d'uso *UC1: Gestisci Operazione*;
- Implementazione del caso d'uso *UC5: Gestisci Paziente*, il cui scopo è quello di inserire, modificare ed eliminare un paziente dal database.

Il passo iniziale di ciascuna iterazione è stato quello di effettuare un'analisi dei requisiti orientata agli oggetti, cioè un'analisi basata sulla creazione di una descrizione del dominio da un punto di vista degli oggetti. Per fornire tale descrizione, sono stati usati diversi strumenti: Modello di Dominio, SSD (Sequence System Diagram) e Contratti delle Operazioni.

2.2 Modello di Dominio

La disciplina che, in termini di UP, si occupa di fornire dettagli sul dominio è la modellazione del Business. In particolare, essa comprende la stesura del Modello di Dominio, ovvero un elaborato grafico in cui vengono identificati i concetti, gli attributi e le associazioni considerati significativi. Tenendo conto del contributo dato da ciascuna iterazione, il Modello di Dominio è dato da:



Per una maggiore visibilit , si consiglia di vedere l'allegato *ModelloDiDominioCompleto.png* dell'ultima iterazione. Come si pu  notare dall'immagine soprastante, sono identificate le seguenti classi concettuali:

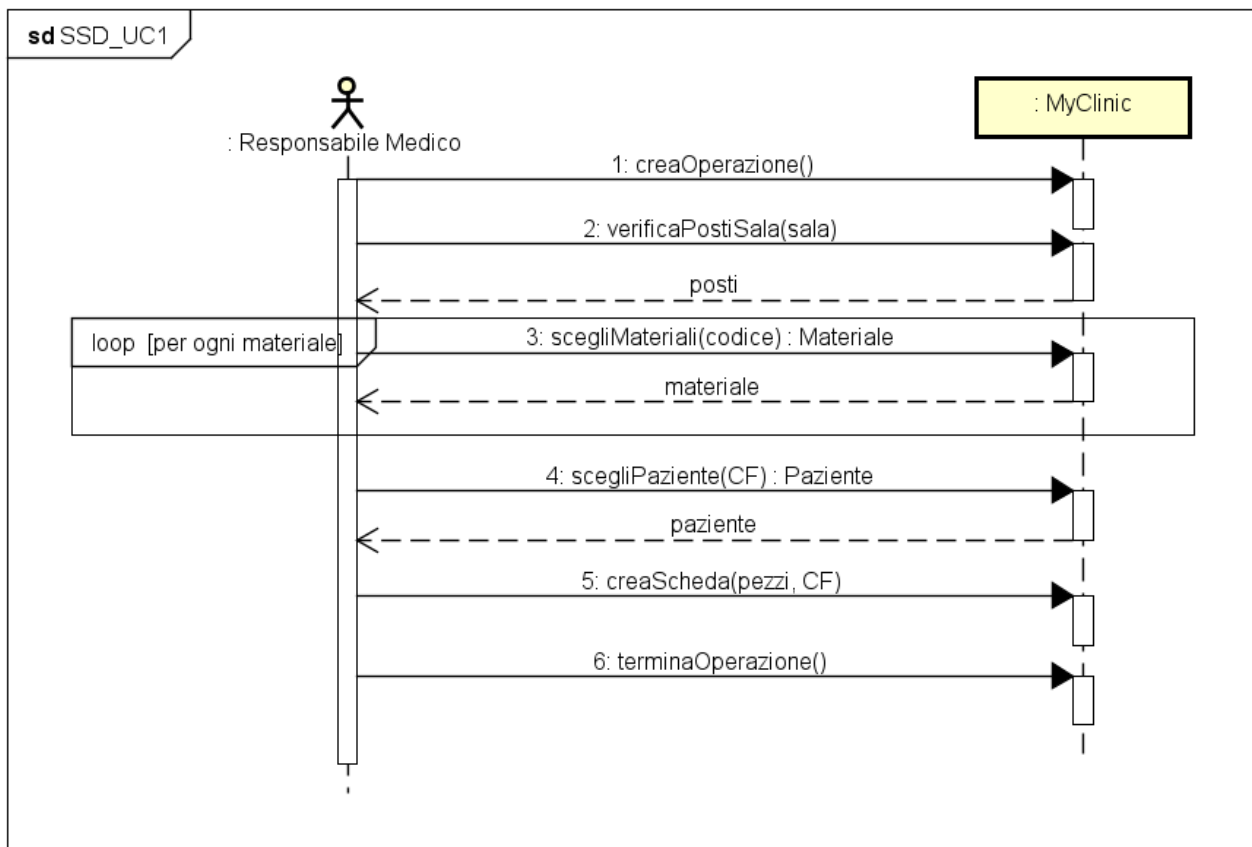
- **Responsabile medico:** rappresenta l'attore principale che interagisce col sistema;
- **MyClinic:** rappresenta il sistema;
- **Operazione:** contiene il riferimento alla scheda, al paziente e al pagamento;
- **Scheda:** contiene i dettagli relativi all'operazione;
- **Materiale:** rappresenta i materiali necessari all'operazione;

- **Paziente:** rappresenta colui che deve usufruire del servizio di cura dentistica;
- **Sala Operatoria:** rappresenta il reparto della clinica a cui è destinato il paziente da curare;
- **Prenotazione:** rappresenta la prenotazione di un paziente per una determinata Sala Operatoria;

2.3 SSD e Contratti

➤ Iterazione 1

Il Diagramma di Sequenza di Sistema del caso d'uso *UC1* è il seguente:



Contratto C01: creaOperazione()

- **Operazione:**
 - *creaOperazione()*
- **Riferimenti:**
 - Caso d'uso: Gestisci Operazione
- **Pre-condizioni:**
 - Un paziente vuole curarsi.
- **Post-condizioni:**
 - Viene creata una nuova istanza *Operazione*;
 - Gli attributi di *Operazione* sono stati inizializzati;
 - L'*Operazione* viene aggiunto all'*ArrayList operazioni*;

- L'Operazione è stata associata a MyClinic tramite l'associazione "gestisce".

Contratto C02: verificaPostiSala(sala)

- **Operazione:**
 - *verificaPostiSala(sala)*
- **Riferimenti:**
 - Caso d'uso: Gestisci Operazione
- **Pre-condizioni:**
 - È in corso un'Operazione.
- **Post-condizioni:**
 - È noto se sono presenti posti liberi nella sala di lavoro.

Contratto C03: scegliMateriali(codice)

- **Operazione:**
 - *scegliMateriali(codice: Integer)*
- **Riferimenti:**
 - Caso d'uso: Gestisci Operazione
- **Pre-condizioni:**
 - È in corso un'Operazione.
- **Post-condizioni:**
 - È noto se il materiale è presente nelle giacenze.

Contratto C04: scegliPaziente(CF)

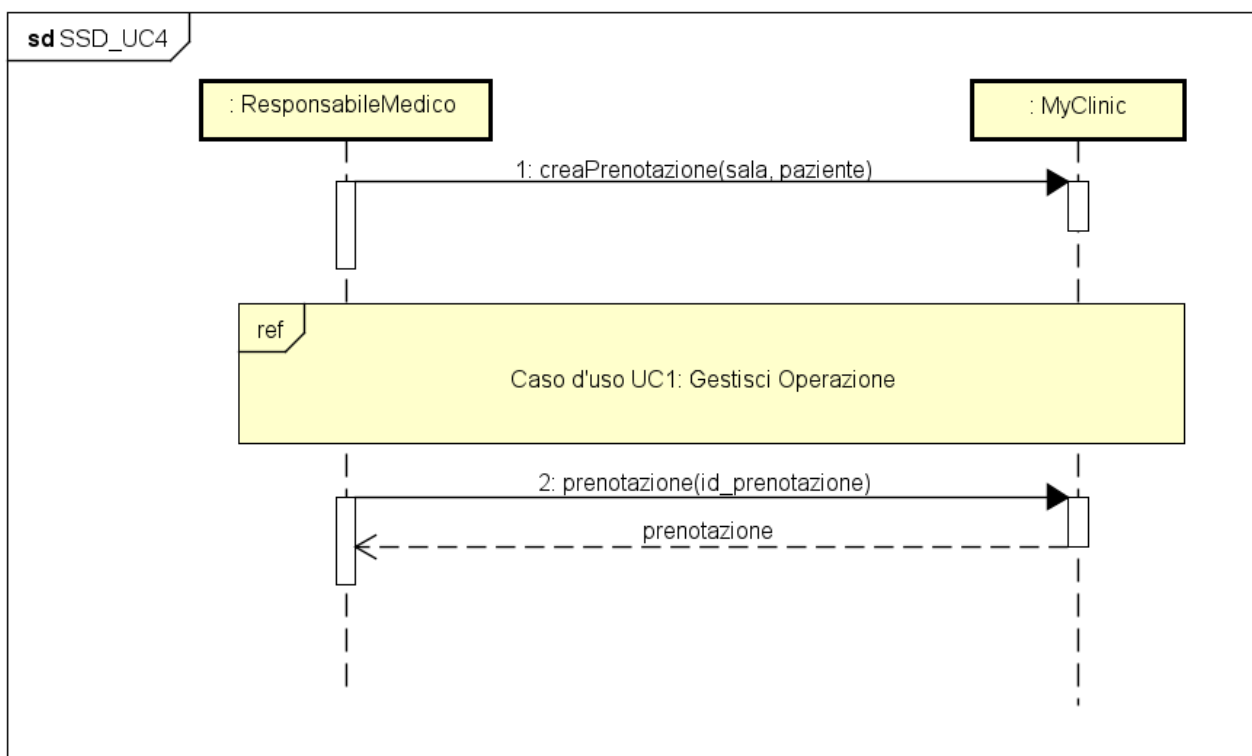
- **Operazione:**
 - *scegliPaziente(CF: Integer)*
- **Riferimenti:**
 - Caso d'uso: Gestisci Operazione
- **Pre-condizioni:**
 - È in corso un'Operazione.
- **Post-condizioni:**
 - È noto se il paziente è già presente nella lista pazienti.

Contratto C05: creaScheda(materiali, CF)

- **Operazione:**
 - *creaScheda(materiali, CF)*
- **Riferimenti:**
 - Caso d'uso: Gestisci Operazione
- **Pre-condizioni:**
 - È in corso un'Operazione.
- **Post-condizioni:**
 - È stata creata una nuova scheda che contiene il costo totale dei materiali.

➤ Iterazione 2

Il Diagramma di Sequenza di Sistema del caso d'uso *UC4* è il seguente:

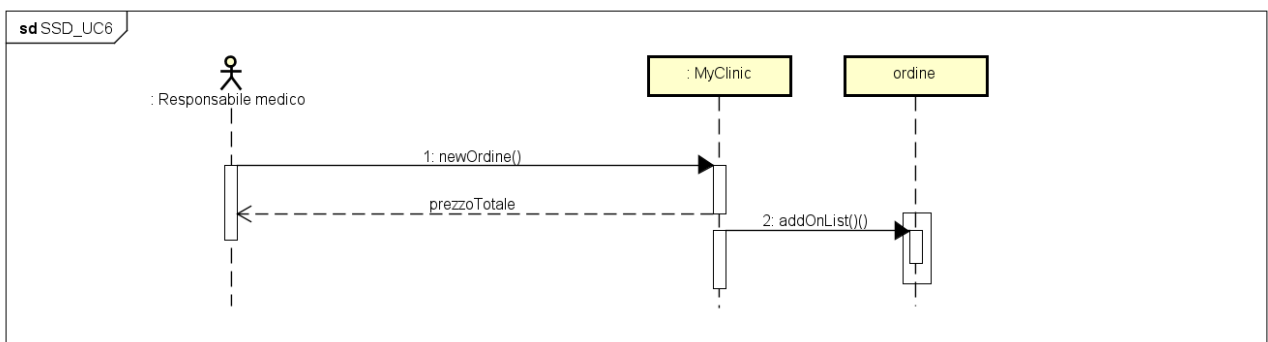


Contratto C01: creaPrenotazione(sala, paziente)

- **Operazione:**
 - *creaPrenotazione(sala, paziente)*
- **Riferimenti:**
 - Caso d'uso: Gestisci Prenotazione
- **Pre-condizioni:**
 - Non ci sono posti disponibili nella sala operatoria designata.
- **Post-condizioni:**
 - È stata inizializzata una nuova istanza *Prenotazione* legata al paziente e alla sala operatoria designata.

➤ Iterazione 3

Il Diagramma di Sequenza di Sistema del caso d'uso UC6 è il seguente:

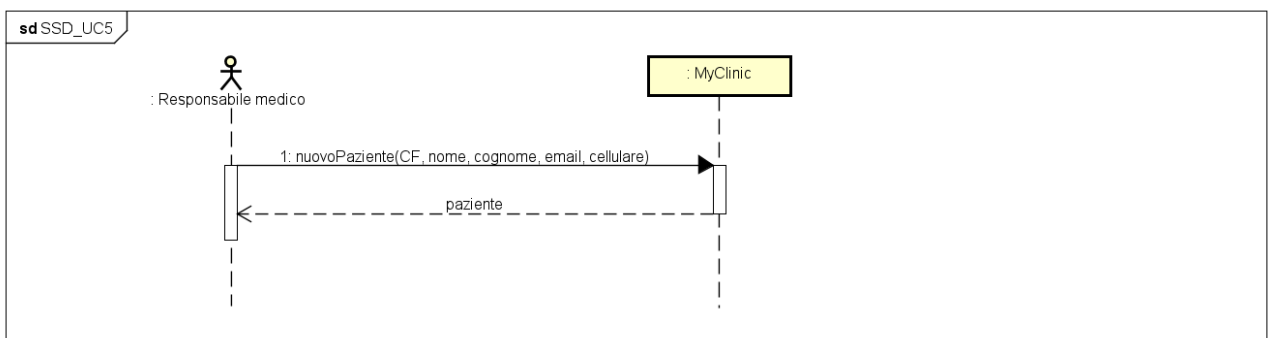


Contratto C01: newOrdine()

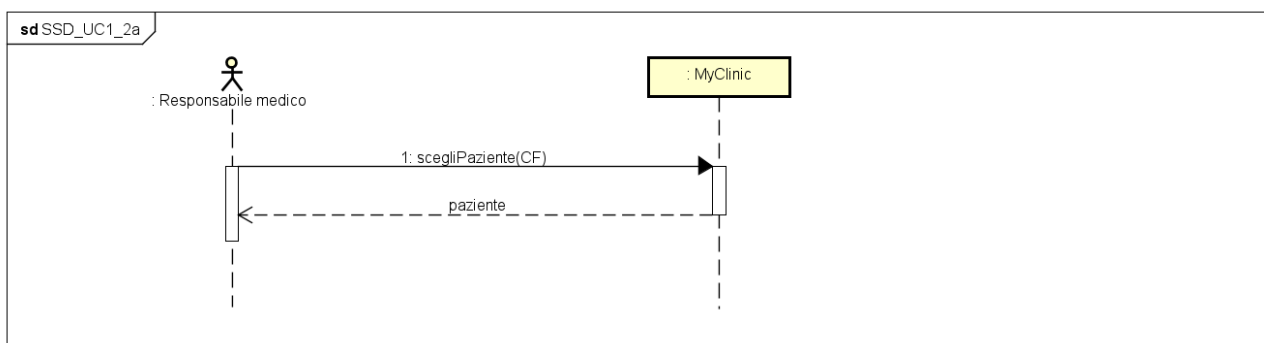
- **Operazione:**
 - *newOrdine()*
- **Riferimenti:**
 - Caso d'uso: Ordina Materiale
- **Pre-condizioni:**
 - E' richiesto un materiale e/o strumento di lavoro che non è presente nelle scorte.
- **Post-condizioni:**
 - Viene creato nuovo ordine a cui associare uno o più materiale richiesti.

➤ Iterazione 4

Il Diagramma di Sequenza di Sistema del caso d'uso UC5 è il seguente:



Invece, per quanto riguarda lo scenario alternativo **2a** di *UC1* avremo:



Contratto C01: nuovoPaziente(CF, nome, cognome, email, cellulare)

- **Operazione:**
 - *nuovoPaziente(CF, nome, cognome, email, cellulare)*
- **Riferimenti:**
 - Caso d'uso: Gestisci Paziente
- **Pre-condizioni:**
 - È presente un paziente da registrare.
- **Post-condizioni:**
 - Il paziente è stato registrato.

Contratto C02: scegliPaziente(CF)

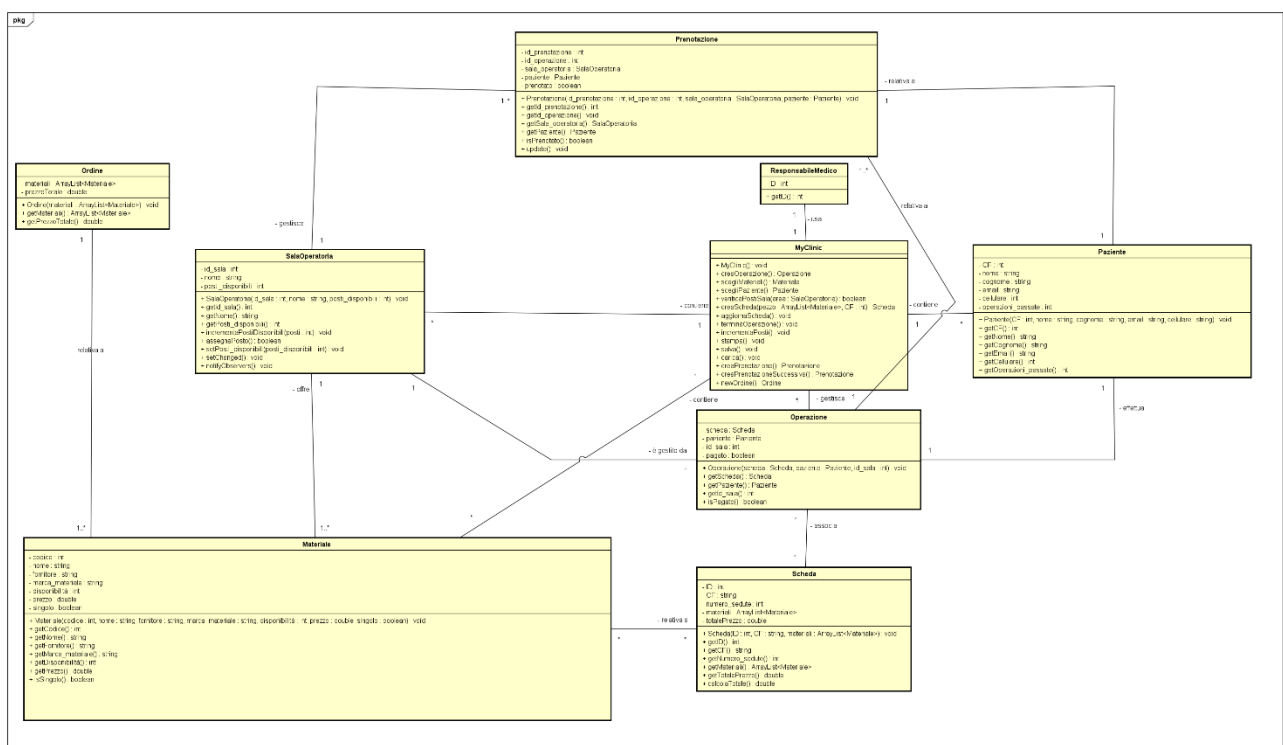
- **Operazione:**
 - *scegliPaziente(CF)*
- **Riferimenti:**
 - Caso d'uso: Gestisci Paziente
- **Pre-condizioni:**
 - C'è un paziente da ricercare.
- **Post-condizioni:**
 - Il paziente è stato trovato.

3 Progettazione

3.1 Diagramma delle Classi

La progettazione orientata agli oggetti è interessata alla definizione degli oggetti software, delle loro responsabilità e a come questi collaborano per soddisfare i requisiti individuati nei passi precedenti. L'elaborato principale che è stato preso in considerazione in questa fase è il Modello di Progetto, ovvero l'insieme dei diagrammi che descrivono la progettazione logica sia da un punto di vista dinamico (Diagrammi di Iterazione) che da un punto di vista statico (Diagramma delle Classi). Come si può notare dal Diagramma delle Classi, si è fatto uso dei pattern *Observer* (che in questo caso fa in modo che la sala operatoria comunichi alla prenotazione la disponibilità di posti liberi).

Il Diagramma delle Classi sarà:

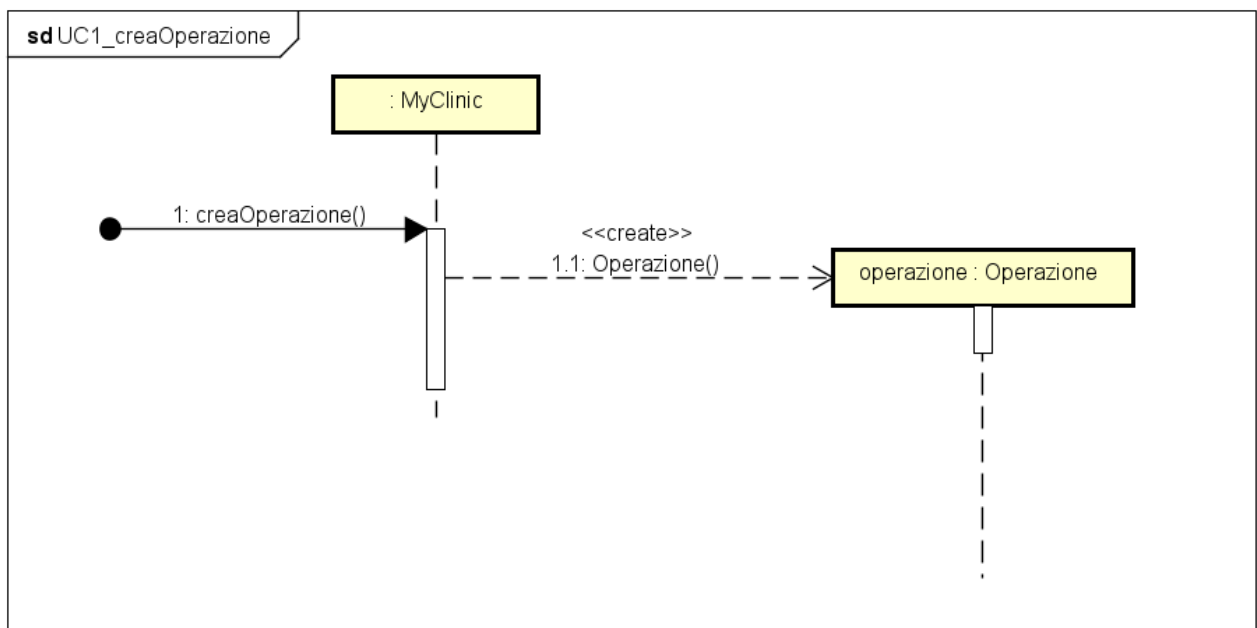


Per una maggiore visibilità, si consiglia di vedere l'allegato *Diagramma delle classi* nell'ultima iterazione.

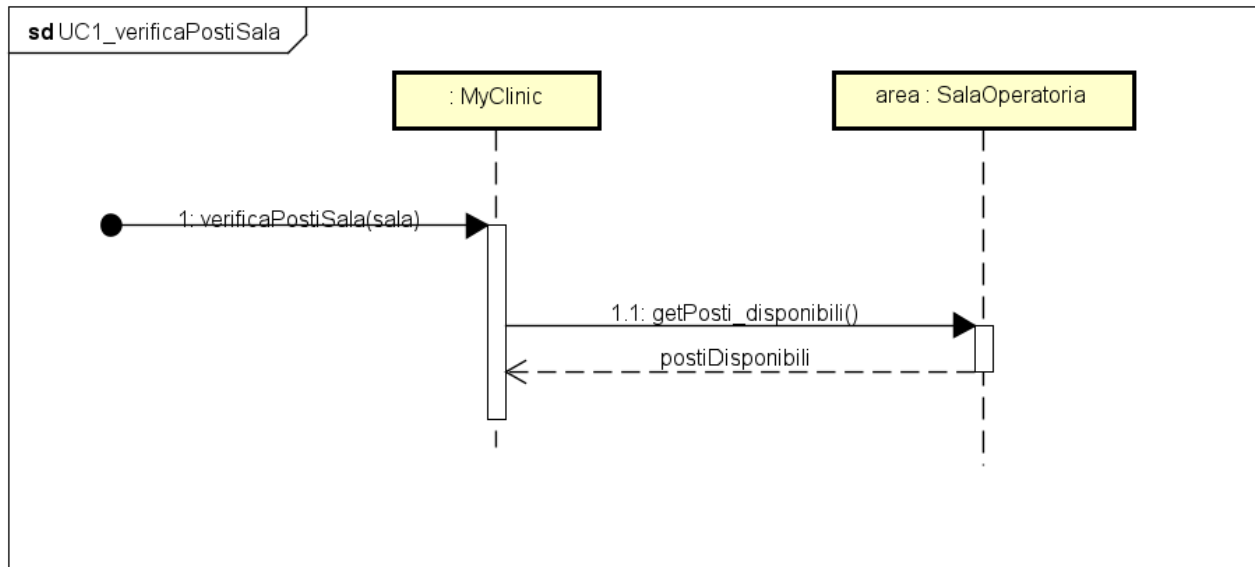
3.2 Diagramma di Sequenza

Di seguito, vengono presentati i Diagrammi di Sequenza che sono stati creati nelle varie iterazioni:

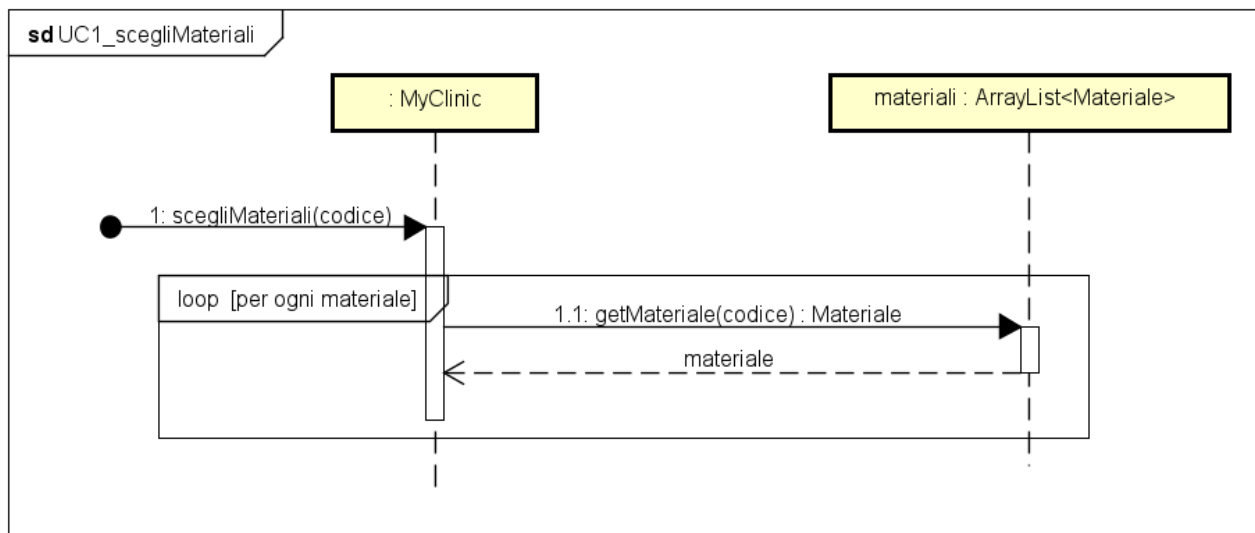
- ***creaOperazione()***



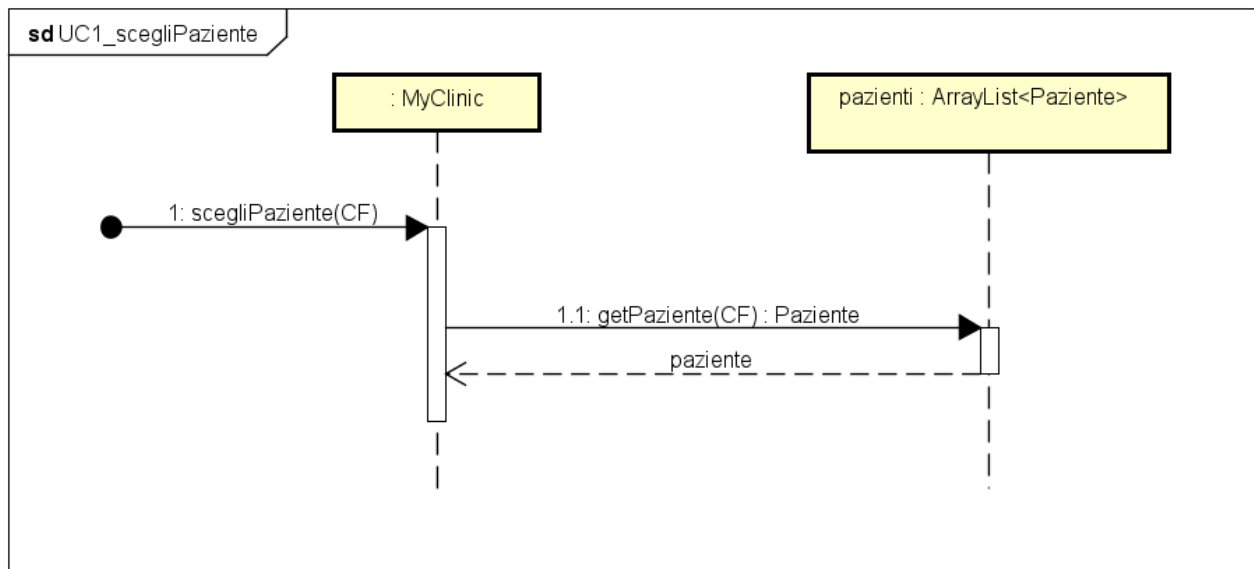
- **verificaPostiSala(sala)**



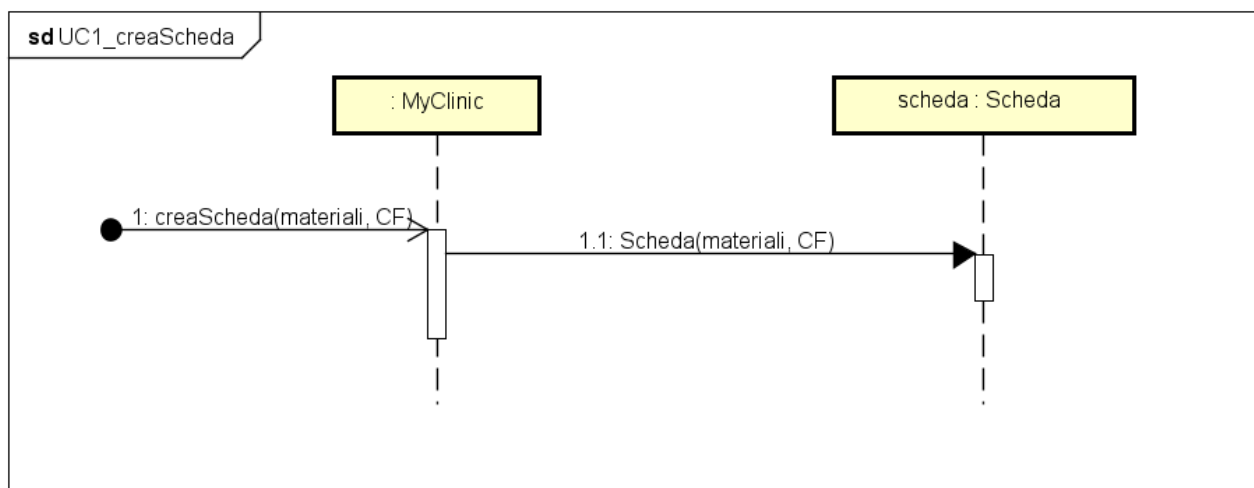
- **scegliMateriali(codice)**



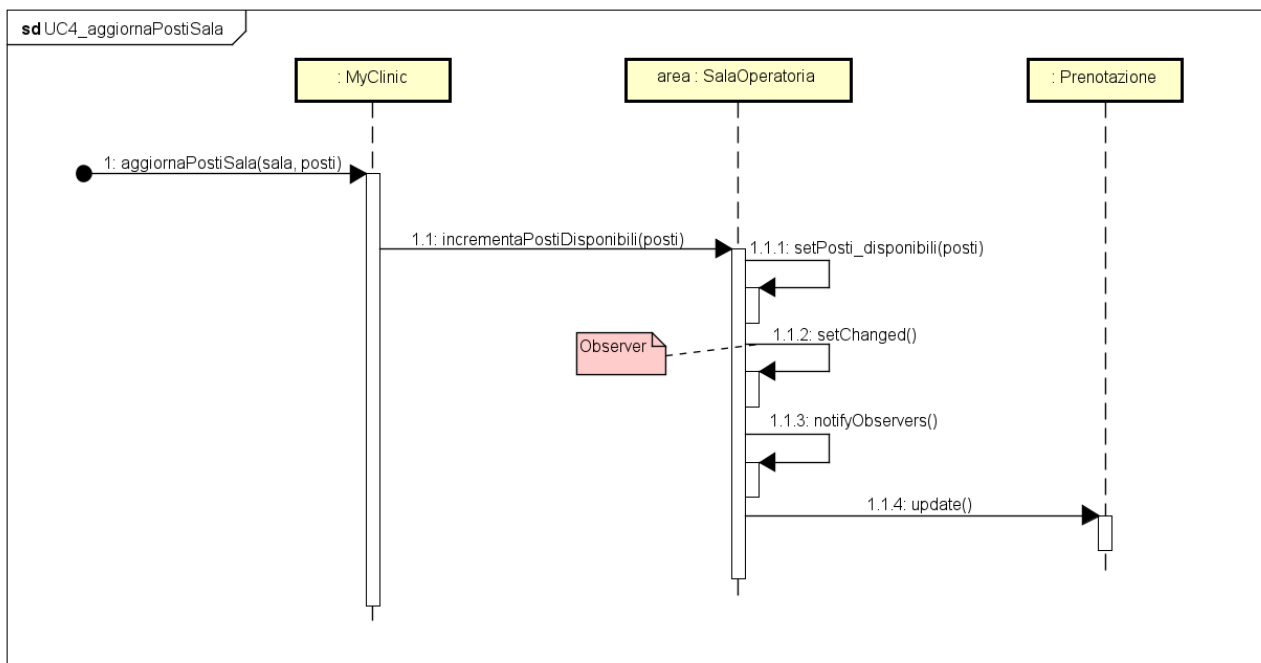
- ***scegliPaziente(CF)***



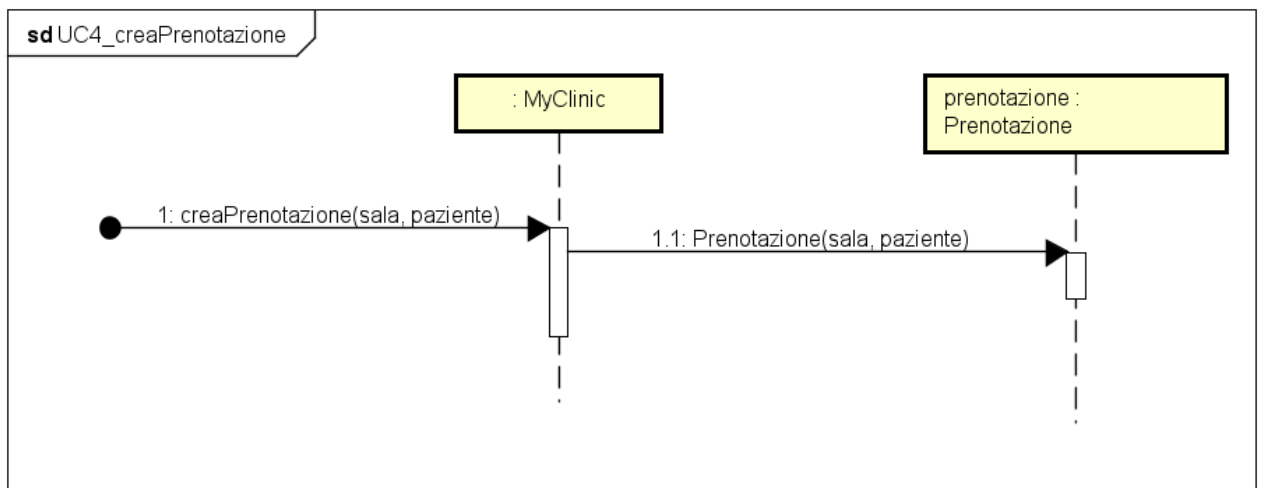
- ***creaScheda(materiali, CF)***



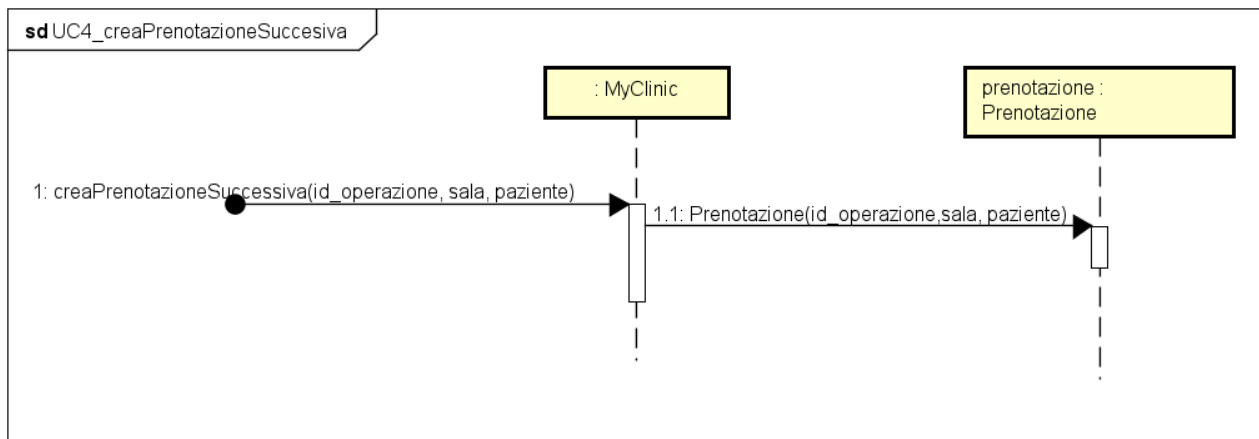
- ***aggiornaPostiSala()***



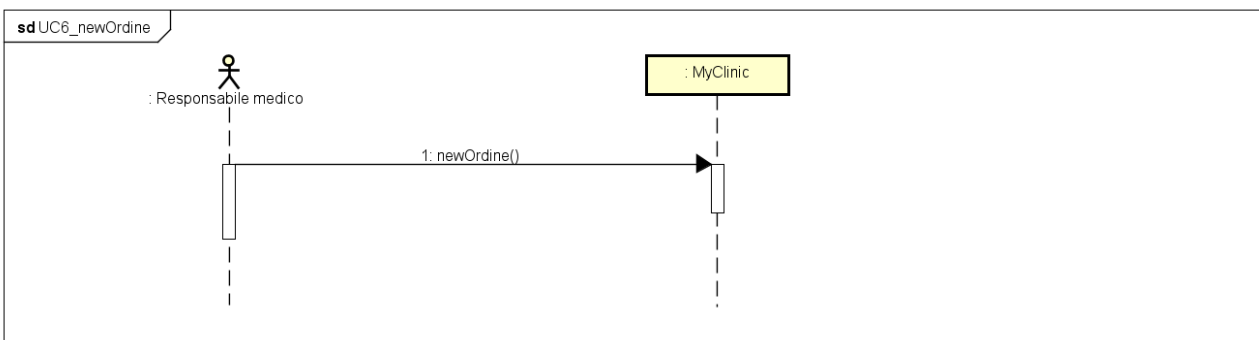
- **creaPrenotazione()**



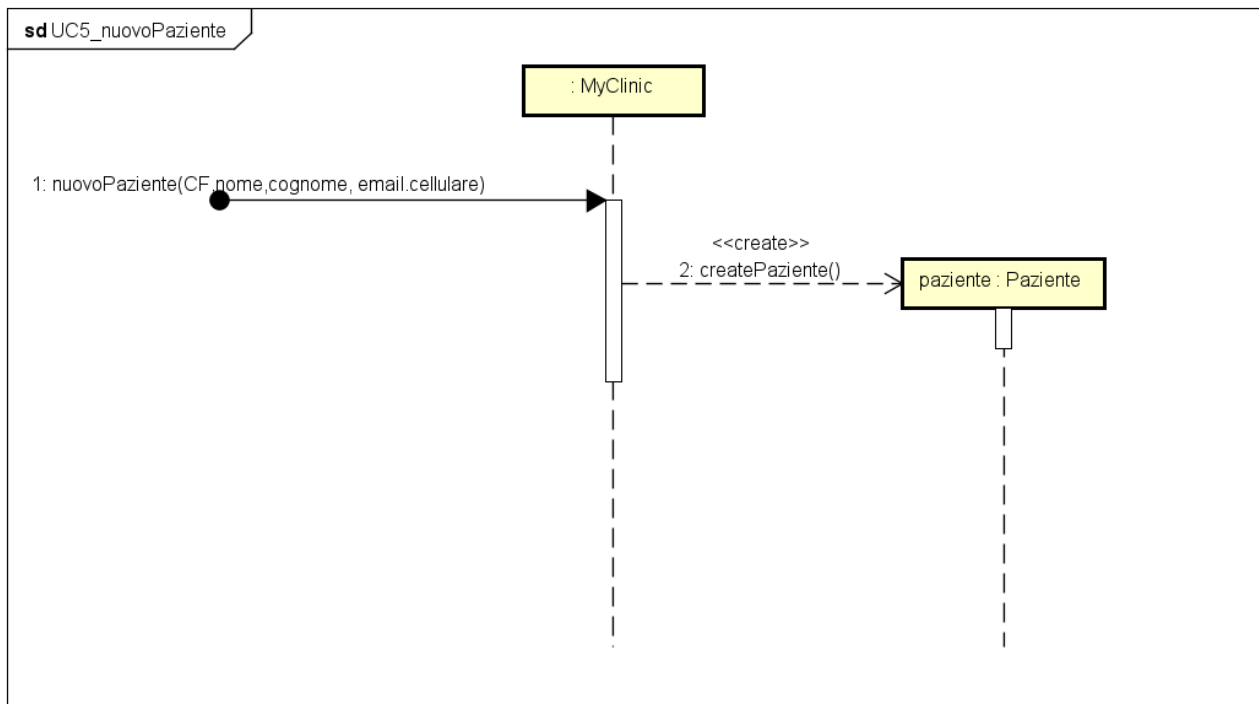
- ***creaPrenotazioneSuccessiva()***



- ***newOrdine ()***



- **nuovoPaziente()**



4 Testing

4.1 Introduzione

Il testing è una fase essenziale del processo di sviluppo di un programma robusto ed efficiente. Esso permette di ridurre in modo sostanziale i costi di manutenzione di un'applicazione. La probabilità di rilevare malfunzionamenti è proporzionale al numero di test eseguiti, tuttavia è difficile testare un programma completamente. Infatti, le possibili combinazioni di valori di input da prendere in considerazione sono enormi e non sempre possono essere riprodotte in un tempo ragionevole. Tuttavia, un testing progettato accuratamente può rivelare comportamenti anomali ed indesiderati, al fine di rendere il software realizzato funzionante e funzionale secondo le specifiche del committente.

I test si dividono principalmente in due famiglie:

1. **Test unitari** (*Unit Test*): sono semplici test che vanno a verificare la correttezza direttamente del codice, in ogni sua piccola parte. L'idea dello *Unit Test* in Java è quella di valutare ogni singolo metodo in funzione dei valori attesi. Esistono diversi tool per Unit Testing per i diversi linguaggi di programmazione, in particolare *JUnit* è il framework più diffuso attualmente per l'automazione del testing di unità di programmi Java;
2. **Test funzionali**: sono dei test che vanno a verificare che il sistema software nella sua completezza funzioni correttamente. Questi test trattano il sistema come se fosse una *black box* alla quale danno degli input e verificano la correttezza degli output.

Per eseguire il testing dell'applicazione realizzata, si è scelto di concentrarsi principalmente su test unitari eseguiti tramite l'ausilio di JUnit.

4.2 Individuazione dei casi di test e Testing Unitario

In una fase preliminare al testing, è stata affrontata una discussione e ispezione del codice sorgente scritto, al fine di operare una scelta sulle classi e sui metodi da testare. Il criterio scelto per individuare tali metodi su cui concentrare i test di unità è stato quello di dare priorità a tutti i metodi che si occupano dell'inserimento di nuove istanze, dell'incremento dei posti disponibili e del calcolo dei fattori economici (prezzo totale). Per ciascuno sono state individuate le classi di equivalenza tali da poter ridurre il numero di test al minimo insieme necessario, in particolare:

➤ SalaOperatoria

- *incrementaPostiDisponibili*:
 - caso in cui si incrementa con numero positivo;
 - caso in cui si incrementa numero negativo. In questo caso, l'incremento non avviene.
- *assegnaPosto*:
 - caso in cui si assegna un posto in una sala operatoria se disponibile;
 - caso in cui l'assegnazione non avviene perché la sala operatoria è piena.

➤ Scheda

- *calcolaTotale*:
 - caso standard con prezzi positivi dei vari materiali e per le varie sedute;
 - caso in cui si hanno materiali con prezzi negativi e questi non sono considerati nel calcolo.

A seguito dell'esecuzione di questi test unitari con *JUnit*, si è passati a correggere le relative porzioni di codice che portavano al fallimento del sistema di alcuni test.

4.3 Test di Sistema

In previsione di una fase di refactoring in cui verrà integrato un database con l'applicazione esistente, si è scelto di eseguire dei test di sistema manuali lanciando l'applicazione e provando a portare a termine i principali casi d'uso. A seguito di tali test, non sono emersi grandi errori nel funzionamento e le prestazioni del sistema si sono mantenute piuttosto elevate. Tuttavia, sono state rilevate delle imperfezioni e anomalie nell'esecuzione di alcune funzionalità:

- Terminata un'operazione, non veniva liberato il posto occupato nella specifica sala operatoria;
- Selezionando i materiali per un'operazione non era possibile selezionare più volte lo stesso materia

5 Refactoring e Conclusioni

5.1 Database e Refactoring

In questa fase conclusiva, le operazioni prevalentemente svolte sono state:

- La correzione delle anomalie individuate con il precedente Test di Sistema;
- L'integrazione nel software di un database per gestire la persistenza dei dati. In particolare, il database utilizzato è un DB locale usando *phpMyAdmin* per creare la tabella Pazienti. L'interfacciamento del DB con l'applicazione è stato realizzato con la libreria *Hibernate ORM*;
- L'aggiunta di alcune funzionalità secondarie dell'applicazione, quali la visualizzazione e il salvataggio in locale delle componenti presenti nel sistema, quali: operazioni, prenotazioni, materiali, schede e prezzi.

5.2 Test di Accettazione

A conclusione delle attività necessarie per portare a completamento i passi precedenti e dopo un'attenta revisione di tutto il codice sorgente, è stato eseguito il test di accettazione conclusivo. Simulando di ritrovarsi nei panni dell'utente (ovvero, del responsabile medico), sono state testate le principali funzionalità del programma, al fine di dimostrare che è possibile portare a termine con successo i casi d'uso implementati.