

Luca Dimarco – Antonino Alessio Salemi



A.A. 2021-2022
CORSO DI LAUREA MAGISTRALE IN INGEGNERIA INFORMATICA
Ingegneria del Software

PREFAZIONE

Il seguente documento presenta una descrizione relativa allo sviluppo dell'applicazione OverClock implementata durante il corso di Ingegneria del Software. Il software è stato implementato nel linguaggio Java sfruttando l'ambiente di sviluppo NetBeans, mentre la progettazione in UML è stata realizzata con il software Astah Professional. È stato poi utilizzato un database SQL in combinazione al pattern architetturale DAO (Data Access Object) per la persistenza dei dati. 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 state apportate delle migliorie al codice.

SOMMARIO

| | | |
|-------|--|----|
| 1 | Ideazione e analisi dei requisiti..... | 5 |
| 1.1 | Introduzione | 5 |
| 1.2 | Requisiti..... | 5 |
| 1.3 | Obiettivi e casi d'uso | 6 |
| 1.4 | Modello dei casi d'uso | 7 |
| | UC1: Gestisci Vendita..... | 7 |
| | UC2: Crea Preventivo Riparazione..... | 9 |
| | UC3: Aggiorna Preventivo Riparazione | 11 |
| | UC4: Avvio Riparazione | 12 |
| | UC5: Gestisci Stato Riparazione..... | 13 |
| | UC6: Gestisci Acquisto Usato | 14 |
| | UC7: Crea Ordine Distributore | 15 |
| | UC8: Gestisci Dispositivo (CRUD) | 16 |
| | UC9: Gestisci Cliente (CRUD) | 17 |
| | UC10: Gestisci Distributore (CRUD) | 18 |
| | UC11: Crea Prenotazione | 19 |
| | UC12: Gestisci Stato Prenotazione | 20 |
| | UC13: Ricerca Vendita (CRUD) | 21 |
| | UC14: Gestisci Carta Fedeltà (CRUD) | 22 |
| | UC15: Applica Sconto Carta | 23 |
| | UC16: Gestisci Promozione..... | 24 |
| 1.5 | Documento di Visione..... | 25 |
| 1.6 | Regole di business..... | 25 |
| 1.7 | Specifiche Supplementari | 26 |
| 1.7.1 | Usabilità | 26 |
| 1.7.2 | Affidabilità | 26 |
| 1.7.3 | Vincoli hardware e software | 26 |
| 1.7.4 | Vincoli di sviluppo del software | 26 |
| 1.7.5 | Aspetti legali..... | 26 |
| 1.8 | Glossario | 27 |
| 2 | Analisi Orientata agli oggetti..... | 28 |
| 2.1 | Introduzione | 28 |
| 2.2 | Modello di Dominio..... | 29 |

| | | |
|-----|--|----|
| 2.3 | SSD e Contratti..... | 30 |
| | Scenario principale del caso d'uso UC1: Gestisci Vendita | 30 |
| | ___ Contratto CO1: ricercaDispositivo(nome) | 30 |
| | ___ Contratto CO2: aggiungiDispositivo(dispositivo, quantità) | 31 |
| | ___ Contratto CO3: associaCarta(idCarta) | 31 |
| | ___ Contratto CO4: registraVendita(id, punti) | 31 |
| | ___ Contratto CO4: registraVendita() | 32 |
| | Scenario principale del caso d'uso UC7: Crea Ordine Distributore..... | 32 |
| | Scenario principale del caso d'uso UC11: Crea Prenotazione | 33 |
| | ___ Contratto CO1: aggiungiDispositivo(dispositivo, quantità) | 33 |
| | ___ Contratto CO2: nuovoOrdine(schedaOrdine) | 33 |
| | ___ Contratto CO3: inviaOrdine() | 34 |
| | ___ Contratto CO4: ricercaCliente(email) | 34 |
| | ___ Contratto CO5: aggiungiPrenotazione(cliente,dispositivo) | 34 |
| | Scenario principale del caso d'uso UC2: Crea Preventivo Riparazione | 35 |
| | Scenario principale del caso d'uso UC3: Aggiorna Preventivo Riparazione | 35 |
| | Scenario principale del caso d'uso UC4: Avvio Riparazione | 36 |
| | Scenario principale del caso d'uso UC4: Avvio Riparazione Alternativo | 36 |
| | ___ Contratto CO1: nuovaRiparazione(cliente, dispositivo) | 37 |
| | ___ Contratto CO2: ricercaRiparazione(email) | 37 |
| | ___ Contratto CO3: selezionaStatoRiparazione(stato) | 37 |
| | ___ Contratto CO4: aggiornaRiparazione(schedaRiparazione,stato,listaPezzi,costo) | 37 |
| | ___ Contratto CO5: notificaCliente(messaggio,pezzi,email) | 38 |
| | Scenario principale del caso d'uso UC6: Gestisci Acquisto Usato | 38 |
| | ___ Contratto CO1: aggiungiAcquisto(cliente, dispositivo, danni) | 39 |
| | Scenario principale del caso d'uso UC16: Gestisci Promozione | 39 |
| | ___ Contratto CO1: aggiungiPromozione(dispositivo, sconto) | 39 |
| 3 | Progettazione..... | 41 |
| 3.1 | Diagramma delle Classi | 41 |
| 3.2 | Diagrammi di Sequenza | 42 |
| | Ricerca di un dispositivo | 42 |
| | Aggiunta di un dispositivo al carrello..... | 42 |
| | Associazione carta fedeltà alla vendita | 42 |
| | Conferma e registrazione della vendita | 43 |
| | Aggiunta di un dispositivo all'ordine..... | 43 |
| | Conferma e registrazione dell'ordine | 43 |

| | |
|---|----|
| Invio dell'e-mail ai distributori interessati nell'ordine | 44 |
| Ricerca del cliente in registro | 44 |
| Conferma e registrazione della prenotazione | 44 |
| Observer per verificare lo stato della prenotazione | 45 |
| Conferma e registrazione del preventivo | 45 |
| Conferma e registrazione acquisto usato..... | 45 |
| Conferma e registrazione della promozione | 45 |
| 4 Testing | 46 |
| 4.1 Introduzione | 46 |
| 4.2 Individuazione dei casi di test e Testing Unitario | 46 |
| 4.3 Test di Sistema..... | 49 |
| 5 Refactoring e Conclusioni..... | 50 |
| 5.1 Database e Refactoring | 50 |
| 5.2 Test di accettazione | 50 |

1 IDEAZIONE E ANALISI DEI REQUISITI

1.1 INTRODUZIONE

Durante la fase di ideazione viene effettuata un'analisi generale del progetto da realizzare per capirne la fattibilità e stimarne i tempi e le risorse necessarie. In questa fase sono stati presi in considerazione diversi moduli per comprendere in modo esaustivo i requisiti del progetto ed il modo in cui svilupparlo. Nello specifico, verranno mostrati i documenti seguenti: Modello dei Casi d'Uso, Documento di Visione, Specifiche Supplementari, Regole di Business e Glossario.

1.2 REQUISITI

Il titolare di un negozio di vendita e riparazioni di prodotti elettronici richiede un software per migliorare la gestione degli articoli e dei pezzi di ricambio. Il software deve essere uno strumento integrato che tiene traccia dei dispositivi nuovi e usati disponibili alla vendita, quelli in riparazione e quelli riparati in attesa di ritiro da parte dei clienti. In particolare:

- Il titolare deve poter gestire l'anagrafica dei clienti e tenere traccia degli acquisti effettuati e delle riparazioni richieste da essi.
- Il titolare deve poter consultare e modificare un catalogo di articoli disponibili alla vendita, anche se non presenti in magazzino.
- Il titolare deve poter consultare e modificare un registro di articoli in riparazione.
- Il titolare deve poter consultare e modificare un catalogo di pezzi di ricambio, anche se non presenti in magazzino.
- I clienti devono poter richiedere un articolo al titolare; il sistema dovrà verificare l'esistenza dell'articolo nel catalogo e la disponibilità nel magazzino. L'esito della ricerca viene comunicato al cliente, che potrà scegliere se:
 - a. Acquistare immediatamente l'articolo, qualora sia presente.
 - b. Prenotare l'articolo, qualora non sia presente.
 - c. Non acquistare né prenotare l'articolo.
- Nel caso a. viene creata una nuova scheda di vendita relativa alle richieste del cliente; il sistema calcolerà l'importo da pagare, dopo aver applicato i relativi sconti, e lo mostrerà a video. Nel caso b. verrà creata una prenotazione con i dati del cliente e dell'articolo. Se il cliente procede alla conferma, il sistema inoltrerà le informazioni alla cassa e le operazioni verranno concluse con il pagamento.
- Nel caso a. il cliente potrà ritirare l'articolo immediatamente dopo il pagamento; il titolare aggiornerà la scheda di vendita, specificando che è avvenuta la consegna.
- Nel caso b. il cliente andrà via dopo il pagamento in attesa di una comunicazione da parte del titolare che notificherà l'arrivo dell'articolo richiesto e la possibilità di ritirarlo.
- I clienti devono poter richiedere una riparazione di un dispositivo al titolare; il titolare dovrà verificare i danni del dispositivo e gli eventuali pezzi di ricambio che servono; il sistema dovrà calcolare il preventivo e verificare la presenza dei pezzi di ricambio in magazzino. L'esito viene comunicato al cliente, che potrà scegliere se:
 - a. Accettare il preventivo e lasciare il dispositivo in riparazione.
 - b. Non accettare il preventivo e non lasciare il dispositivo.
- Nel caso a. il cliente verrà notificato una volta terminata la riparazione.

- I clienti devono poter vendere articoli usati al titolare in cambio di punti sulla carta fedeltà; il sistema dovrà calcolare se è acquistabile e i punti fedeltà da proporre al cliente in base al tipo di articolo proposto e al grado di danneggiamento. L'esito del calcolo verrà proposto al cliente, che potrà scegliere se:
 - a. Vendere l'articolo, che verrà aggiunto al catalogo degli articoli disponibili alla vendita.
 - b. Non vendere l'articolo.
- Il titolare deve poter creare degli ordini per i distributori tenendo conto delle richieste da parte dei clienti ma anche in modo arbitrario scegliendoli dal catalogo.
- All'arrivo degli articoli ordinati ai distributori, il titolare deve poter aggiornare la loro disponibilità nel catalogo e notificare i clienti, nel caso in cui si tratti di articoli da loro richiesti.
- Il titolare deve poter notificare i clienti dell'avvenuta riparazione dei loro dispositivi lasciati in negozio.
- Il titolare deve poter creare e gestire sconti e promozioni; il sistema deve applicarli durante la vendita.

1.3 OBIETTIVI E CASI D'USO

Facendo riferimento ai requisiti riportati nel paragrafo precedente, si può individuare l'attore principale che utilizzerà il sistema. Analizzando gli obiettivi che si propone di raggiungere, sono stati ricavati i casi d'uso principali.

| Attore | Obiettivo | Caso d'uso |
|----------|--|--------------------------------------|
| Titolare | Gestire la creazione di una nuova vendita di uno o più dispositivi ad un cliente, tenendo conto di eventuali prenotazioni e promozioni. | UC1: Gestisci Vendita |
| Titolare | Creare un preventivo (con tempi, costi e dettagli di riparazione) di un dispositivo di un cliente. | UC2: Crea Preventivo Riparazione |
| Titolare | Modificare i dettagli di un preventivo di riparazione di un dispositivo e notificare il cliente. | UC3: Aggiorna Preventivo Riparazione |
| Titolare | Avviare la riparazione di un dispositivo di un cliente, tenendo conto di eventuali prenotazioni di pezzi di ricambio. | UC4: Avvio Riparazione |
| Titolare | Verificare lo stato delle riparazioni in corso e notificare i clienti, qualora siano completate. | UC5: Gestisci Stato Riparazione |
| Titolare | Gestire l'acquisto di uno o più dispositivi usati di un cliente ed inserirlo/i in magazzino pronto/i per la vendita. | UC6: Gestisci Acquisto Usato |
| Titolare | Creare un ordine destinato ad un distributore per richiedere dispositivi e/o pezzi di ricambio, richiesti dal cliente o da aggiungere al catalogo. | UC7: Crea Ordine Distributore |

| | | |
|----------|--|-------------------------------------|
| Titolare | Inserire, modificare, ricercare o rimuovere un dispositivo nel catalogo. | UC8: Gestisci Dispositivo (CRUD) |
| Titolare | Inserire, modificare, ricercare o rimuovere i dati di un cliente. | UC9: Gestisci Cliente (CRUD) |
| Titolare | Inserire, modificare, ricercare o rimuovere un distributore. | UC10: Gestisci Distributore (CRUD) |
| Titolare | Creare una nuova prenotazione di uno o più dispositivi associata ad un cliente. | UC11: Crea Prenotazione |
| Titolare | Verificare le prenotazioni in sospeso e controllare la disponibilità in magazzino. Inviare eventuali notifiche ai clienti. | UC12: Gestisci Stato Prenotazione |
| Titolare | Ricercare una vendita, tramite codice identificativo, o tutte le vendite relative ad un cliente. | UC13: Ricerca Vendita (CRUD) |
| Titolare | Creare, modificare, rimuovere una carta o leggerne i punti disponibili. | UC14: Gestisci Carta Fedeltà (CRUD) |
| Titolare | Verificare lo sconto disponibile tramite carta fedeltà ed eventualmente associarlo a una vendita. | UC15: Applica Sconto Carta |
| Titolare | Ricercare, creare, modificare o rimuovere una promozione. | UC16: Gestisci Promozione (CRUD) |

1.4 MODELLO DEI CASI D'USO

Di seguito viene riportata una descrizione in dettaglio dei casi d'uso, a partire dalle definizioni in breve riportate nel paragrafo precedente. Pertanto, si ha:

UC1: Gestisci Vendita

| | |
|--------------------------------------|--|
| Nome del caso d'uso | UC1: Gestisci Vendita |
| Portata | Sistema OverClock |
| Livello | Obiettivo utente |
| Attore primario | Titolare |
| Parti interessate e interessi | <ul style="list-style-type: none"> - Titolare del negozio: vuole gestire le vendite in maniera chiara e veloce; vuole che le informazioni relative alla vendita dei dispositivi da parte dei clienti siano registrate e aggiornate. - Cliente: vuole poter acquistare i dispositivi con rapidità e semplicità. |
| Pre-condizioni | Il cliente ha deciso cosa acquistare. |

| | |
|--|---|
| Garanzia di successo | Il titolare viene pagato dal cliente che eventualmente ritira il/i dispositivo/i; il sistema aggiorna le giacenze dei dispositivi. |
| Scenario principale di successo | <ol style="list-style-type: none"> 1. Un cliente arriva in negozio e richiede al titolare uno o più dispositivi da acquistare. 2. Il titolare cerca il dispositivo richiesto nel catalogo [vedi UC8: Gestisci Dispositivo]. Il sistema mostra l'esito positivo della ricerca e le giacenze residue al titolare, che informa il cliente della disponibilità. 3. Il cliente intende acquistare il dispositivo. 4. Il titolare utilizzando OverClock aggiunge il dispositivo ricercato alla scheda di vendita. 5. Ripetere i passi 2, 3 e 4 finché ci sono altri dispositivi che il cliente desidera acquistare. 6. Il cliente comunica di voler terminare l'acquisto; il titolare conferma l'operazione. 7. Il sistema calcola il totale dell'acquisto, applica eventuali sconti [vedi Regole di business] e mostra a video il risultato. 8. Il titolare chiede al cliente la carta fedeltà [vedi UC14: Gestisci Carta Fedeltà]. 9. Il sistema applica lo sconto associato ai punti della carta fedeltà [vedi UC15: Applica Sconto Carta; vedi Regole di business]. 10. Il titolare comunica al cliente l'importo totale dell'ordine. 11. Il cliente decide di proseguire l'acquisto e il sistema inoltra alla cassa le informazioni contenute nella scheda di vendita; il cliente paga e ritira la ricevuta. 12. Il titolare conferma l'acquisto; il sistema converte il conto totale in punti fedeltà, aggiorna il saldo della carta fedeltà del cliente aggiungendo un punto per ogni euro di spesa (arrotondato a numero intero per difetto) [vedi UC14: Gestisci Carta Fedeltà; vedi Regole di business] e aggiorna le giacenze residue in magazzino [vedi UC8: Gestisci Dispositivo]. 13. Il cliente ritira il/i dispositivo/i e va via. |
| Estensioni | <p>*a. In qualsiasi momento, il sistema fallisce e ha un arresto improvviso.</p> <ol style="list-style-type: none"> 1. Il titolare riavvia il software e richiede il ripristino dello stato precedente del sistema. 2. Il sistema ricostruisce lo stato precedente. <p>2a. La ricerca del dispositivo nel catalogo non ha prodotto alcun risultato.</p> <ol style="list-style-type: none"> 1. Il titolare comunica al cliente che il dispositivo non è presente nel catalogo. 2. Se vi sono altri dispositivi, lo scenario riprende dal punto 5. Altrimenti, il titolare annulla la scheda di vendita e si conclude. |

| | |
|--|---|
| | <p>2b. Il prodotto ricercato non è disponibile in magazzino e il titolare informa il cliente.</p> <ol style="list-style-type: none"> Il cliente comunica al titolare che intende prenotare il dispositivo [vedi UC11: Crea Prenotazione]. <ol style="list-style-type: none"> Se vi sono altri dispositivi, lo scenario riprende dal punto 5. Altrimenti, il titolare annulla la scheda di vendita e si conclude. Se vi sono altri dispositivi, lo scenario riprende dal punto 5. Altrimenti, il titolare annulla la scheda di vendita e si conclude. <p>3a. Il cliente non intende acquistare il dispositivo.</p> <ol style="list-style-type: none"> Se vi sono altri dispositivi, lo scenario riprende dal punto 5. Altrimenti, il titolare annulla la scheda di vendita e si conclude. <p>8a. Il cliente non possiede la carta fedeltà.</p> <ol style="list-style-type: none"> Il titolare chiede al cliente se intende fare la carta fedeltà; il cliente accetta [vedi UC14: Gestisci Carta Fedeltà]. <ol style="list-style-type: none"> Il titolare chiede al cliente se intende fare la carta fedeltà; il cliente rifiuta. <ol style="list-style-type: none"> Lo scenario riprende dal punto 12. <p>11a. Il cliente decide di non proseguire l'acquisto e va via.</p> <ol style="list-style-type: none"> Il titolare annulla la scheda di vendita e termina l'esecuzione del software. <p>13a. I dispositivi acquistati non erano disponibili in magazzino.</p> <ol style="list-style-type: none"> Il cliente va via in attesa di una comunicazione da parte del titolare che lo informerà dell'arrivo del/dei dispositivo/i prenotato/i [vedi UC12: Gestisci Stato Prenotazione]. |
| Requisiti speciali | |
| Elenco delle varianti tecnologiche e dei dati | |
| Frequenza di ripetizioni | Legata all'affluenza dei clienti e agli acquisti che intendono effettuare. |
| Varie | |

UC2: Crea Preventivo Riparazione

| | |
|----------------------------|--------------------------------------|
| Nome del caso d'uso | UC2: Gestisci Preventivo Riparazione |
| Portata | Sistema OverClock |
| Livello | Obiettivo utente |
| Attore primario | Titolare |

| | |
|--|--|
| Parti interessate e interessi | <ul style="list-style-type: none"> - Titolare del negozio: vuole gestire i preventivi in maniera chiara e veloce; vuole che le informazioni relative ai preventivi dei dispositivi dei clienti siano registrate e aggiornate. - Cliente: vuole conoscere tempi e costi di riparazione del proprio dispositivo con rapidità e semplicità. |
| Pre-condizioni | Il cliente ha uno o più dispositivi da riparare. |
| Garanzia di successo | Il cliente lascia il dispositivo da riparare al titolare, che lo visionerà per valutare il preventivo. |
| Scenario principale di successo | <ol style="list-style-type: none"> 1. Un cliente arriva in negozio e richiede al titolare uno o più preventivi per ogni dispositivo che intende riparare. 2. Il titolare apre una nuova scheda preventivo. 3. Il titolare chiede al cliente i dati personali; Il cliente li comunica. 4. Il titolare ricerca il cliente nell'archivio interno [vedi UC9: Gestisci Cliente]; il software restituisce il risultato richiesto (cliente già presente nell'archivio). 5. Il sistema associa il cliente alla scheda preventivo. 6. Il titolare chiede al cliente il modello del dispositivo da riparare; Il cliente lo comunica. 7. Il titolare ricerca il modello del dispositivo nel catalogo del sistema [vedi UC8: Gestisci Dispositivo]; Il sistema mostra l'esito positivo della ricerca al titolare che informa il cliente della disponibilità alla riparazione. 8. Il titolare conferma la scheda preventivo ed essa viene salvata nel sistema. 9. Ripetere i passi 2, 5, 6, 7 e 8 per ogni dispositivo che il cliente desidera riparare. 10. Il cliente lascia il/i dispositivo/i e va via in attesa dei dettagli sul preventivo. |
| Estensioni | <p>*a. In qualsiasi momento, il sistema fallisce e ha un arresto improvviso.</p> <ol style="list-style-type: none"> 1. Il titolare riavvia il software e richiede il ripristino dello stato precedente del sistema. 2. Il sistema ricostruisce lo stato precedente. <p>4a. La ricerca del cliente nell'archivio interno non ha prodotto alcun risultato.</p> <ol style="list-style-type: none"> 1. Il sistema chiede al titolare di inserire i dati del cliente prima di proseguire. 2. Il titolare inserisce i dati del cliente [vedi UC9: Gestisci Cliente]. 3. Il sistema associa il cliente appena inserito alla scheda preventivo. 4. Lo scenario riprende dal punto 6. <p>7a. La ricerca del dispositivo nel catalogo non ha prodotto alcun risultato.</p> <ol style="list-style-type: none"> 1. Il titolare comunica al cliente che il dispositivo non è riparabile. |

| | |
|--|--|
| | 2. Il titolare chiude la scheda preventivo e lo scenario riprende dal punto 9. |
| Requisiti speciali | |
| Elenco delle varianti tecnologiche e dei dati | |
| Frequenza di ripetizioni | Legata all'affluenza dei clienti e alle riparazioni che intendono effettuare. |
| Varie | |

UC3: Aggiorna Preventivo Riparazione

| | |
|--|--|
| Nome del caso d'uso | UC3: Aggiorna Preventivo Riparazione |
| Portata | Sistema OverClock |
| Livello | Obiettivo utente |
| Attore primario | Titolare |
| Parti interessate e interessi | <ul style="list-style-type: none"> - Titolare del negozio: vuole modificare le informazioni relative ai preventivi e notificare i clienti in maniera rapida. - Cliente: vuole essere informato riguardo ai dettagli del preventivo, associato al proprio dispositivo da riparare, in maniera chiara. |
| Pre-condizioni | Il titolare ha visionato il dispositivo da riparare. |
| Garanzia di successo | Il cliente viene a conoscenza del preventivo. |
| Scenario principale di successo | <ol style="list-style-type: none"> 1. Il titolare ricerca nel sistema la scheda preventivo relativa al dispositivo che ha visionato. 2. Il titolare aggiorna le informazioni relative al guasto e inserisce il costo della riparazione; Il sistema notifica il cliente. |
| Estensioni | <p>*a. In qualsiasi momento, il sistema fallisce e ha un arresto improvviso.</p> <ol style="list-style-type: none"> 1. Il titolare riavvia il software e richiede il ripristino dello stato precedente del sistema. 2. Il sistema ricostruisce lo stato precedente. |
| Requisiti speciali | |
| Elenco delle varianti tecnologiche e dei dati | |
| Frequenza di ripetizioni | Legata alla quantità di dispositivi visionati dal titolare. |
| Varie | |

UC4: Avvio Riparazione

| | |
|--|--|
| Nome del caso d'uso | UC4: Avvio Riparazione |
| Portata | Sistema OverClock |
| Livello | Obiettivo utente |
| Attore primario | Titolare |
| Parti interessate e interessi | <ul style="list-style-type: none"> - Titolare del negozio: vuole gestire le riparazioni in maniera chiara e veloce; vuole che le informazioni relative alle riparazioni dei dispositivi da parte dei clienti siano registrate e aggiornate. - Cliente: vuole che la riparazione del proprio dispositivo sia chiara e rapida. |
| Pre-condizioni | Il cliente ha preso una decisione riguardo un preventivo. |
| Garanzia di successo | Il titolare comincia o annulla la riparazione del dispositivo. |
| Scenario principale di successo | <ol style="list-style-type: none"> 1. Il cliente accetta il preventivo. 2. Il titolare ricerca e seleziona nel sistema la scheda preventivo riparazione relativa al dispositivo [vedi UC3: Aggiorna Preventivo Riparazione]. 3. Il sistema verifica che tutti i pezzi relativi alla scheda preventivo riparazione selezionata siano disponibili, aggiorna le giacenze residue e imposta il suo stato in riparazione. 4. Il titolare comincia la riparazione. |
| Estensioni | <p>*a. In qualsiasi momento, il sistema fallisce e ha un arresto improvviso.</p> <ol style="list-style-type: none"> 1. Il titolare riavvia il software e richiede il ripristino dello stato precedente del sistema. 2. Il sistema ricostruisce lo stato precedente. <p>1a. Il cliente rifiuta il preventivo.</p> <ol style="list-style-type: none"> 1. Il titolare elimina la scheda preventivo. 2. Il titolare apre una scheda vendita e inserisce l'importo di 10€ e causale preventivo riparazione. 3. Il cliente paga; il titolare consegna il dispositivo al proprietario. <p>2a. La disponibilità non è soddisfatta per tutti i pezzi relativi alla riparazione del dispositivo.</p> <ol style="list-style-type: none"> 1. Il sistema mostra a video che i pezzi mancanti verranno ordinati [vedi UC7: Crea Ordine Distributore]. 2. All'arrivo dei pezzi lo scenario riprende dal punto 3. |
| Requisiti speciali | |

| | |
|--|--|
| Elenco delle varianti tecnologiche e dei dati | |
| Frequenza di ripetizioni | Legata al numero di preventivi definiti. |
| Varie | |

UC5: Gestisci Stato Riparazione

| | |
|--|--|
| Nome del caso d'uso | UC5: Gestisci Stato Riparazione |
| Portata | Sistema OverClock |
| Livello | Obiettivo utente |
| Attore primario | Titolare |
| Parti interessate e interessi | <ul style="list-style-type: none"> - Titolare del negozio: vuole visualizzare ed eventualmente aggiornare lo stato delle riparazioni in sospeso in maniera semplice e veloce; vuole che il sistema invii notifiche al cliente sullo stato della riparazione. - Cliente: vuole poter ricevere notifiche chiare sullo stato della riparazione. |
| Pre-condizioni | |
| Garanzia di successo | Invio di una notifica sullo stato della riparazione e/o consistenza dei dati in output. |
| Scenario principale di successo | <ol style="list-style-type: none"> 1. Il titolare ricerca e seleziona nel sistema la scheda riparazione relativa al dispositivo che ha riparato [vedi UC4: Avvio Riparazione]; il sistema mostra a video le informazioni. 2. Il titolare aggiorna lo stato della riparazione, confermando che è conclusa; il sistema invia una notifica al cliente, proprietario del dispositivo riparato. |
| Estensioni | <p>*a. In qualsiasi momento, il sistema fallisce e ha un arresto improvviso.</p> <ol style="list-style-type: none"> 1. Il titolare riavvia il software e richiede il ripristino dello stato precedente del sistema. 2. Il sistema ricostruisce lo stato precedente. <p>2a. Il titolare vuole soltanto visualizzare lo stato della riparazione, senza aggiornare lo stato.</p> |
| Requisiti speciali | |
| Elenco delle varianti tecnologiche e dei dati | |
| Frequenza di ripetizioni | Legata al numero di riparazioni effettuate. |

| | |
|--------------|--|
| Varie | |
|--------------|--|

UC6: Gestisci Acquisto Usato

| | |
|--|---|
| Nome del caso d'uso | UC6: Gestisci Acquisto Usato |
| Portata | Sistema OverClock |
| Livello | Obiettivo utente |
| Attore primario | Titolare |
| Parti interessate e interessi | <ul style="list-style-type: none"> - Titolare del negozio: vuole acquistare in maniera chiara e veloce il dispositivo del cliente; vuole che il sistema ricerchi rapidamente le informazioni relative al dispositivo. - Cliente: vuole vendere il suo dispositivo in maniera rapida e trasparente. |
| Pre-condizioni | Il cliente vuole vendere un dispositivo. |
| Garanzia di successo | Il titolare accetta o rifiuta il dispositivo. |
| Scenario principale di successo | <ol style="list-style-type: none"> 1. Il cliente mostra al titolare il dispositivo da vendere; il titolare valuta le condizioni del dispositivo. 2. Il titolare inserisce il modello del dispositivo; il sistema ricerca il dispositivo nel catalogo [vedi UC8: Gestisci Dispositivo], ritorna l'esito positivo e apre una nuova scheda d'acquisto. 3. Il titolare inserisce il grado di danneggiamento del dispositivo; il sistema ritorna i punti fedeltà [vedi Regole di business]. 4. Il cliente accetta i punti fedeltà proposti. 5. Il titolare chiede la carta fedeltà al cliente [vedi UC14: Gestisci Carta Fedeltà]. 6. Il titolare inserisce i dati della carta e conferma l'acquisto; il sistema registra l'acquisto e aggiorna il saldo della carta fedeltà, caricando i punti. |
| Estensioni | <p>*a. In qualsiasi momento, il sistema fallisce e ha un arresto improvviso.</p> <ol style="list-style-type: none"> 1. Il titolare riavvia il software e richiede il ripristino dello stato precedente del sistema. 2. Il sistema ricostruisce lo stato precedente. <p>1a. Il dispositivo è troppo danneggiato.</p> <ol style="list-style-type: none"> 1. Il titolare informa il cliente; lo scenario si conclude. <p>2a. Il dispositivo non è presente nel catalogo.</p> <ol style="list-style-type: none"> 2. Il titolare informa il cliente; lo scenario si conclude. <p>4a. Il cliente rifiuta il prezzo proposto.</p> <ol style="list-style-type: none"> 1. Il titolare chiude la scheda d'acquisto; lo scenario si conclude. |

| | |
|--|---|
| | <p>5a. Il cliente non possiede la carta fedeltà.</p> <p>1. Il titolare chiede al cliente se intende fare la carta fedeltà; il cliente accetta [vedi UC14: Gestisci Carta Fedeltà].</p> <p>1a. Il titolare chiede al cliente se intende fare la carta fedeltà; il cliente rifiuta.</p> <p>1. Il titolare chiude la scheda di acquisto.</p> |
| Requisiti speciali | |
| Elenco delle varianti tecnologiche e dei dati | |
| Frequenza di ripetizioni | Legata all'affluenza dei clienti e al numero di dispositivi usati che intendono vendere. |
| Varie | |

UC7: Crea Ordine Distributore

| | |
|--|--|
| Nome del caso d'uso | UC7: Crea Ordine Distributore |
| Portata | Sistema OverClock |
| Livello | Obiettivo utente |
| Attore primario | Titolare |
| Parti interessate e interessi | <ul style="list-style-type: none"> - Titolare del negozio: vuole assicurare una corretta rifornimento delle giacenze in magazzino; nella creazione degli ordini desidera un inserimento dei dati preciso, rapido e senza errori. - Distributore: vuole ricevere gli ordini in un formato preciso e semplice che consenta una visualizzazione chiara delle informazioni relative ai dispositivi ordinati dal titolare. |
| Pre-condizioni | Il titolare vuole rifornire il magazzino. |
| Garanzia di successo | Viene inoltrato un ordine al distributore. |
| Scenario principale di successo | <ol style="list-style-type: none"> 1. Il titolare apre una nuova scheda ordine. 2. Il titolare seleziona il tipo di articolo da ordinare (PC, telefoni, tablet e pezzi di ricambio); il sistema mostra a video l'elenco degli articoli ordinabili. 3. Il titolare seleziona modello e quantità dell'articolo da ordinare. 4. Ripetere il punto 3 per ogni articolo che il titolare intende ordinare. 5. Il titolare conferma l'ordine; Il sistema lo inoltra al distributore. |

| | |
|--|---|
| Estensioni | <p>*a. In qualsiasi momento, il sistema fallisce e ha un arresto improvviso.</p> <ol style="list-style-type: none"> 1. Il titolare riavvia il software e richiede il ripristino dello stato precedente del sistema. 2. Il sistema ricostruisce lo stato precedente. |
| Requisiti speciali | |
| Elenco delle varianti tecnologiche e dei dati | |
| Frequenza di ripetizioni | Elevata, il magazzino viene rifornito settimanalmente e vengono creati ordini per ogni prenotazione [Vedi UC11: Crea Prenotazione]. |
| Varie | |

UC8: Gestisci Dispositivo (CRUD)

| | |
|--|--|
| Nome del caso d'uso | UC8: Gestisci Dispositivo |
| Portata | Sistema OverClock |
| Livello | Obiettivo utente |
| Attore primario | Titolare |
| Parti interessate e interessi | <ul style="list-style-type: none"> - Titolare del negozio: vuole gestire (aggiungere, modificare, cancellare e visualizzare) le liste degli articoli da vendere. |
| Pre-condizioni | |
| Garanzia di successo | Consistenza dei dati in output. |
| Scenario principale di successo | <ol style="list-style-type: none"> 1. Il titolare richiede al sistema di inserire un nuovo articolo nel catalogo. 2. Il sistema chiede al titolare di inserire tutte le informazioni relative al dispositivo da aggiungere nel catalogo; il titolare le inserisce. 3. Il sistema inserisce il modello nel catalogo e aggiorna le giacenze in magazzino. 4. Ripetere i punti 2 e 3 per ogni articolo che il titolare intende inserire. |
| Estensioni | <p>*a. In qualsiasi momento, il sistema fallisce e ha un arresto improvviso.</p> <ol style="list-style-type: none"> 1. Il titolare riavvia il software e richiede il ripristino dello stato precedente del sistema. 2. Il sistema ricostruisce lo stato precedente. <p>1a. Il titolare richiede al sistema la modifica della quantità di un dispositivo.</p> <ol style="list-style-type: none"> 1. Il sistema richiede al titolare i dati aggiornati. |

| | |
|--|---|
| | 2. Il titolare aggiorna e conferma. 2a. Il codice modello esiste già nel sistema. 1. Il sistema notifica il titolare e aggiorna automaticamente le giacenze. 1b. Il titolare richiede al sistema di ricercare un dispositivo. 1. Il sistema richiede al titolare di inserire le informazioni nei campi di ricerca. 2. Il titolare inserisce le informazioni. 3. Il sistema mostra l'elenco dei dispositivi che soddisfano i criteri di ricerca. 1c. Il titolare richiede al sistema di eliminare un dispositivo dal catalogo. 1. Il sistema elimina il dispositivo. |
| Requisiti speciali | |
| Elenco delle varianti tecnologiche e dei dati | |
| Frequenza di ripetizioni | Legata al numero di dispositivi in arrivo da aggiungere al magazzino. |
| Varie | |

UC9: Gestisci Cliente (CRUD)

| | |
|--|---|
| Nome del caso d'uso | UC9: Gestisci Cliente |
| Portata | Sistema OverClock |
| Livello | Obiettivo utente |
| Attore primario | Titolare |
| Parti interessate e interessi | - Titolare del negozio: vuole gestire le informazioni sui clienti in maniera veloce e chiara. |
| Pre-condizioni | |
| Garanzia di successo | Consistenza dei dati in output. |
| Scenario principale di successo | 1. Il titolare richiede al sistema di inserire un nuovo cliente. 2. Il sistema richiede al titolare le informazioni relative al cliente. 3. Il titolare inserisce i dati del cliente. |
| Estensioni | *a. In qualsiasi momento, il sistema fallisce e ha un arresto improvviso. 1. Il titolare riavvia il software e richiede il ripristino dello stato precedente del sistema. 2. Il sistema ricostruisce lo stato precedente. |

| | |
|--|---|
| | <p>1 a. Il titolare richiede al sistema la modifica delle informazioni di un cliente.</p> <ol style="list-style-type: none"> 1. Il sistema richiede al titolare i dati aggiornati. 2. Il titolare aggiorna e conferma. <p>1 b. Il titolare richiede al sistema di ricercare un cliente.</p> <ol style="list-style-type: none"> 1. Il sistema richiede al titolare di inserire le informazioni nei campi di ricerca. 2. Il titolare inserisce le informazioni. 3. Il sistema mostra a video l'esito della ricerca. <p>1 c. Il titolare richiede al sistema di eliminare un cliente.</p> <ol style="list-style-type: none"> 1. Il sistema elimina il cliente. |
| Requisiti speciali | |
| Elenco delle varianti tecnologiche e dei dati | |
| Frequenza di ripetizioni | Legata al numero di nuovi clienti che arrivano in negozio e acquistano/riparano un dispositivo. |
| Varie | |

UC10: Gestisci Distributore (CRUD)

| | |
|--|--|
| Nome del caso d'uso | UC10: Gestisci Distributore |
| Portata | Sistema OverClock |
| Livello | Obiettivo utente |
| Attore primario | Titolare |
| Parti interessate e interessi | <ul style="list-style-type: none"> - Titolare del negozio: vuole gestire le informazioni sui distributori in maniera veloce e chiara. |
| Pre-condizioni | |
| Garanzia di successo | Concretezza dei dati in output. |
| Scenario principale di successo | <ol style="list-style-type: none"> 1. Il titolare richiede al sistema di inserire un nuovo distributore. 2. Il sistema richiede al titolare le informazioni relative al distributore. 3. Il titolare inserisce i dati del distributore. |
| Estensioni | <p>*a. In qualsiasi momento, il sistema fallisce e ha un arresto improvviso.</p> <ol style="list-style-type: none"> 1. Il titolare riavvia il software e richiede il ripristino dello stato precedente del sistema. 2. Il sistema ricostruisce lo stato precedente. <p>1 a. Il titolare richiede al sistema la modifica delle informazioni di un distributore.</p> <ol style="list-style-type: none"> 1. Il sistema richiede al titolare i dati aggiornati. |

| | |
|--|---|
| | 2. Il titolare aggiorna e conferma. 1b. Il titolare richiede al sistema di ricercare un distributore. 1. Il sistema richiede al titolare di inserire le informazioni nei campi di ricerca. 2. Il titolare inserisce le informazioni. 3. Il sistema mostra a video l'esito della ricerca. 1c. Il titolare richiede al sistema di eliminare un distributore. 1. Il sistema elimina il distributore. |
| Requisiti speciali | |
| Elenco delle varianti tecnologiche e dei dati | |
| Frequenza di ripetizioni | Bassa poiché si prevede che i distributori cambino raramente. |
| Varie | |

UC11: Crea Prenotazione

| | |
|--|---|
| Nome del caso d'uso | UC11: Crea Prenotazione |
| Portata | Sistema OverClock |
| Livello | Obiettivo utente |
| Attore primario | Titolare |
| Parti interessate e interessi | <ul style="list-style-type: none"> - Titolare del negozio: vuole poter registrare nel sistema le informazioni sulle prenotazioni in maniera chiara e veloce. - Cliente: vuole poter prenotare uno o più dispositivi in maniera semplice e veloce. |
| Pre-condizioni | È in corso una vendita e il cliente vuole prenotare uno o più dispositivi. |
| Garanzia di successo | Il sistema registra la prenotazione e aggiunge i dispositivi prenotati alla scheda di vendita. |
| Scenario principale di successo | <ol style="list-style-type: none"> 1. Il titolare cerca il dispositivo richiesto nel catalogo [vedi UC8: Gestisci Dispositivo]. Il sistema mostra l'esito positivo della ricerca al titolare, che informa il cliente. 2. Il cliente intende prenotare il dispositivo. 3. Il titolare aggiunge il dispositivo ricercato alla scheda di prenotazione. 4. Ripetere i punti 1, 2 e 3 per ogni dispositivo che il cliente intende prenotare. |

| | |
|--|--|
| Estensioni | <p>*a. In qualsiasi momento, il sistema fallisce e ha un arresto improvviso.</p> <ol style="list-style-type: none"> 1. Il titolare riavvia il software e richiede il ripristino dello stato precedente del sistema. 2. Il sistema ricostruisce lo stato precedente. <p>1a. La ricerca del dispositivo nel catalogo non ha prodotto alcun risultato.</p> <ol style="list-style-type: none"> 1. Il titolare comunica al cliente che il dispositivo non è presente nel catalogo. 2. Lo scenario riprende dal punto 4. <p>2a. Il cliente non intende prenotare il dispositivo.</p> <ol style="list-style-type: none"> 1. Se vi sono altri dispositivi, lo scenario riprende dal punto 4. Altrimenti, il titolare annulla la scheda di prenotazione e si conclude. |
| Requisiti speciali | |
| Elenco delle varianti tecnologiche e dei dati | |
| Frequenza di ripetizioni | Legata all'affluenza dei clienti e al numero di prenotazioni da loro effettuate. |
| Varie | |

UC12: Gestisci Stato Prenotazione

| | |
|--|---|
| Nome del caso d'uso | UC12: Gestisci Stato Prenotazione |
| Portata | Sistema OverClock |
| Livello | Obiettivo utente |
| Attore primario | Titolare |
| Parti interessate e interessi | <ul style="list-style-type: none"> - Titolare del negozio: vuole visualizzare ed eventualmente aggiornare lo stato delle prenotazioni in sospeso in maniera semplice e veloce; vuole che il sistema invii notifiche al cliente sullo stato della prenotazione. - Cliente: vuole poter ricevere notifiche chiare sullo stato della prenotazione. |
| Pre-condizioni | |
| Garanzia di successo | Invio di una notifica sullo stato della prenotazione e/o consistenza dei dati in output. |
| Scenario principale di successo | <ol style="list-style-type: none"> 1. Il sistema aggiorna automaticamente lo stato della prenotazione ad "in attesa di ritiro" quando viene incrementata la quantità in magazzino del dispositivo interessato. 2. Il sistema notifica il cliente interessato. |

| | |
|--|--|
| | 3. Quando il cliente ritira l'ordine, il titolare aggiorna lo stato in "ritirato". |
| Estensioni | *a. In qualsiasi momento, il sistema fallisce e ha un arresto improvviso. <ol style="list-style-type: none"> 1. Il titolare riavvia il software e richiede il ripristino dello stato precedente del sistema. 2. Il sistema ricostruisce lo stato precedente. *a. Il titolare vuole soltanto visualizzare lo stato della prenotazione, senza aggiornare lo stato. |
| Requisiti speciali | |
| Elenco delle varianti tecnologiche e dei dati | |
| Frequenza di ripetizioni | Legata al numero di prenotazioni e di ordini arrivati in magazzino. |
| Varie | |

UC13: Ricerca Vendita (CRUD)

| | |
|--|---|
| Nome del caso d'uso | UC13: Ricerca Vendita |
| Portata | Sistema OverClock |
| Livello | Obiettivo utente |
| Attore primario | Titolare |
| Parti interessate e interessi | - Titolare del negozio: vuole visualizzare le vendite registrate nel sistema in maniera chiara. |
| Pre-condizioni | |
| Garanzia di successo | Consistenza dei dati in output. |
| Scenario principale di successo | 1. Il titolare ricerca nel sistema una vendita, inserendo il relativo codice identificativo [vedi UC1: Gestisci Vendita]; il sistema mostra a video le informazioni sulla vendita. |
| Estensioni | *a. In qualsiasi momento, il sistema fallisce e ha un arresto improvviso. <ol style="list-style-type: none"> 1. Il titolare riavvia il software e richiede il ripristino dello stato precedente del sistema. 2. Il sistema ricostruisce lo stato precedente. 1a. Il titolare ricerca nel sistema le vendite relative ad un cliente, inserendo il suo codice identificativo [vedi UC9: Gestisci Cliente]; il sistema mostra a video le informazioni sulle vendite. |

| | |
|--|--|
| Requisiti speciali | |
| Elenco delle varianti tecnologiche e dei dati | |
| Frequenza di ripetizioni | Bassa, si presume che il titolare non debba ricontrollare spesso l'elenco di tutte le vendite. |
| Varie | |

UC14: Gestisci Carta Fedeltà (CRUD)

| | |
|--|--|
| Nome del caso d'uso | UC14: Gestisci Carta Fedeltà |
| Portata | Sistema OverClock |
| Livello | Obiettivo utente |
| Attore primario | Titolare |
| Parti interessate e interessi | <ul style="list-style-type: none"> - Titolare del negozio: vuole gestire (creare, modificare, eliminare e visualizzare) le carte fedeltà registrate nel sistema. - Cliente: vuole creare, modificare, eliminare la propria carta fedeltà. |
| Pre-condizioni | |
| Garanzia di successo | Consistenza dei dati in output. |
| Scenario principale di successo | <ol style="list-style-type: none"> 1. Il titolare crea una nuova carta fedeltà. 2. Il titolare chiede al cliente di comunicare i propri dati. 3. Il titolare ricerca il cliente nell'archivio interno [vedi UC9: Gestisci Cliente]; il software restituisce il risultato richiesto (cliente già presente nell'archivio) e lo associa alla carta fedeltà. 4. Il titolare conferma; il software registra la carta fedeltà nel sistema. |
| Estensioni | <p>*a. In qualsiasi momento, il sistema fallisce e ha un arresto improvviso.</p> <ol style="list-style-type: none"> 1. Il titolare riavvia il software e richiede il ripristino dello stato precedente del sistema. 2. Il sistema ricostruisce lo stato precedente. <p>1a. Il titolare richiede al sistema la modifica di una carta fedeltà selezionata.</p> <ol style="list-style-type: none"> 1. Il sistema richiede al titolare i dati aggiornati. 2. Il titolare aggiorna e conferma. <p>1b. Il titolare richiede al sistema di ricercare una carta fedeltà.</p> <ol style="list-style-type: none"> 1. Il sistema richiede al titolare di inserire le informazioni nei campi di ricerca. 2. Il titolare inserisce le informazioni. |

| | |
|--|--|
| | <p>3. Il sistema mostra la carta fedeltà che soddisfa i criteri di ricerca.</p> <p>1 c. Il titolare richiede al sistema di eliminare una carta fedeltà selezionata.</p> <p>1. Il sistema chiede al titolare di confermare l'eliminazione.</p> <p>3. Il titolare conferma; il sistema elimina la carta fedeltà.</p> <p>3a. La ricerca del cliente nell'archivio interno non ha prodotto alcun risultato.</p> <p>1. Il sistema chiede al titolare di inserire i dati del cliente prima di proseguire.</p> <p>2. Il titolare inserisce i dati del cliente [vedi UC9: Gestisci Cliente].</p> <p>3. Il sistema associa il cliente appena inserito alla carta fedeltà.</p> |
| Requisiti speciali | |
| Elenco delle varianti tecnologiche e dei dati | |
| Frequenza di ripetizioni | Legata al numero di clienti che intendono creare, modificare, eliminare la propria carta fedeltà. |
| Varie | |

UC15: Applica Sconto Carta

| | |
|--|--|
| Nome del caso d'uso | UC15: Applica Sconto Carta |
| Portata | Sistema OverClock |
| Livello | Obiettivo utente |
| Attore primario | Titolare |
| Parti interessate e interessi | <ul style="list-style-type: none"> - Titolare del negozio: vuole gestire visualizzare in modo chiaro gli sconti, applicabili alla vendita, usufruendo dei punti fedeltà. - Cliente: vuole vedere il saldo sulla propria carta fedeltà in maniera chiara e decidere se usare i punti per usufruire di uno sconto. |
| Pre-condizioni | È in corso una vendita. |
| Garanzia di successo | |
| Scenario principale di successo | <p>1. Il cliente mostra la carta fedeltà; il sistema ricerca la carta fedeltà e mostra a video il saldo disponibile [vedi UC14: Gestisci Carta Fedeltà].</p> <p>2. Il cliente desidera verificare le informazioni sullo sconto; il sistema calcola i punti da scalare dal saldo, applica</p> |

| | |
|--|--|
| | <p>1€ di sconto per ogni 10 punti scalati dal saldo [vedi Regole di business], ricalcola il totale e lo mostra a video.</p> <p>3. Il cliente comunica al titolare che vuole confermare lo sconto, quest'ultimo procede alla conferma.</p> |
| Estensioni | <p>*a. In qualsiasi momento, il sistema fallisce e ha un arresto improvviso.</p> <ol style="list-style-type: none"> 1. Il titolare riavvia il software e richiede il ripristino dello stato precedente del sistema. 2. Il sistema ricostruisce lo stato precedente. <p>1a. Il cliente non possiede la carta fedeltà.</p> <ol style="list-style-type: none"> 1. Il titolare chiede al cliente se intende fare la carta fedeltà; il cliente accetta [vedi UC14: Gestisci Carta Fedeltà]. 1a. Il titolare chiede al cliente se intende fare la carta fedeltà; il cliente rifiuta. 1. Lo scenario si conclude. <p>3a. Il cliente comunica al titolare che non vuole confermare lo sconto.</p> <ol style="list-style-type: none"> 1. Lo scenario si conclude. |
| Requisiti speciali | |
| Elenco delle varianti tecnologiche e dei dati | |
| Frequenza di ripetizioni | Legata all'affluenza dei clienti e agli acquisti che intendono effettuare usufruendo di una carta fedeltà. |
| Varie | |

UC16: Gestisci Promozione

| | |
|--|---|
| Nome del caso d'uso | UC16: Gestisci Promozione |
| Portata | Sistema OverClock |
| Livello | Obiettivo utente |
| Attore primario | Titolare |
| Parti interessate e interessi | - Titolare del negozio: vuole gestire (creare, modificare, eliminare e visualizzare) una promozione. |
| Pre-condizioni | |
| Garanzia di successo | Consistenza dei dati in output. |
| Scenario principale di successo | <ol style="list-style-type: none"> 1. Il titolare richiede al sistema di inserire una nuova promozione. 2. Il sistema mostra a video gli articoli attualmente in catalogo; il titolare seleziona un articolo. |

| | |
|--|--|
| | <ol style="list-style-type: none"> 3. Il sistema richiede al titolare di inserire lo sconto percentuale da applicare. 4. Il titolare inserisce lo sconto e conferma; il sistema crea la promozione. |
| Estensioni | <p>*a. In qualsiasi momento, il sistema fallisce e ha un arresto improvviso.</p> <ol style="list-style-type: none"> 1. Il titolare riavvia il software e richiede il ripristino dello stato precedente del sistema. 2. Il sistema ricostruisce lo stato precedente. <p>1a. Il titolare richiede al sistema di ricercare una promozione.</p> <ol style="list-style-type: none"> 1. Il sistema richiede al titolare di inserire le informazioni nei campi di ricerca. 2. Il titolare inserisce le informazioni a sua disposizione. 3. Il sistema mostra l'elenco delle promozioni che soddisfano i criteri di ricerca. <p>1b. Il titolare richiede al sistema di eliminare una promozione.</p> <ol style="list-style-type: none"> 1. Il sistema elimina la promozione. |
| Requisiti speciali | |
| Elenco delle varianti tecnologiche e dei dati | |
| Frequenza di ripetizioni | Legata al numero di promozioni che il titolare crea/ha creato. |
| Varie | |

1.5 DOCUMENTO DI VISIONE

Il Documento di Visione è stato redatto congiuntamente alla presente documentazione, lo si può trovare in allegato [vedi Documento di Visione].

1.6 REGOLE DI BUSINESS

Per l'utilizzo corretto dell'applicazione occorre seguire le seguenti regole di business:

| ID | Regola | Modificabilità | Sorgente |
|----|--|--|-------------------------------|
| R1 | L'importo totale della spesa viene convertito, arrotondando per difetto, in punti fedeltà sulla carta del cliente (ad esempio, con una spesa di 99,90€ vengono ricaricati 99 punti sulla carta fedeltà). | Bassa, decisa al momento del rilascio della carta. | Politica interna del negozio. |

| | | | |
|----|--|--|-------------------------------|
| R2 | Usufruendo di 10 punti fedeltà, il cliente ha diritto ad 1€ di sconto sull'importo totale della vendita. | Bassa, decisa al momento del rilascio della carta. | Politica interna del negozio. |
| R3 | Il cliente che vende un dispositivo usato viene pagato in punti fedeltà, i quali vengono ricaricati sulla propria carta. Vengono ricaricati tanti punti quanto costa il dispositivo nel catalogo moltiplicati per 0.8 se il dispositivo è in ottime condizioni, per 0.6 se è in buone condizioni e per 0.4 se è in condizioni accettabili. | Bassa, decisa al momento del rilascio della carta. | Politica interna del negozio. |
| R4 | Al momento dell'aggiunta di un articolo al carrello, se l'importo totale supera 200€ verranno applicati 20€ di sconto; se supera 500€ ne verranno applicati 50; se supera 1000€ ne verranno applicati 100. | Bassa, politica del negozio. | Politica interna del negozio. |
| R5 | Il titolare può creare ulteriori promozioni e sconti percentuali da applicare ad articoli a scelta in vendita nel negozio | Alta, è il titolare a decidere quando e come creare tali promozioni. | Politica interna del negozio. |

1.7 SPECIFICHE SUPPLEMENTARI

1.7.1 Usabilità

- L'interfaccia grafica deve essere intuitiva e semplice anche per un utente non esperto.
- L'interazione con il sistema non deve presentare un elevato grado di complessità.

1.7.2 Affidabilità

- Il software sviluppato deve essere affidabile e deve poter salvare e recuperare in locale i dati salvati dall'utente.

1.7.3 Vincoli hardware e software

- Il sistema operativo non deve avere requisiti particolari, purché sia presente la Java Virtual Machine e l'accesso al database SQL.
- Il sistema che esegue il software deve essere dotato di una connessione ad Internet per inviare e-mail.

1.7.4 Vincoli di sviluppo del software

- Il software è stato scritto utilizzando Java.

1.7.5 Aspetti legali

- Le tecnologie utilizzate per progettare e realizzare il software sono di tipo open source o freeware.

1.8 GLOSSARIO

Di seguito vengono riportati tutti i termini significativi del progetto, con le relative definizioni.

- **Acquisto/Acquisto usato:** si riferisce all'intero processo di acquisto di articoli usati dei clienti.
- **Carta fedeltà:** ha un codice univoco e dei punti fedeltà residui. È legata ad un solo cliente e a sconti nel momento della vendita.
- **Catalogo:** è una lista di tutti i dispositivi vendibili e acquistabili nel/dal negozio.
- **Cliente:** è una parte interessata nella vendita, acquisto e riparazione. Ha una serie di attributi che servono a identificarlo come il nome, cognome, e-mail e telefono.
- **Codice modello:** è identificativo del modello del dispositivo, dispositivi uguali hanno il medesimo codice modello.
- **Disponibilità in magazzino/Giacenza:** indica se vi sono articoli di un determinato dispositivo in magazzino e la loro quantità.
- **Disponibilità nel catalogo:** indica se un determinato dispositivo si può acquistare/vendere nel negozio.
- **Dispositivo:** ha una serie di attributi, tra i quali i più significativi sono il nome, il codice modello e la quantità, che servono a verificarne la disponibilità nel catalogo e la quantità in magazzino. Il nome e il codice sono univoci.
- **Distributore:** colui a cui sono destinati gli ordini per rifornire il magazzino. Ha una serie di attributi identificativi come l'e-mail e il tipo di prodotti che vende.
- **Prenotazione:** riferito ad un articolo che il cliente intende acquistare ma non è disponibile in magazzino. Ha degli attributi che lo legano ad un cliente e al codice modello del dispositivo e che ne indicano lo stato ("ordinato", "in attesa di ritiro" o "ritirato").
- **Preventivo:** ha degli attributi che ne indicano lo stato ("in corso" o "concluso") e il prezzo e attributi che lo legano al cliente proprietario e al dispositivo da riparare.
- **Promozione/Sconto:** creati e/o gestiti dal titolare, si applicano al momento della vendita. Possono essere di diversi tipi (vedi Regole di Business).
- **Punti fedeltà:** sono i punti residui sulla carta fedeltà, possono essere ricaricati vendendo articoli usati o sfruttati per usufruire di sconti (vedi Regole di Business).
- **Riparazione:** si riferisce all'intero processo di riparazione di un dispositivo danneggiato di un cliente.
- **Sconto:** si riferisce allo sconto fisso applicato al momento della vendita, al raggiungimento di una soglia di spesa (vedi Regole di Business).
- **Titolare:** colui che utilizza il software e gestisce ogni operazione.
- **Vendita:** si riferisce all'intero processo che va dalle richieste dei clienti alla vendita di articoli del negozio.

2 ANALISI ORIENTATA AGLI OGGETTI

2.1 INTRODUZIONE

Seguendo l'approccio iterativo evolutivo, la realizzazione dell'applicazione è stata articolata su 5 iterazioni. In questo modo è stato possibile implementare in maniera iterativa il nucleo dell'architettura del software, sono state risolte le problematiche relative ai rischi maggiori ed è stata affrontata un'analisi dei requisiti graduale in modo da limitare al minimo il danno causato da eventuali errori di progettazione e implementazione. Per ciascuna iterazione in particolare ci si è occupati di gestire le seguenti problematiche:

❖ Iterazione 1 - Vendite

- Implementazione scenario principale UC1: Gestisci Vendita tralasciando, per semplicità, l'implementazione di promozioni e sconti e la prenotazione.

❖ Iterazione 2 – Ordini e Prenotazioni

- Implementazione del caso d'uso UC7: Crea Ordine Distributore, esaminato con i suoi scenari alternativi. Definisce la possibilità da parte del titolare di rifornire il magazzino richiedendo articoli.
- Implementazione del caso d'uso UC11: Crea Prenotazione, che permette ai clienti di poter prenotare degli articoli attualmente non disponibili in magazzino.
- Implementazione del caso d'uso UC12: Gestisci Stato Prenotazione, descrive come il titolare o il sistema gestiscano le prenotazioni in corso.

❖ Iterazione 3 - Riparazioni

- Implementazione del caso d'uso UC2: Crea Preventivo Riparazione. Definisce le modalità e i vincoli di creazione di preventivi inerenti alla riparazione di un dispositivo di un cliente.
- Implementazione del caso d'uso UC3: Aggiorna Preventivo Riparazione, che analizza le modalità di notifiche verso il cliente e l'aggiunta di pezzi di ricambio ai preventivi in corso.
- Implementazione del caso d'uso UC4: Avvio Riparazione, scenario in cui oltre ad avvisare nuovamente il cliente dello stato della riparazione, vengono effettuati ordini ai distributori se vi sono pezzi mancanti.
- Implementazione del caso d'uso UC5: Gestisci Stato Riparazione, scenario in cui vengono descritte le modalità di modifica dei preventivi, dalla gestione dello stato all'eliminazione.

❖ Iterazione 4 – Acquisto Usato

- Implementazione del caso d'uso UC6: Gestisci Acquisto Usato. Descrive la possibilità da parte di un cliente di vendere articoli usati, le modalità di assegnazione dei punti fedeltà e l'inserimento del dispositivo in catalogo.

❖ Iterazione 5 – Promozioni e Sconti

- Implementazione degli sconti legati all'importo totale durante le vendite.
- Implementazione del caso d'uso UC16: Gestisci Promozione, che permette al titolare di creare e associare promozioni ad un particolare articolo.

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 ad oggetti. Per fornire tale descrizione sono stati utilizzati

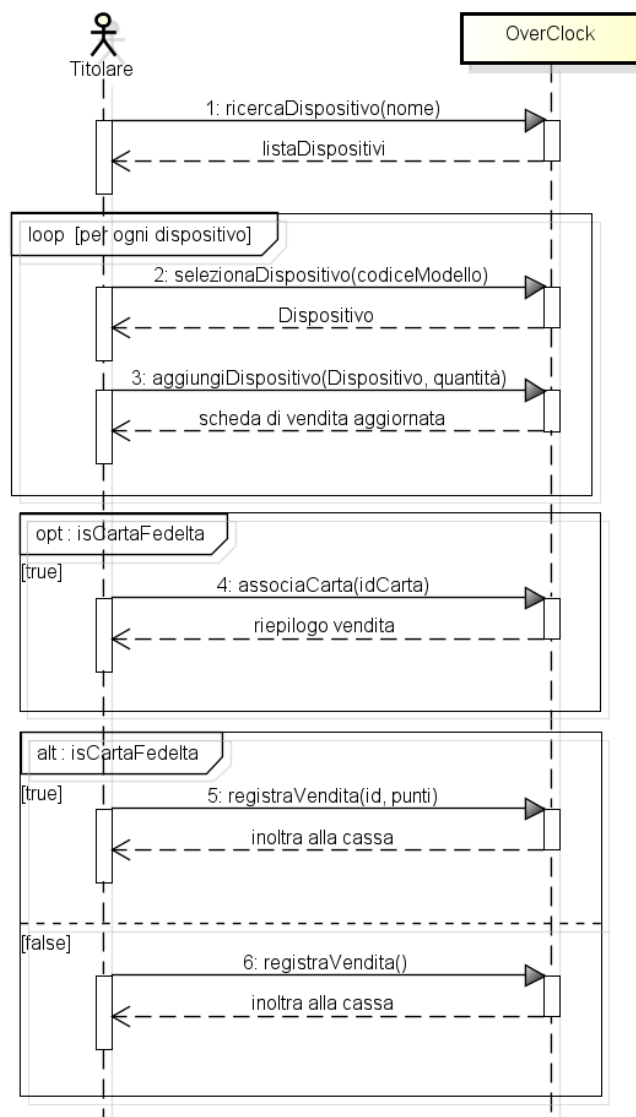
- **SchedaAcquisto:** rappresenta la scheda relativa alla riparazione, associata ad un dispositivo associata e ad un cliente.
- **Promozione:** contiene le informazioni relative alle promozioni attualmente applicate agli articoli in catalogo.
- **Sconto:** rappresenta gli sconti applicati al momento della vendita.

2.3 SSD E CONTRATTI

Procedendo con l'analisi Orienta agli Oggetti, il passo successivo è la creazione dei Diagramma di Sequenza di Sistema (SSD) al fine di illustrare il corso degli eventi di input e di output per i vari casi d'uso esaminati in ciascuna iterazione. Inoltre, le principali operazioni di sistema individuate negli SSD verranno descritte attraverso i Contratti.

❖ Iterazione 1 - Vendite

Scenario principale del caso d'uso UC1: Gestisci Vendita



Contratto CO1: ricercaDispositivo(nome)

Operazione: ricercaDispositivo(nome: String)

Riferimenti: caso d'uso: Gestisci Vendita

Pre-condizioni: è in corso una vendita schedaDiVendita.

Post-condizioni:

- È stata effettuata una ricerca in catalogo tramite id.
- Catalogo ha creato una lista dispositiviRicercati di Dispositivi.
- I risultati della ricerca sono stati aggiunti a dispositiviRicercati.
- È stata ritornata dispositiviRicercati.

Contratto CO2: aggiungiDispositivo(dispositivo, quantità)

Operazione: aggiungiDispositivo(dispositivo: Dispositivo, quantità: int)

Riferimenti: caso d'uso: Gestisci Vendita

Pre-condizioni: è in corso una vendita schedaDiVendita ed è stato selezionato un dispositivo tramite il metodo getDispositivo(codice) su dispositiviRicercati.

Post-condizioni:

- È stata creata un'istanza rv di RigaDiVendita.
- Gli attributi di rv sono stati aggiornati con dispositivo e quantità.
- L'attributo prezzo di rv è stato aggiornato moltiplicando quantità con l'attributo prezzo di dispositivo.
- rv è stata associata a schedaDiVendita.
- È stato verificato il nuovo importo totale della schedaDiVendita tramite il metodo getTotale().
- In base all'importo totale è stato calcolato lo sconto da applicare tramite una catena di responsabilità.
- È stato aggiornato l'importo totale di schedaDiVendita tramite il metodo setTotale() sottraendo lo sconto dal totale.

Contratto CO3: associaCarta(idCarta)

Operazione: associaCarta(idCarta: String)

Riferimenti: caso d'uso: Gestisci Vendita

Pre-condizioni: è stata creata una scheda di vendita schedaDiVendita.

Post-condizioni:

- È stata ritornata cartaRicercata dopo una ricerca tramite id nella lista CarteFedelta.
- schedaDiVendita è stata associata a cartaRicercata.

Contratto CO4: registraVendita(id, punti)

Operazione: confermaVenditaCarta(idCarta: String, punti: int)

Riferimenti: caso d'uso: Gestisci Vendita

Pre-condizioni: è stata creata una scheda di vendita schedaDiVendita, carta è stata associata a schedaDiVendita.

Post-condizioni:

- L'attributo quantità di dispositivo è stato aggiornato sottraendo la quantità presente in schedaDiVendita.
- L'attributo punti di carta è stato aggiornato convertendo ad intero l'attributo importoTotale di schedaDiVendita.
- schedaDiVendita è stata aggiunta all'archivio delle vendite.
- La vendita è stata registrata nel database delle vendite.

Contratto C04: registraVendita()

Operazione: confermaVendita()

Riferimenti: caso d'uso: Gestisci Vendita

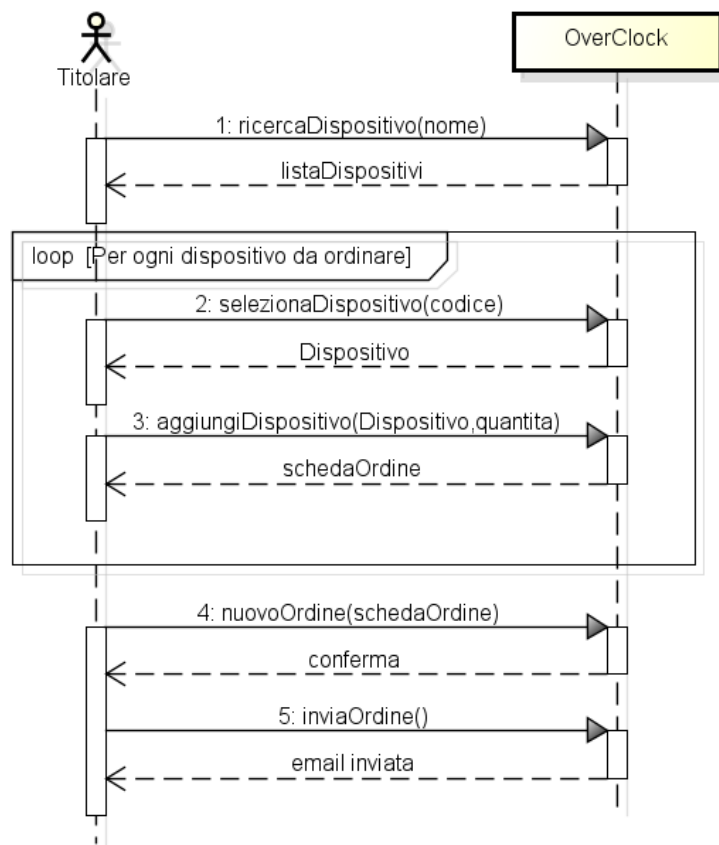
Pre-condizioni: è stata creata una scheda di vendita schedaDiVendita.

Post-condizioni:

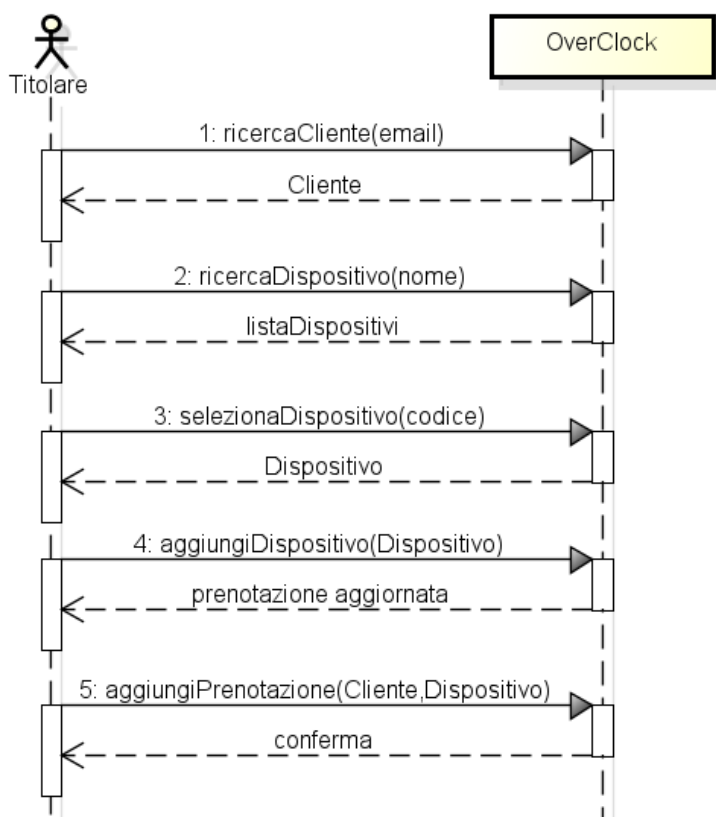
- L'attributo quantità di dispositivo è stato aggiornato sottraendo la quantità presente in schedaDiVendita.
- schedaDiVendita è stata aggiunta all'archivio delle vendite.
- La vendita è stata registrata nel database delle vendite.

❖ Iterazione 2 – Ordini e Prenotazioni

Scenario principale del caso d'uso UC7: Crea Ordine Distributore



Scenario principale del caso d'uso UC11: Crea Prenotazione



Riguardo i casi d'uso UC7 e UC11, omettendo l'analisi di ricerca cliente e ricerca dispositivo già trattate in precedenza, sono state analizzate le seguenti operazioni:

Contratto CO1: **aggiungiDispositivo(dispositivo, quantità)**

Operazione: `aggiungiDispositivo(dispositivo: Dispositivo, quantità: int)`

Riferimenti: caso d'uso: Crea Ordine Distributore

Pre-condizioni: è in corso un ordine `schedaOrdine`.

Post-condizioni:

- È stata effettuata una ricerca tramite nome in `listaDispositivi` di `Dispositivo`.
- È stato selezionato un dispositivo della lista tramite codice.
- È stata selezionata una quantità intera.
- È stata creata un'istanza `r` di `RigaOrdine`.
- Gli attributi di `r` sono stati aggiornati con il dispositivo ricercato e la quantità selezionata.
- L'attributo `subTotale` di `r` è stato calcolato moltiplicando il costo del dispositivo per la quantità.
- `r` è stata legata a riga d'ordine tramite l'associazione `contiene`.

Contratto CO2: **nuovoOrdine(schedaOrdine)**

Operazione: `nuovoOrdine(schedaOrdine: SchedaOrdine)`

Riferimenti: caso d'uso: Crea Ordine Distributore

Pre-condizioni: è in corso un ordine `schedaOrdine`.

Post-condizioni:

- schedaOrdine è stata aggiunta al registro degli ordini.
- Il database degli ordini è stato aggiornato inserendo la schedaOrdine.

Contratto CO3: inviaOrdine()

Operazione: inviaOrdine()

Riferimenti: caso d'uso: Crea Ordine Distributore

Pre-condizioni: l'ordine schedaOrdine è stato aggiornato e salvato correttamente.

Post-condizioni:

- È stata creato un vettore di stringhe listaEmail.
- listaEmail è stato riempito con le e-mail dei distributori interessati nell'ordine, tramite il metodo getEmail() di Distributore legato a Dispositivo.
- È stata creata una stringa listaArticoli.
- Per ogni elemento di listaEmail, listaArticoli è stata aggiornata con la lista dei dispositivi della riga d'ordine inerente a quel distributore, con la relativa quantità.
- Per ogni elemento di listaEmail è stata inviata una e-mail, contenente l'oggetto dell'ordine e il messaggio con la listaArticoli.

Contratto CO4: ricercaCliente(email)

Operazione: ricercaCliente(email: String)

Riferimenti: caso d'uso: Crea Prenotazione

Pre-condizioni: è in corso una prenotazione.

Post-condizioni:

- È stata effettuata una ricerca tramite e-mail in listaClienti di Cliente.
- È stata ritornata l'istanza cliente del cliente ricercato.

Contratto CO5: aggiungiPrenotazione(cliente,dispositivo)

Operazione: aggiungiPrenotazione(cliente: Cliente, dispositivo: Dispositivo)

Riferimenti: caso d'uso: Crea Prenotazione

Pre-condizioni: è in corso una prenotazione e sono stati selezionati un cliente e un dispositivo.

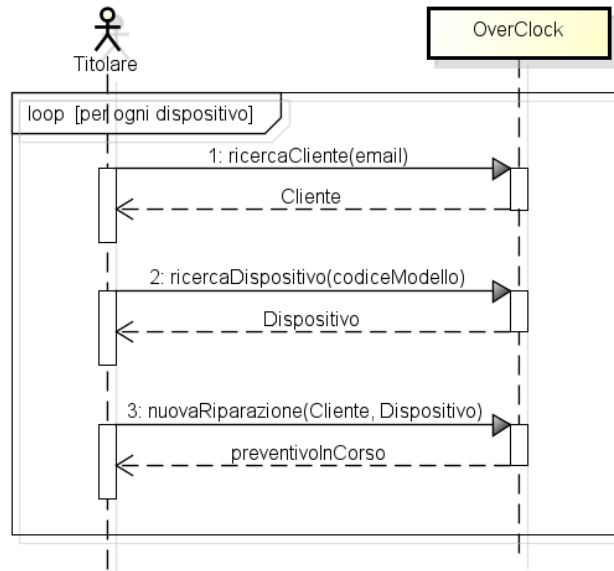
Post-condizioni:

- È stata creata un'istanza p di prenotazione.
- Gli attributi di prenotazione sono stati aggiornati con cliente, dispositivo e la data corrente.
- p è stata aggiunta al registro delle riparazioni.
- Il database delle riparazioni è stato aggiornato inserendo p.

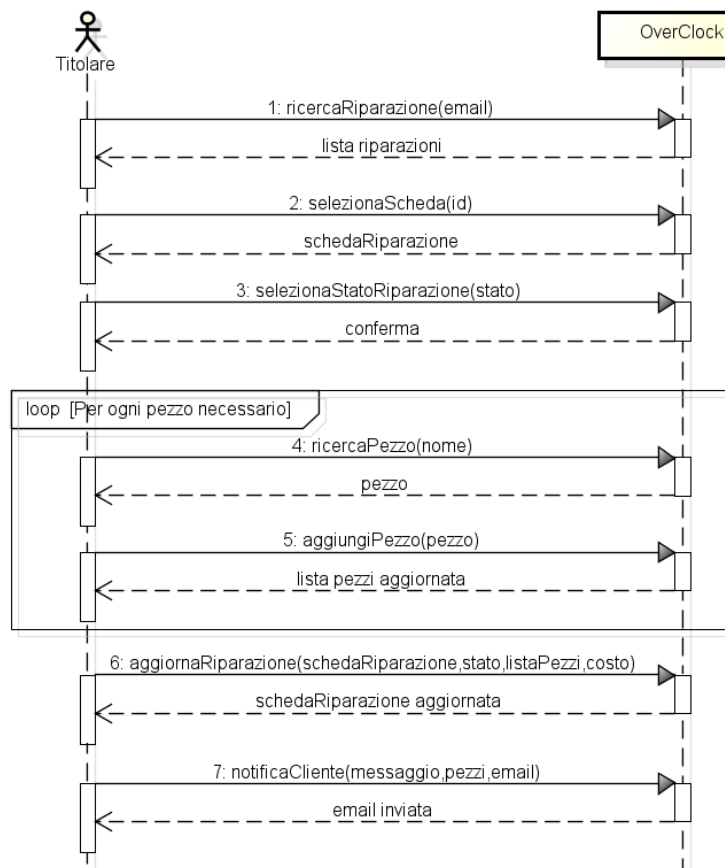
❖ Iterazione 3– Riparazioni

I Diagrammi di Sequenza di Sistema (SSD) sulle riparazioni riportati di seguito sono relativi ai casi d'uso UC2, UC3, UC4 e lo scenario alternativo di UC4, quando i pezzi non sono disponibili in magazzino. UC5 non è stato riportato poiché lo scenario principale è analogo ad UC4.

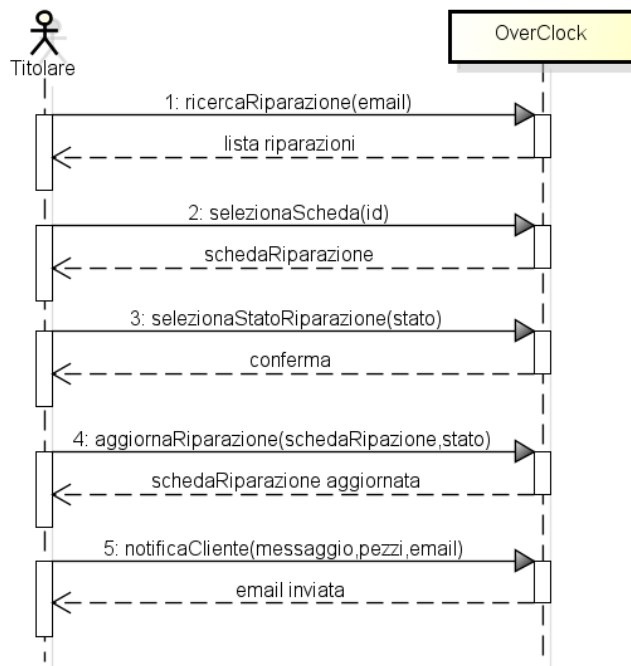
Scenario principale del caso d'uso UC2: Crea Preventivo Riparazione



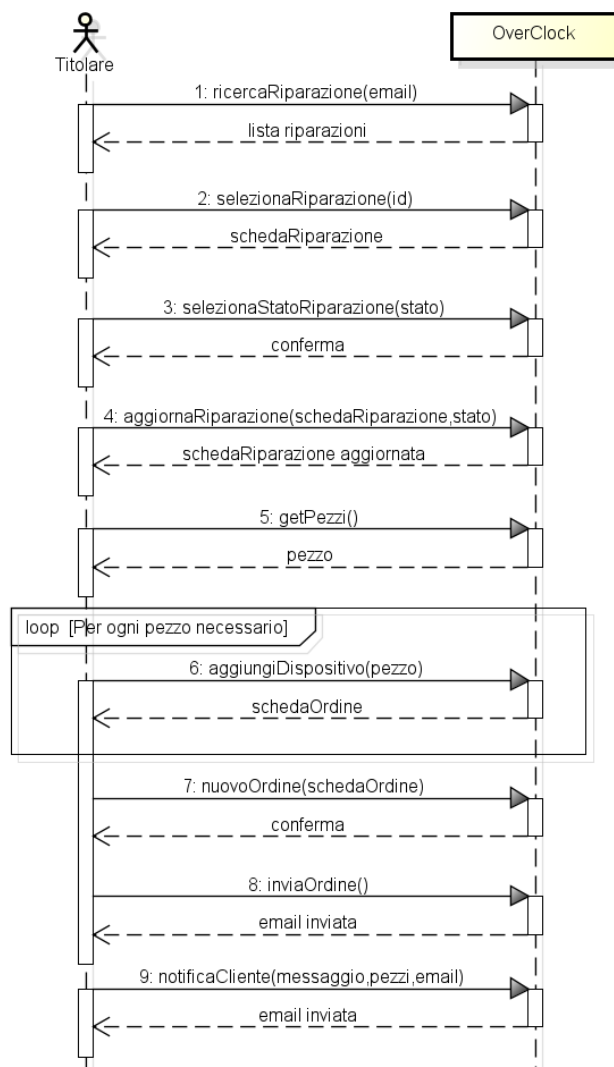
Scenario principale del caso d'uso UC3: Aggiorna Preventivo Riparazione



Scenario principale del caso d'uso UC4: Avvio Riparazione



Scenario principale del caso d'uso UC4: Avvio Riparazione Alternativo



Molte delle operazioni riguardanti i casi d'uso sulle riparazioni sono simili tra loro o sono state già trattate in precedenza, come la verifica della disponibilità in magazzino o l'invio di ordini, pertanto non verranno ripetute. Verranno riportate le principali.

Contratto CO1: nuovaRiparazione(cliente, dispositivo)

Operazione: nuovaRiparazione(cliente: Cliente, dispositivo: Dispositivo)

Riferimenti: caso d'uso: Crea Preventivo Riparazione

Pre-condizioni: viene richiesta una riparazione e cliente e dispositivo sono stati ritornati correttamente.

Post-condizioni:

- È stata creata un'istanza r di SchedaRiparazione.
- Gli attributi di r sono stati aggiornati con cliente, dispositivo e data.
- L'enum di r che indica lo stato della riparazione è stato impostato su "preventivo in corso".
- r è stata aggiunta all'archivio delle riparazioni in corso.

Contratto CO2: ricercaRiparazione(email)

Operazione: ricercaRiparazione(email: String)

Riferimenti: caso d'uso: Aggiorna Preventivo Riparazione

Pre-condizioni: vi è una riparazione salvata nel database delle riparazioni.

Post-condizioni:

- È stata inserita una stringa contenente l'e-mail.
- Tramite il metodo getRiparazione() sono state ricercate le riparazioni associate all'e-mail.

Contratto CO3: selezionaStatoRiparazione(stato)

Operazione: selezionaStatoRiparazione(stato: String)

Riferimenti: caso d'uso: Aggiorna Preventivo Riparazione

Pre-condizioni: è stata selezionata una schedaRiparazione da aggiornare tramite il metodo getSchedaRiparazione(id) sulla lista delle riparazioni ricercate.

Post-condizioni:

- È stato selezionato un stato tramite il metodo getSelectedIndex().
- Lo stato selezionato è stato assegnato ad una variabile stato di tipo Enum.

Contratto CO4: aggiornaRiparazione(schedaRiparazione,stato,listaPezzi,costo)

Operazione: aggiornaRiparazione(schedaRiparazione: SchedaRiparazione, stato: enum, listaPezzi: List<Dispositivo>, costo:float)

Riferimenti: caso d'uso: Aggiorna Preventivo Riparazione

Pre-condizioni: è stata selezionata una schedaRiparazione da aggiornare, uno stato ed i pezzi sono stati ricercati in magazzino e aggiunti ad una listaPezzi di Dispositivo.

Post-condizioni:

- Gli attributi di schedaRiparazione sono stati aggiornati con stato, listaPezzi e costo tramite il metodo aggiornamentoRiparazione() di SchedaRiparazione.
- La schedaRiparazione è stata aggiornata nel database.

Contratto CO5: notificaCliente(messaggio,pezzi,email)

Operazione: notificaCliente(messaggio: String, pezzi: String, email: String)

Riferimenti: caso d'uso: Aggiorna Preventivo Riparazione

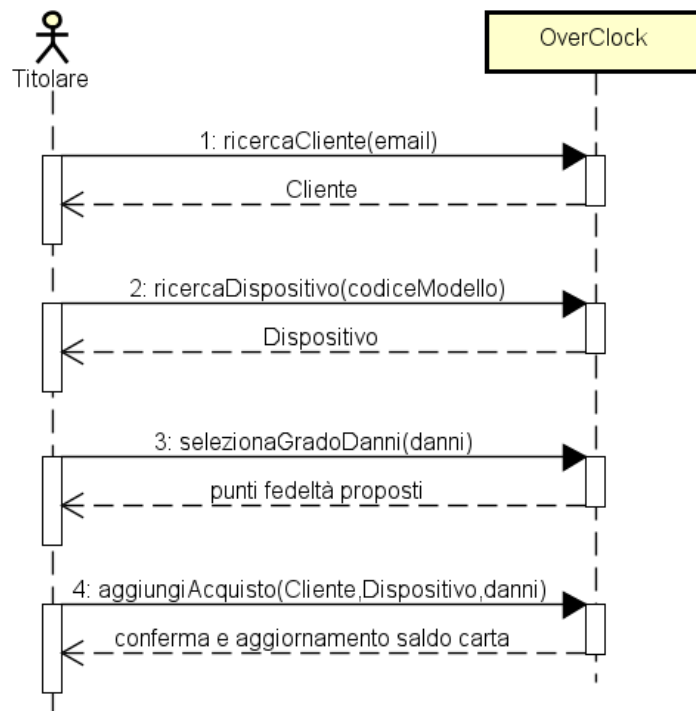
Pre-condizioni: è stata aggiornata una schedaRiparazione e sono stati salvati i pezzi necessari in un vettore.

Post-condizioni:

- È stata creata una stringa pezzi.
- Per ogni elemento del vettore dei pezzi è stata aggiunta una riga a pezzi, contenente il nome del pezzo e la quantità, ottenuti tramite getNome() e getQuantita() di Dispositivo.
- È stata inviata una mail contenente un messaggio e la stringa pezzi all'e-mail del cliente associato alla schedaRiparazione.

❖ Iterazione 4 – Acquisto Usato

Scenario principale del caso d'uso UC6: Gestisci Acquisto Usato



Le operazioni di ricerca sono state già trattate in precedenza, mentre la selezione del grado di danneggiamento è simile alla selezione dello stato di riparazione. Pertanto, verrà riportata solamente l'operazione di aggiunta dell'acquisto.

Contratto CO1: aggiungiAcquisto(cliente, dispositivo, danni)

Operazione: aggiungiAcquisto(cliente: Cliente, dispositivo: Dispositivo, danni: Enum)

Riferimenti: caso d'uso: Gestisci Acquisto Usato

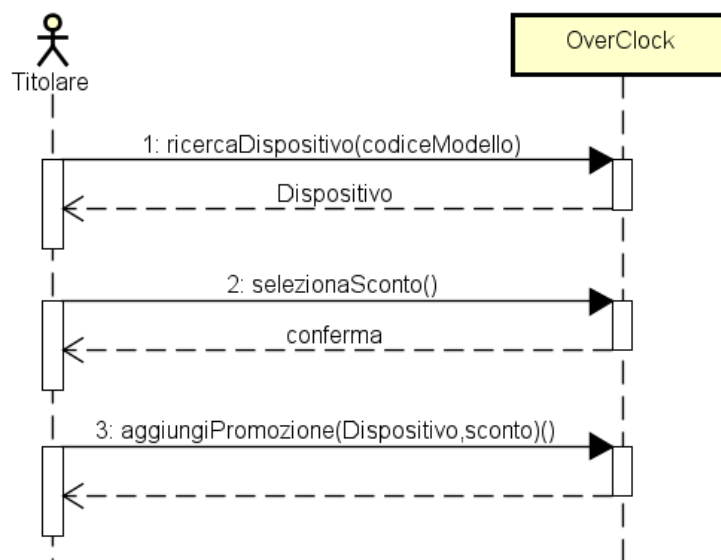
Pre-condizioni: un cliente registrato vuole vendere un dispositivo usato, vendibile in negozio e sono stati già valutati danni e punti.

Post-condizioni:

- È stata creata un'istanza s di SchedaAcquisto.
- Gli attributi di s sono stati aggiornati con cliente, dispositivo, danni e data.
- s è stata aggiunta all'archivio degli acquisti.
- Il saldo sulla carta fedeltà del cliente è stato aggiornato tramite il metodo aggiornaSaldo().
- Il dispositivo è stato aggiunto al catalogo di vendita.
- È stato aggiornato il database dei dispositivi e degli acquisti.

❖ Iterazione 5 – Promozioni e Sconti

Scenario principale del caso d'uso UC16: Gestisci Promozione



Escludendo le operazioni già evidenziate in precedenza, di seguito è riportata l'operazione analizzata per il caso d'uso UC16:

Contratto CO1: aggiungiPromozione(dispositivo, sconto)

Operazione: aggiungiPromozione(dispositivo: Dispositivo, sconto:int)

Riferimenti: caso d'uso: Gestisci Promozione

Pre-condizioni: il titolare vuole inserire la promozione dopo aver ricercato il dispositivo ed aver inserito lo sconto desiderati.

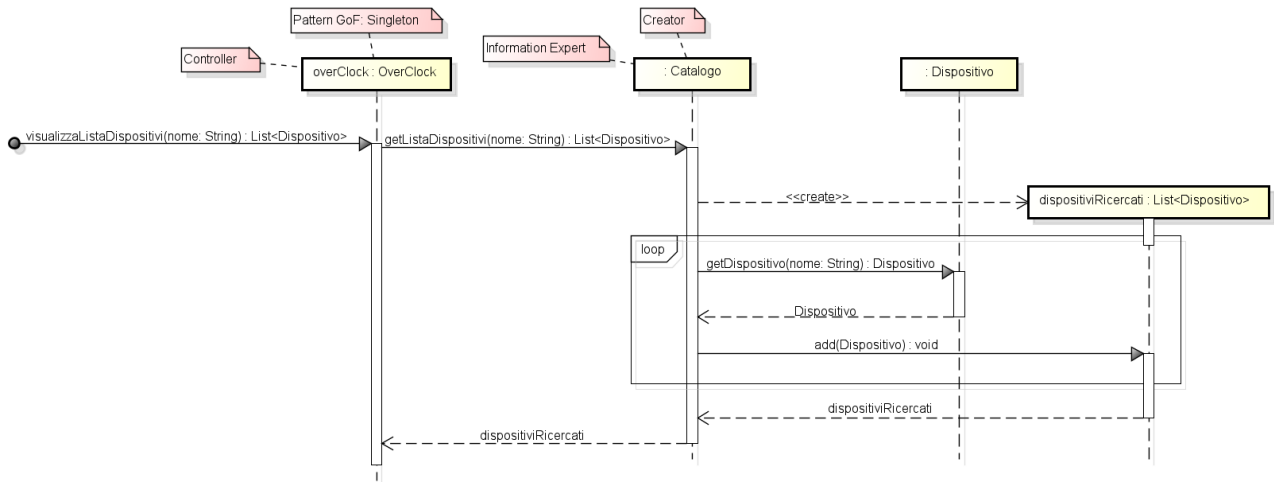
Post-condizioni:

- È stata creata un'istanza p di Promozione.
- Gli attributi di p sono stati aggiornati con dispositivo, sconto e data.
- p è stata aggiunta all'archivio delle promozioni.
- È stato aggiornato il database delle promozioni.

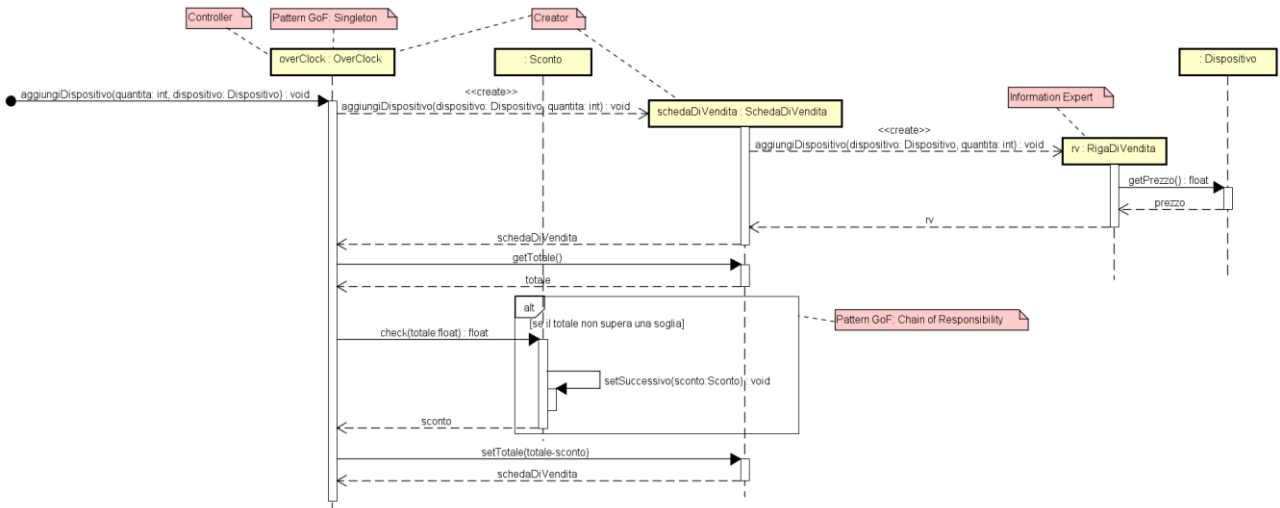
3.2 DIAGRAMMI DI SEQUENZA

Vengono ora presentati i più importanti diagrammi di sequenza, per maggiori dettagli e/o per visionarli tutti si vedano i documenti sorgente creato con il software Astah Professional.

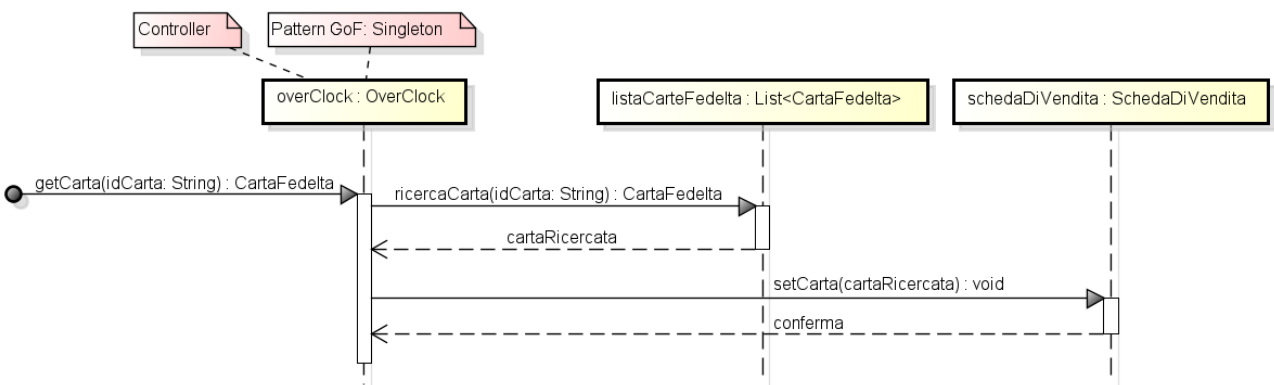
Ricerca di un dispositivo



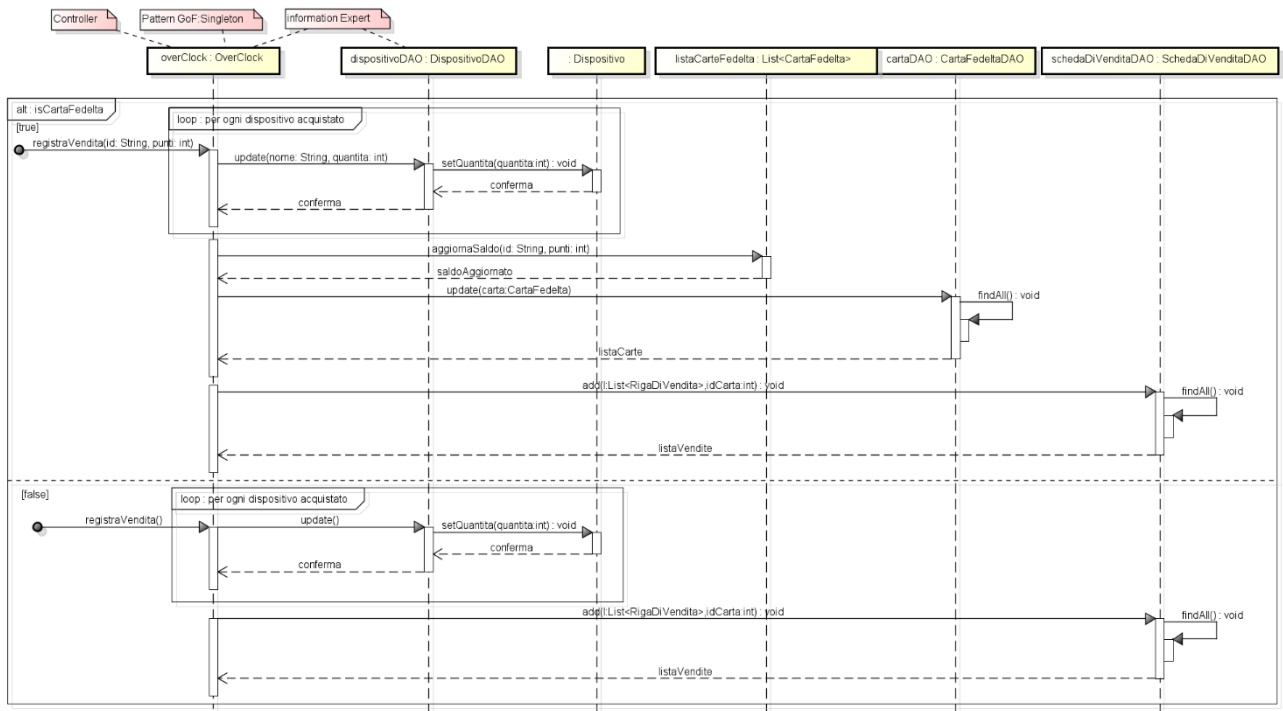
Aggiunta di un dispositivo al carrello



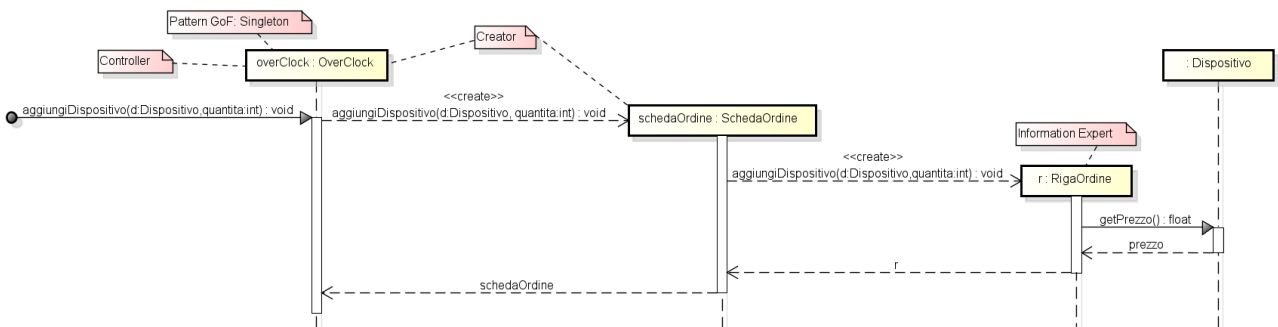
Associazione carta fedeltà alla vendita



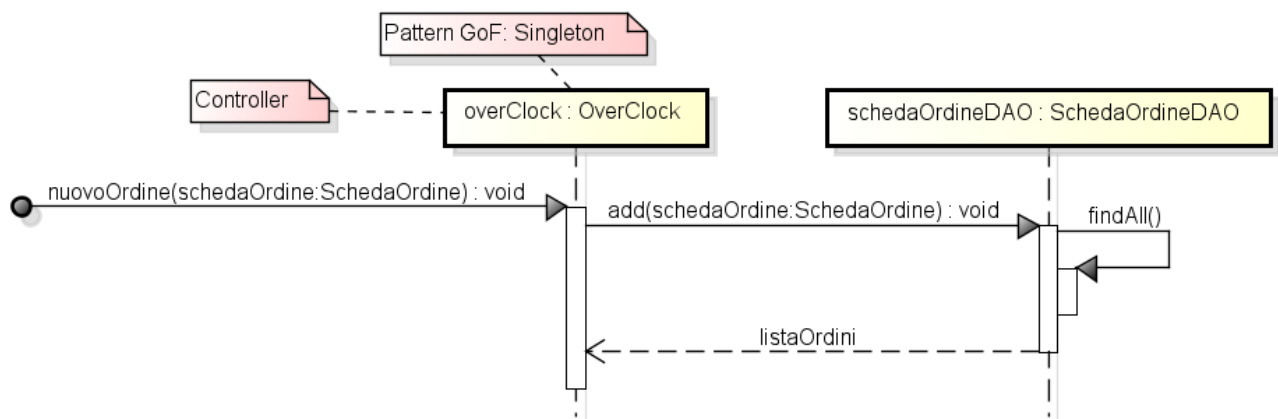
Conferma e registrazione della vendita



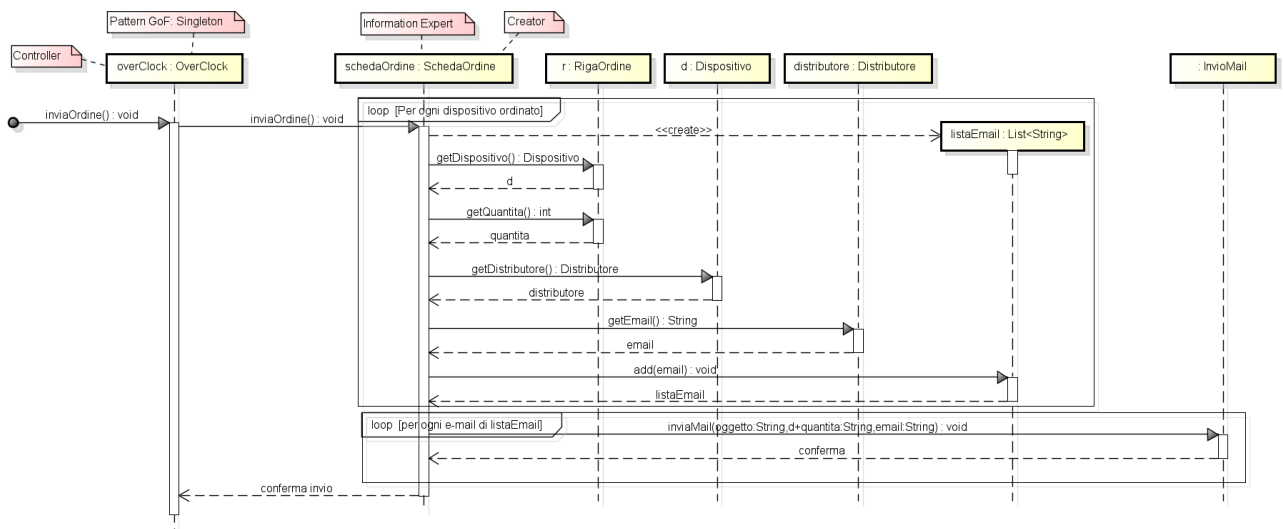
Aggiunta di un dispositivo all'ordine



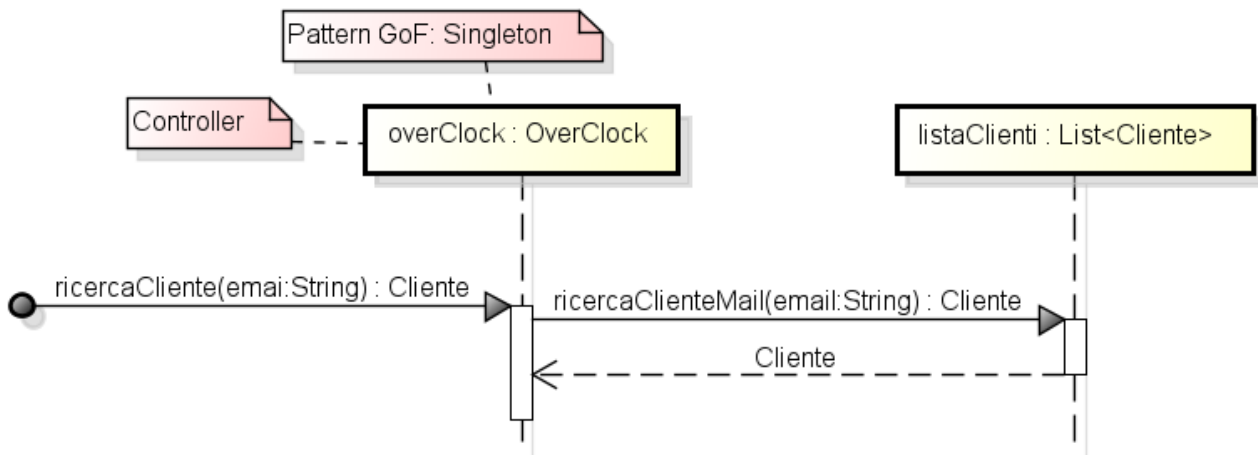
Conferma e registrazione dell'ordine



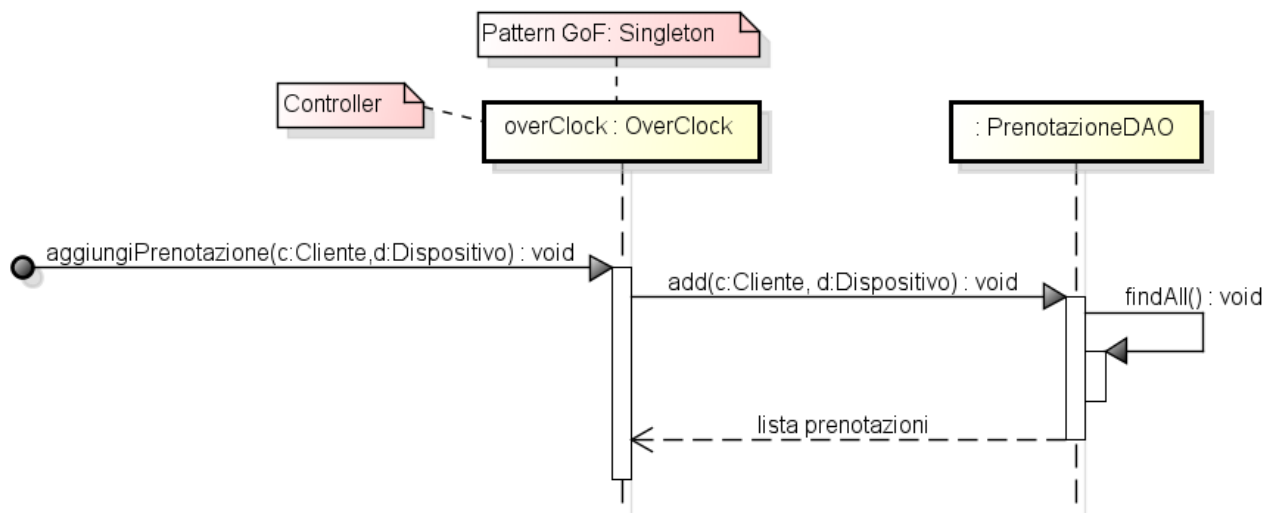
Invio dell'e-mail ai distributori interessati nell'ordine



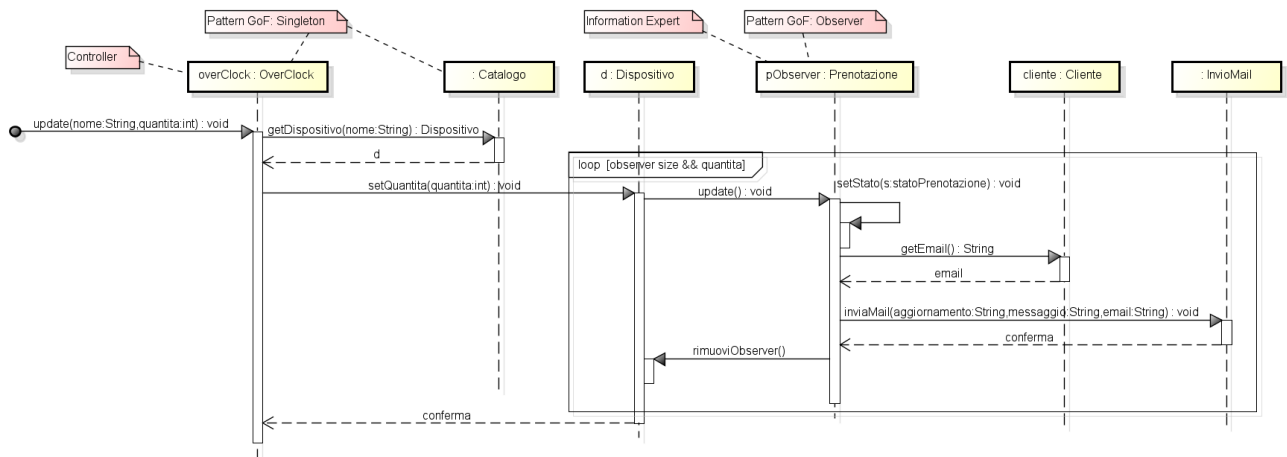
Ricerca del cliente in registro



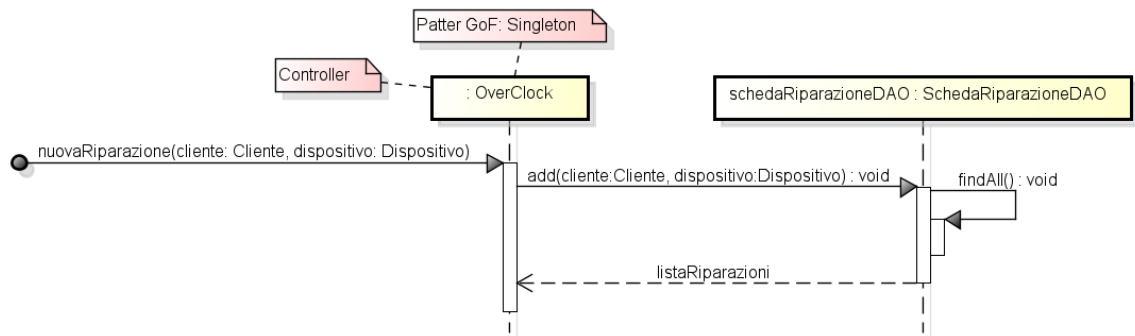
Conferma e registrazione della prenotazione



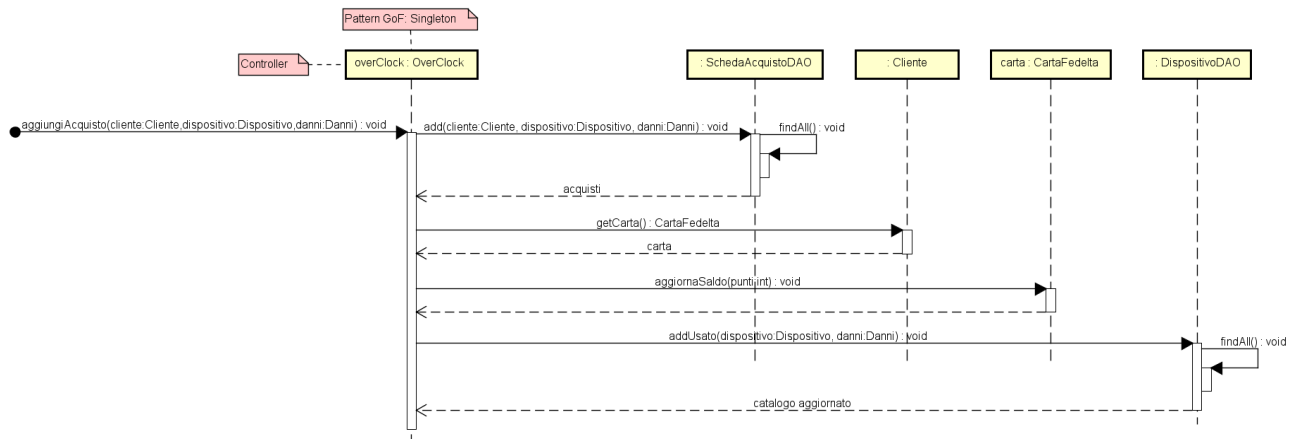
Observer per verificare lo stato della prenotazione



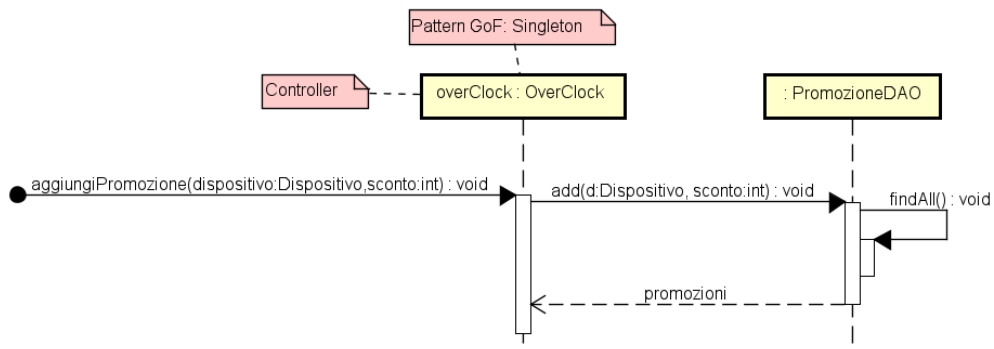
Conferma e registrazione del preventivo



Conferma e registrazione acquisto usato



Conferma e registrazione della promozione



4 TESTING

4.1 INTRODUZIONE

Per lo sviluppo di un programma efficiente e robusto si effettuano una serie di test. La fase di testing è essenziale per ridurre al minimo gli errori, i malfunzionamenti e i costi di manutenzione. Tuttavia, le combinazioni di input da prendere in considerazione sono innumerevoli. Per cui, per un testing accurato si effettuano due tipi di procedure:

- I test unitari (Unit Test) → Sono test effettuati direttamente sul codice, per verificarne la correttezza. L'idea dell'Unit Test in Java è valutare ogni metodo in funzione di valori attesi. Per effettuare tali test esistono vari tool, il più utilizzato è il framework JUnit.
- I test funzionali → Sono dei test che vanno a verificare che il sistema software nella sua completezza funzioni correttamente. Si danno degli input e si verifica 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 e quindi di seguire un approccio BottomUp collaudando prima le piccole unità che comporranno poi il programma completo.

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'aggiornamento degli articoli in magazzino e del calcolo dei fattori economici quali: quantità, calcolo del subtotal e del totale. Per ciascuno sono state individuate le classi di equivalenza tali da poter ridurre il numero di test al minimo insieme necessario, in particolare:

- Vendite
 - aggiungiDispositivo
 - Viene inserita una riga di vendita.
 - Viene verificato l'inserimento del dispositivo nella riga di vendita.
 - rimuoviDispositivo
 - Viene verificato la rimozione di un dispositivo dalla riga di vendita.
 - setTotale
 - Viene inserito un importo totale.
 - Viene verificato che il costo totale della vendita effettivo sia quello previsto.
 - getCarta
 - Viene verificato che la carta fedeltà inserita durante la vendita sia quella prevista.
 - getSubTotale
 - Viene verificato che il costo della riga di vendita effettivo sia quello previsto del dispositivo.

- getDispositivo
 - Viene verificato che il dispositivo inserito durante la vendita sia quello previsto.
- Prenotazioni
 - setCliente
 - Dopo l'inserimento del cliente si verifica che sia quello previsto.
 - getCliente
 - Si verifica che il cliente ritornato associato ad una riparazione sia quello previsto.
 - setDispositivo
 - Dopo l'inserimento del dispositivo si verifica che sia quello previsto.
 - getDispositivo
 - Si verifica che il dispositivo ritornato associato ad una riparazione sia quello previsto.
 - setStato
 - Dopo l'inserimento dello stato si verifica che sia quello previsto.
 - getStato
 - Si verifica che lo stato ritornato associato ad una riparazione sia quello previsto.
- Distributori
 - setEmail
 - Dopo l'inserimento di una e-mail per un distributore viene verificato che sia quella prevista.
 - getEmail
 - Viene verificata che l'e-mail associata ad un distributore che viene ritornata sia quella prevista.
- Ordini
 - aggiungiOrdine
 - Viene verificato che venga incrementata la quantità di un articolo ordinato.
 - Se è un articolo nuovo viene verificato che la quantità sia 1.
 - rimuoviOrdine
 - Viene verificato che venga decrementata la quantità di un articolo ordinato.
 - Se è un articolo ha quantità 1 viene rimosso.
 - setGetQuantita
 - Viene verificato che la quantità inserita di un articolo in una riga d'ordine sia quella prevista.
 - Viene verificato che la quantità ritornata di un articolo in una riga d'ordine sia quella prevista.
 - getDispositivo
 - Viene l'inserimento dell'articolo in una riga d'ordine sia corretto.
 - getSubTotale
 - Viene verificato che il sub-totale di una riga d'ordine sia quello previsto per l'articolo inserito.
- Riparazioni
 - getCliente

- Viene che il cliente associato alla riparazione venga ritornato in maniera corretta.
 - getDispositivo
 - Viene che il dispositivo associato alla riparazione venga ritornato in maniera corretta.
 - getStato
 - Viene verificato che lo stato della riparazione ritornato sia lo stato previsto per l'aggiornamento della riparazione.
 - notificaCliente
 - Viene verificato che l'e-mail del cliente e il messaggio vengano visualizzate in maniera corretta, prima dell'invio.
 - aggiungiPezzo
 - Dopo l'aggiornamento del preventivo di riparazione viene verificato che il pezzo inserito sia quello previsto.
 - setGetPezzi
 - Si verifica l'inserimento della lista pezzi di una scheda riparazione.
 - Si verifica che la lista di pezzi inseriti relativi ad una scheda riparazione sia quella prevista.
 - setGetCosto
 - Viene verificato che l'inserimento del costo totale di una riparazione venga effettuato in maniera corretta.
 - Viene verificato che il costo di una scheda di riparazione venga ritornato in maniera corretta.
- Promozioni
 - getDispositivo
 - Viene verificato che il dispositivo associato allo sconto sia quello previsto.
 - getSconto
 - Viene verificato che lo sconto associato ad un dispositivo sia quello previsto.
- Catalogo
 - setListaDispositivi
 - Si verifica che la lista dei dispositivi inserita nel catalogo sia quella prevista.
 - getListaDispositivi
 - Si verifica che la lista dei dispositivi ritornata dal catalogo sia quella prevista.
 - getListaDispositiviUsati
 - Si verifica che la lista di dispositivi ritornati dal catalogo sia solamente quella di dispositivi usati.
 - getDispositivoByDistributore
 - Viene verificato che la lista di dispositivi ritornati dal catalogo sia quella associata al distributore inserito.
- Cliente
 - setCartaFedelta
 - Viene verificato che la carta fedeltà inserita per un cliente sia quella prevista.

- Viene verificato che se un cliente si registra senza carta fedeltà, il valore di quest'ultima venga impostato a null.

4.3 TEST DI SISTEMA

Prima di giungere alla fase di refactoring, si è deciso di procedere con l'esecuzione di alcuni test di sistema. I test sono stati effettuati manualmente provando il programma e i principali casi d'uso. A seguito dei test non sono emerse particolari problematiche, vengono riportate di seguito alcune di queste:

- Gli articoli non hanno un prezzo di ordine ma solamente di vendita.
- È possibile cambiare lo stato della riparazione direttamente da "preventivo in corso" a "riparazione in corso", senza passare per "preventivo concluso".
- Le riparazioni non hanno la manodopera inclusa nel totale.

5 REFACTORING E CONCLUSIONI

5.1 DATABASE E REFACTORING

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

- Correzione delle anomalie individuate con il precedente Test di Sistema.
- Aggiunta di funzionalità all'applicazione al fine di completare le operazioni dei casi d'uso rimaste in sospeso.
- Correzione di tutta l'interfaccia grafica dell'applicazione.
- Prova di comunicazione con il database SQL e ricezione corretta di tutti i dati salvati.

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, 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 e i loro scenari alternativi.