Progetto: PC Ready

Sommario

[**Ideazione e analisi dei requisiti** 2](#_Toc32678153)

[Requisiti 2](#_Toc32678154)

[Caso d’uso 1: Crea configurazione cliente 3](#_Toc32678155)

[Caso d’uso 2: Crea bundle 5](#_Toc32678156)

[Caso d’uso 3: Inserisci nuovo componente 8](#_Toc32678157)

[Caso d’uso 4: Inserisci copia componente 10](#_Toc32678158)

[Caso d’uso 5: Effettua acquisto 11](#_Toc32678159)

[Caso d’uso 6: Registrazione cliente 13](#_Toc32678160)

[Caso d’uso 7: Login 14](#_Toc32678161)

[Caso d’uso 8: Rimozione componente 15](#_Toc32678162)

[Caso d’uso 9: Crea promozione 16](#_Toc32678163)

[Caso d’uso 10: Modifica ordine 17](#_Toc32678164)

[Regole di dominio 18](#_Toc32678165)

[Glossario 19](#_Toc32678166)

[**Documento di visione** 21](#_Toc32678167)

[Introduzione 21](#_Toc32678168)

[Scopo 21](#_Toc32678169)

[Portata 22](#_Toc32678170)

[Definizioni, acronimi e abbreviazioni 22](#_Toc32678171)

[Posizionamento 22](#_Toc32678172)

[Parti interessate e descrizioni utente 23](#_Toc32678173)

[Sviluppo di PC Ready 24](#_Toc32678174)

[Riepilogo veloce delle iterazioni 24](#_Toc32678175)

[Modello di dominio 25](#_Toc32678176)

[SSD e Contratti 27](#_Toc32678177)

[UC1 27](#_Toc32678178)

[UC2 30](#_Toc32678179)

[UC3 31](#_Toc32678180)

[UC4 32](#_Toc32678181)

[UC5 34](#_Toc32678182)

[UC6 35](#_Toc32678183)

[UC7 36](#_Toc32678184)

[UC8 36](#_Toc32678185)

[UC9 36](#_Toc32678186)

[UC10 37](#_Toc32678187)

# **Ideazione e analisi dei requisiti**

## Requisiti

Il proprietario di un negozio di informatica vuole permettere ai propri clienti di assemblare il proprio PC desktop, facendo scegliere loro le componenti desiderate, mediante un’ applicazione di facile utilizzo.

Tale applicazione deve essere in grado di mostrare al cliente tutte le componenti per PC per categoria, mostrando per ciascuna componente il prezzo, la disponibilità, la compatibilità con altre componenti (Esempio: CPU AMD Ryzen 3 compatibile con socket AM4) e il consumo energetico.

Il proprietario del negozio, gradirebbe che l’applicazione impedisca al cliente di assemblare un PC con componenti incompatibili, o con consumo energetico superiore rispetto al PSU scelto.

Oltre alle singole componenti, l’applicazione deve prevedere delle soluzioni già pronte, ideate dal proprietario per i clienti che non sanno con quali componenti dover assemblare un desktop: tali soluzioni devono essere divise in base alla fascia di prezzo – prestazioni offerte.

Si vuole consentire ai clienti di visionare dei preventivi sui pezzi/sulla configurazione scelta senza obbligo d’acquisto: saranno richiesti i dati del cliente (dati anagrafici, indirizzo e-mail e indirizzo di spedizione) solo se l’acquisto viene confermato.

Il proprietario, grazie ai suoi accordi con i dovuti istituti bancari, vuole consentire ai propri clienti di pagare con la loro carta di credito, con carte prepagate, ed eventualmente mediante contrassegno.

Infine, i prezzi dei singoli componenti sono basati su un listino esistente, tuttavia, se un componente non è presente in magazzino potrebbe essere aggiunto un sovrapprezzo dovuto alla necessità di ordinare il pezzo: tale informazione deve essere notificata all’acquirente.

## Caso d’uso 1: Crea configurazione cliente

|  |  |
| --- | --- |
| **Nome del caso d’uso** | **UC1: Crea configurazione cliente** |
| **Portata** | Applicazione PC Ready |
| **Livello** | Obiettivo utente |
| **Attore primario** | Cliente |
| **Parti interessate e interessi** | * **Cliente:** vuole creare una configurazione PC desktop personalizzata scegliendo i pezzi da lui desiderati * **Sistema:** vuole mostrare al cliente tutte le componenti disponibili o ordinabili per comporre un PC desktop. |
| **Pre-condizioni** | Il cliente ha selezionato l’opzione di creazione di una configurazione. |
| **Garanzia di successo** | La creazione della configurazione si conclude con successo se:   * Non vi sono incompatibilità tra le componenti della configurazione * Il PSU (alimentatore) scelto rientra nei consumi energetici della configurazione * Sono presenti tutti i pezzi fondamentali (CPU – GPU – Motherboard – Storage – Memoria – Case) per il corretto funzionamento del PC desktop |
| **Scenario principale di successo** | 1. Il cliente seleziona l’opzione per la creazione di un PC desktop. 2. Il sistema inizializza una nuova configurazione vuota, impostando dei valori di default per prezzo, nome e consumo energetico. 3. Il sistema restituisce al cliente tutte le categorie di componenti 4. Il cliente seleziona una categoria di componenti 5. Il sistema restituisce al cliente una lista di tutte le componenti della categoria selezionata 6. Il cliente seleziona un prodotto della lista di componenti 7. Il sistema restituisce al cliente tutte le informazioni relative al componente selezionato: per ciascun componente verranno mostrati il prezzo, il consumo energetico, le eventuali specifiche di compatibilità a seconda della categoria del componente stesso, e una breve descrizione. 8. Il cliente aggiunge il componente scelto alla configurazione 9. Il sistema esegue il controllo della compatibilità del componente scelto con i componenti già scelti per la configurazione, e in caso di successo aggiorna i valori di prezzo e di consumo energetico della configurazione e li mostra al cliente   I passi da 4 a 9 vengono ripetuti finché servono   1. Il cliente seleziona l’opzione per confermare il termine “dell’assemblaggio” della configurazione. 2. Il sistema esegue i controlli sulla presenza dei componenti fondamentali, e controlli sul consumo energetico ai fini di garantire al cliente che la configurazione da lui creata possa essere effettivamente funzionante. Conclusi tali controlli il sistema mostra al cliente il costo totale della configurazione e un riepilogo delle componenti selezionate. 3. Il cliente conferma la configurazione da lui creata. |
| **Estensioni** | **a9:** Il cliente vuole rimuovere un componente non più gradito   1. Il cliente seleziona la componente da rimuovere dalla configurazione attuale 2. Il sistema conferma la rimozione, aggiornando i valori di prezzo e consumo energetico della configurazione, dunque, consente al cliente la scelta di un nuovo componente (ripresa passi 4 e 5 del flusso principale)   **a9:** Il sistema rileva una incompatibilità del componente da inserire   1. Il sistema cerca una componente alternativa compatibile con le componenti già scelte dal cliente. 2. Il cliente può cercare una nuova componente, e decidere se seguire il consiglio fornitogli dal sistema, o scegliere una nuova componente. 3. Il sistema esegue un nuovo controllo di compatibilità.   I punti 1 a 3 si ripetono fino a quando non viene inserita una componente compatibile.   1. Il sistema aggiorna i valori di prezzo e consumo energetico della configurazione attuale, e aggiunge il nuovo componente sostitutivo (ripresa passi 4 e 5 del flusso principale).   **b11:** Il sistema rileva un consumo energetico eccessivo rispetto alla potenza massima fornita dal PSU scelto   1. Il sistema mostra al cliente un PSU compatibile con il consumo energetico della sua configurazione, e rimuove il precedente risultato inadeguato. 2. Il cliente cerca un nuovo PSU, e decide se seguire il consiglio fornitogli dal sistema, o scegliere un PSU diverso. 3. Il sistema sostituisce il vecchio PSU, con quello nuovo scelto dal cliente.   **c11:** Il sistema rileva la mancanza di una delle componenti fondamentali per il funzionamento del PC desktop   1. Il sistema mostra la categoria di componenti mancante al cliente, impedendo la conferma della configurazione. 2. Il sistema consente al cliente la scelta di un nuovo componente sostitutivo (ripresa passi 4 e 5 del flusso principale) |
| **Requisiti speciali** | Nessun requisito speciale individuato. |
| **Tecnologie adoperate** | Nessuna tecnologia specifica individuata. |
| **Problemi aperti** | * Come fa il sistema a riconoscere un’incompatibilità tra componenti della configurazione? |

## Caso d’uso 2: Crea bundle

|  |  |
| --- | --- |
| **Nome del caso d’uso** | **UC2: Crea bundle** |
| **Portata** | Applicazione PC Ready |
| **Livello** | Obiettivo utente |
| **Attore primario** | Amministratore del sistema |
| **Parti interessate e interessi** | * **Amministratore:** vuole creare una configurazione PC desktop, direttamente acquistabile da un cliente, inserendo un prezzo che possa differire dall’acquisto delle singole componenti. * **Sistema:** vuole mostrare all’amministratore tutte le componenti disponibili per assemblare il nuovo PC desktop, e al termine inserire quest’ultimo tra i “bundle acquistabili”. |
| **Pre-condizioni** | L’amministratore ha selezionato l’opzione di creazione di un “bundle acquistabile”. |
| **Garanzia di successo** | La creazione del bundle acquistabile si conclude con successo se:   * Non vi sono incompatibilità tra le componenti della configurazione creata. * Il PSU (alimentatore) scelto rientra nei consumi energetici della configurazione assemblata. * Sono presenti tutti i pezzi fondamentali (CPU – GPU – Motherboard – Storage – Memoria – Case) per il corretto funzionamento del PC desktop |
| **Scenario principale di successo** | 1. Un amministratore seleziona l’opzione per la creazione di un PC desktop. 2. Il sistema inizializza una nuova configurazione vuota, impostando dei valori di default per prezzo e consumo energetico. 3. Il sistema restituisce all’amministratore tutte le categorie di componenti 4. L’amministratore seleziona una categoria di componenti 5. Il sistema restituisce all’amministratore una lista di tutte le componenti della categoria selezionata 6. L’amministratore seleziona un prodotto della lista di componenti 7. Il sistema restituisce all’amministratore tutte le informazioni relative al componente selezionato: per ciascun componente verranno mostrati il prezzo, il consumo energetico, le eventuali specifiche di compatibilità a seconda della categoria del componente stesso, e una breve descrizione. 8. L’amministratore seleziona un componente da aggiungere alla configurazione. 9. Il sistema esegue il controllo della compatibilità del componente scelto con i componenti già scelti per la configurazione, e in caso di successo aggiorna i valori di prezzo e di consumo energetico della configurazione e li mostra all’amministratore.   I passi 4 e 9 vengono ripetuti finché servono   1. L’amministratore seleziona l’opzione per confermare il termine “dell’assemblaggio” della configurazione. 2. Il sistema esegue i controlli sulla presenza dei componenti fondamentali, e controlli sul consumo energetico ai fini di garantire all’amministratore che la configurazione da lui creata possa essere effettivamente funzionante. Conclusi tali controlli il sistema mostra al cliente il costo totale della configurazione e un riepilogo delle componenti selezionate. 3. L’amministratore conferma la configurazione da lui creata. 4. L’amministratore inserisce il nome, il fattore di sconto e la descrizione del bundle che conterrà la nuova configurazione creata. 5. Il sistema crea un nuovo bundle relativo alla configurazione creata. |
| **Estensioni** | **a9:** L’amministratore vuole rimuovere un componente non più gradito   1. L’amministratore seleziona la componente da rimuovere dalla configurazione attuale. 2. Il sistema conferma la rimozione, aggiornando i valori di prezzo e consumo energetico della configurazione, dunque, consente all’amministratore la scelta di un nuovo componente (ripresa passi 4 e 5 del flusso principale).   **a9:** Il sistema rileva una incompatibilità del componente da inserire   1. Il sistema cerca una componente alternativa compatibile con le componenti già scelte dall’amministratore. 2. L’amministratore può cercare una nuova componente, e decidere se seguire il consiglio fornitogli dal sistema, o scegliere una nuova componente. 3. Il sistema esegue un nuovo controllo di compatibilità.   I punti 1 a 3 si ripetono fino a quando non viene inserita una componente compatibile.   1. Il sistema aggiorna i valori di prezzo e consumo energetico della configurazione attuale, e aggiunge il nuovo componente sostitutivo (ripresa passi 4 e 5 del flusso principale).   **b11:** Il sistema rileva un consumo energetico eccessivo rispetto alla potenza massima fornita dal PSU scelto   1. Il sistema mostra all’amministratore un PSU compatibile con il consumo energetico della sua configurazione, e rimuove il precedente risultato inadeguato. 2. L’amministratore cerca un nuovo PSU, e decide se seguire il consiglio fornitogli dal sistema, o scegliere un PSU diverso. 3. Il sistema sostituisce il vecchio PSU, con quello nuovo scelto dall’amministratore.   **c11:** Il sistema rileva la mancanza di una delle componenti fondamentali per il funzionamento del PC desktop   1. Il sistema mostra la categoria di componenti mancante all’amministratore, impedendo la conferma della configurazione. 2. Il sistema consente all’amministratore la scelta di un nuovo componente sostitutivo (ripresa passi 4 e 5 del flusso principale) |
| **Requisiti speciali** | Nessun requisito speciale individuato. |
| **Tecnologie adoperate** | Nessuna tecnologia specifica individuata. |
| **Problemi aperti** | * Come fa il sistema a riconoscere un’incompatibilità tra componenti della configurazione? * Come fa il sistema a gestire i “bundle acquistabili”? |

## Caso d’uso 3: Inserisci nuovo componente

|  |  |
| --- | --- |
| **Nome del caso d’uso** | **UC3: Inserisci nuovo componente** |
| **Portata** | Applicazione PC Ready |
| **Livello** | Obiettivo utente, CRUD |
| **Attore primario** | Amministratore del sistema |
| **Parti interessate e interessi** | * **Amministratore:** vuole un rapido inserimento di un nuovo componente e della quantità disponibile di quest’ultimo. * **Sistema:** vuole assegnare a ciascun componente delle specifiche dettagliate e generare per ciascuno di questi un codice univoco. |
| **Pre-condizioni** | Il sistema deve essere funzionante e l’amministratore deve averne pieno accesso ed essersi autenticato. |
| **Garanzia di successo** | Viene inserito nel sistema un nuovo componente, al quale viene associato un codice identificativo univoco. |
| **Scenario principale di successo** | 1. L’amministratore richiede al sistema l’inserimento di un nuovo componente. 2. Il sistema chiede all’amministratore le informazioni relative al nuovo componente da inserire. 3. L’amministratore fornisce al sistema nome, categoria, consumo energetico, prezzo e descrizione del componente 4. Il sistema inserisce la nuova componente e chiede all’amministratore il numero di copie del componente da inserire. 5. L’amministratore fornisce al sistema il numero di copie del componente da inserire. 6. Il sistema inserisce un numero di copie del componente pari alla quantità specificata. 7. Il sistema mostra all’amministratore i codici univoci di ciascuna copia componente inserita, e stampa   un’etichetta (contenente il codice univoco) per ciascuna di esse.   1. L’amministratore termina l’operazione di inserimento |
| **Estensioni** | Nessuna estensione individuata. |
| **Requisiti speciali** | Nessun requisito speciale individuato. |
| **Tecnologie adoperate** | Nessuna tecnologia specifica individuata. |
| **Problemi aperti** | * Come fa il sistema a stampare le etichette contenenti il codice univoco dei componenti inseriti? |

## Caso d’uso 4: Inserisci copia componente

|  |  |
| --- | --- |
| **Nome del caso d’uso** | **UC4: Inserisci copia componente** |
| **Portata** | Applicazione PC Ready |
| **Livello** | Obiettivo utente, CRUD |
| **Attore primario** | Amministratore del sistema |
| **Parti interessate e interessi** | * **Amministratore:** vuole un rapido inserimento di una o più copie di un componente già esistente * **Sistema:** vuole assegnare al componente identificato dall’amministratore un numero di nuove copie ben definito, e assegnare a ciascuna delle suddette copie un identificativo univoco. |
| **Pre-condizioni** | La componente indicata dall’amministratore deve essere presente nella memoria del sistema, ovvero il sistema deve possedere un’istanza esistente di componente per poter assegnare ad esso delle nuove copie. |
| **Garanzia di successo** | Viene inserito nel sistema una nuova (o più ) copie di un componente, alle quali viene associato un codice identificativo univoco. |
| **Scenario principale di successo** | 1. L’amministratore richiede al sistema l’inserimento di una nuova copia di un componente, fornendone il codice e la categoria. 2. Il sistema individuato il componente esistente, chiede all’amministratore il numero di copie da inserire. 3. L’amministratore fornisce al sistema la quantità di copie da inserire. 4. Il sistema inserisce un numero di copie del componente pari alla quantità specificata. 5. Il sistema mostra all’amministratore i codici univoci di ciascuna copia componente inserita, e stampa un etichetta (contenente il codice univoco) per ciascuna di esse. 6. L’amministratore termina l’operazione di inserimento |
| **Estensioni** | Nessuna estensione individuata. |
| **Requisiti speciali** | Nessun requisito speciale individuato. |
| **Tecnologie adoperate** | Nessuna tecnologia specifica individuata. |

## Caso d’uso 5: Effettua acquisto

|  |  |
| --- | --- |
| **Nome del caso d’uso** | **UC5: Effettua acquisto** |
| **Portata** | Applicazione PC Ready |
| **Livello** | Obiettivo utente |
| **Attore primario** | Cliente |
| **Parti interessate e interessi** | * **Cliente:** vuole effettuare un acquisto di una componente generica esistente nel sistema. * **Sistema:** vuole mostrare al cliente tutte le componenti disponibili o ordinabili. |
| **Pre-condizioni** | Il cliente ha selezionato l’opzione di acquisto di una componente generica |
| **Garanzia di successo** | Il cliente ha effettuato il pagamento delle componenti scelte e ha finalizzato l’acquisto fornendo tutte le informazioni necessarie per la spedizione dei prodotti. |
| **Scenario principale di successo** | 1. Il cliente seleziona l’opzione di acquisto 2. Il sistema restituisce al cliente un catalogo di tipologie di prodotti disponibili (ovvero categorie di componenti o bundle) 3. Il cliente seleziona una tipologia disponibile 4. Il sistema restituisce al cliente una lista di prodotti in base alla tipologia selezionata dal cliente. 5. Il cliente seleziona un prodotto dalla lista. 6. Il sistema restituisce al cliente tutte le informazioni relative al prodotto selezionato. 7. Il cliente chiede al sistema di aggiungere il prodotto al carrello. 8. Il sistema, dopo aver controllato la presenza di una copia del componente, conferma l’avvenuto inserimento del prodotto nel carrello, aggiorna il costo totale dei prodotti contenuti in quest’ultimo e restituisce al cliente la lista dei prodotti contenuti e il costo totale.   I passi da 1 a 8 vengono ripetuti fino a quando il cliente vuole continuare ad acquistare componenti.   1. Il cliente seleziona l’opzione per l’acquisto degli articoli contenuti nel carrello 2. Il sistema chiede al cliente le informazioni sulla spedizione (indirizzo di spedizione, città, CAP) 3. Il cliente inserisce le informazioni richieste dal sistema 4. Il sistema chiede al cliente la modalità di pagamento, mostrando una lista di possibili opzioni, e chiedendo al cliente le informazioni relative al sistema di pagamento scelto (numero carta, codice a tre cifre) 5. Il cliente fornisce i dati relativi alla modalità di pagamento e conferma l’acquisto 6. Il sistema notifica il successo dell’operazione |
| **Estensioni** | **a8:** Il sistema rileva che non sono presenti copie del componente selezionato dal cliente   1. Il sistema notifica al cliente l’assenza di copie del componente scelto, e impedisce l’inserimento nel carrello del componente 2. Il sistema consente al cliente di scegliere una nuova categoria di prodotti (ripresa dal passo 2 del flusso principale).   **b9:** Il cliente vuole rimuovere un componente non più gradito   1. Il cliente seleziona la componente da rimuovere dal suo carrello attuale 2. Il sistema conferma la rimozione, aggiornando la lista delle componenti presente nel carrello, il numero dei pezzi e il prezzo totale delle componenti, e infine consente al cliente la scelta di un nuovo componente presente in catalogo (ripresa dal passo 2 del flusso principale). |
| **Requisiti speciali** | Nessun requisito speciale individuato. |
| **Tecnologie adoperate** | Nessuna tecnologia specifica individuata. |

## Caso d’uso 6: Registrazione cliente

|  |  |
| --- | --- |
| **Nome del caso d’uso** | **UC6: Registrazione cliente** |
| **Portata** | Applicazione PC Ready |
| **Livello** | Obiettivo utente, CRUD |
| **Attore primario** | Cliente |
| **Parti interessate e interessi** | * **Cliente:** vuole effettuare la registrazione al servizio fornito dall’applicazione PC Ready. * **Sistema:** vuole tutte le informazioni relative al nuovo cliente da registrare |
| **Pre-condizioni** | Non deve esistere l’istanza del suddetto cliente in memoria del sistema PC Ready |
| **Garanzia di successo** | La nuova istanza di cliente viene mantenuta nella memoria del sistema. |
| **Scenario principale di successo** | 1. Il cliente richiede al sistema di poter effettuare una nuova registrazione. 2. Il sistema richiede i dati del cliente (nome, cognome, e-mail, password ecc.). 3. Il cliente inserisce tutti i suoi dati. 4. Il sistema controlla se la mail inserita dall’utente è già presente nel sistema, e se la password coincide con il campo “conferma password” 5. Il sistema notifica al cliente l’avvenuta registrazione mostrando un riepilogo delle informazioni non sensibili |
| **Estensioni** | **a4:** Il sistema rileva una mail già esistente in memoria   1. Il sistema notifica al cliente che la mail è già in uso e chiede al cliente nuovi dati per la registrazione 2. Il cliente inserisce un nuovo indirizzo e-mail per completare la registrazione 3. Il sistema controlla nuovamente l’indirizzo inserito   **a4:** Il sistema rileva che il campo password e il campo conferma password non coincidono   1. Il sistema notifica al cliente che i campi password non coincidono e chiede al cliente nuovi dati per la registrazione 2. Il cliente inserisce nuovi dati per la registrazione 3. Il sistema controlla nuovamente i dati inseriti dall’utente |
| **Requisiti speciali** | Nessun requisito speciale individuato. |
| **Tecnologie adoperate** | Nessuna tecnologia specifica individuata. |

## Caso d’uso 7: Login

|  |  |
| --- | --- |
| **Nome del caso d’uso** | **UC7: Login** |
| **Portata** | Applicazione PC Ready |
| **Livello** | Obiettivo utente, CRUD |
| **Attore primario** | Utente generico |
| **Parti interessate e interessi** | * **Utente generico:** vuole effettuare il login per usufruire del servizio fornito dall’applicazione PC Ready. * **Sistema:** vuole tutte le informazioni relative all’utente da autenticare |
| **Pre-condizioni** | Deve esistere l’istanza del suddetto utente in memoria del sistema PC Ready |
| **Garanzia di successo** | L’utente generico viene riconosciuto come un cliente o come amministratore, e può usufruire del servizio |
| **Scenario principale di successo** | 1. L’utente generico richiede al sistema di poter effettuare un login, specificando se vuole eseguire il login come amministratore o come cliente 2. Il sistema richiede i dati dell’utente (e-mail e password). 3. Il cliente inserisce la tipologia di login e tutti i suoi dati. 4. Il sistema notifica al cliente l’avvenuto login mostrando un messaggio di benvenuto |
| **Estensioni** | **a4:** Il sistema rileva che i dati inseriti dall’utente sono errati   1. Il sistema mostra all’utente un messaggio di errore di login e invita l’utente ad inserire nuovamente le sue credenziali |
| **Requisiti speciali** | Nessun requisito speciale individuato. |
| **Tecnologie adoperate** | Nessuna tecnologia specifica individuata. |

## Caso d’uso 8: Rimozione componente

|  |  |
| --- | --- |
| **Nome del caso d’uso** | **UC8: Rimozione componente** |
| **Portata** | Applicazione PC Ready |
| **Livello** | Obiettivo utente, CRUD |
| **Attore primario** | Amministratore del sistema |
| **Parte interessate e interessi** | * **Amministratore:** vuole eliminare un componente presente nel catalogo in quanto non più utile o disponibile. * **Sistema:** vuole tutte le informazioni relative al componente che l’amministratore vuole rimuovere dal catalogo |
| **Pre-condizioni** | L’amministratore deve avere effettuato il login al servizio, e la sua istanza deve essere stata recuperata. |
| **Garanzia di successo** | La componente individuata mediante i dati forniti è stata eliminata. |
| **Scenario principale di successo** | 1. L’amministratore richiede al sistema di poter rimuovere un componente dai prodotti disponibili. 2. Il sistema richiede il codice del componente da rimuovere. 3. L’amministratore fornisce al sistema il codice del componente da rimuovere. 4. Il sistema verifica la presenza del componente e lo rimuove insieme alle sue copie eventualmente presenti. 5. Il sistema rimuove Bundle e Configurazioni che contengono il componente appena eliminato |
| **Estensioni** | **a4:** Il sistema rileva che il codice fornito dall’amministratore non corrisponde a nessun componente esistente   1. Il sistema mostra all’amministratore un messaggio di errore e chiede l’inserimento di un nuovo codice componente valido 2. L’amministratore fornisce al sistema un nuovo codice del componente da rimuovere (ripresa del punto 3 del flusso principale) |
| **Requisiti speciali** | Nessun requisito speciale individuato. |
| **Tecnologie adoperate** | Nessuna tecnologia specifica individuata. |

## Caso d’uso 9: Crea promozione

|  |  |
| --- | --- |
| **Nome del caso d’uso** | **UC9: Crea promozione** |
| **Portata** | Applicazione PC Ready |
| **Livello** | Obiettivo utente, CRUD |
| **Attore primario** | Amministratore del sistema |
| **Parte interessate e interessi** | * **Amministratore:** vuole inserire una promozione per un prodotto, ovvero vuole metterlo in sconto. * **Sistema:** vuole tutte le informazioni relative al componente che l’amministratore vuole mettere in promozione |
| **Pre-condizioni** | Deve esistere l’istanza del componente da mettere in promozione in memoria del sistema PC Ready |
| **Garanzia di successo** | Viene modificato il fattore di promozione del componente selezionato dall’amministratore |
| **Scenario principale di successo** | 1. L’amministratore richiede al sistema di inserire una nuova promozione. 2. Il sistema richiede all’amministratore il prodotto sul quale applicare la promozione. 3. L’amministratore inserisce il codice del prodotto in promozione. 4. Il sistema chiede all’amministratore la percentuale di sconto da applicare. 5. L’amministratore inserisce la percentuale di sconto. 6. Il sistema applica percentuale di sconto ai prodotti indicatogli e comunica all’admin il costo dei prodotti scontati. |
| **Estensioni** | **a4:** Il sistema rileva che il codice fornito dall’amministratore non corrisponde a nessun componente esistente   1. Il sistema mostra all’amministratore un messaggio di errore e chiede l’inserimento di un nuovo codice componente valido 2. L’amministratore fornisce al sistema un nuovo codice del componente da rimuovere (ripresa del punto 3 del flusso principale) |
| **Requisiti speciali** | Nessun requisito speciale individuato. |
| **Tecnologie adoperate** | Nessuna tecnologia specifica individuata. |

## Caso d’uso 10: Modifica ordine

|  |  |
| --- | --- |
| **Nome del caso d’uso** | **UC10: Modifica ordine** |
| **Portata** | Applicazione PC Ready |
| **Livello** | Obiettivo utente, CRUD |
| **Attore primario** | Cliente |
| **Parte interessate e interessi** | * **Cliente:** vuole modificare le informazioni dell’indirizzo di spedizione associato ad un ordine effettuato in precedenza * **Sistema:** vuole conoscere tutte le informazioni relative all’ordine da modificare |
| **Pre-condizioni** | Il cliente deve essersi autenticato, e deve esistere un’istanza di ordine associata ad esso. |
| **Garanzia di successo** | Le informazioni relative all’ordine individuato sono state modificate. |
| **Scenario principale di successo** | 1. Il cliente chiede al sistema i suoi ordini effettuati 2. Il sistema restituisce al cliente i suoi ordini, e chiede al cliente l’identificativo dell’ordine da modificare e quali informazioni dell’ordine vuole modificare (indirizzo di spedizione, città, ecc..). 3. Il cliente fornisce al sistema le nuove informazioni sulla spedizione e l’identificativo dell’ordine da modificare 4. Il sistema registra le modifiche e conferma il buon esito dell’operazione. |
| **Estensioni** | **a2:** Il sistema rileva che il cliente non ha effettuato nessun ordine   1. Il sistema mostra all’utente un messaggio di errore e termina l’operazione.   **b4:** Il sistema rileva che il cliente ha inserito un codice ordine non valido   1. Il sistema mostra all’utente un messaggio di errore e chiede nuovamente all’utente l’inserimento di un nuovo identificativo dell’ordine da modificare e quali informazioni dell’ordine vuole modificare 2. Il cliente fornisce al sistema le nuove informazioni sulla spedizione e l’identificativo dell’ordine da modificare (ripresa del punto 3 del flusso principale) |
| **Requisiti speciali** | Nessun requisito speciale individuato. |
| **Tecnologie adoperate** | Nessuna tecnologia specifica individuata. |

## Regole di dominio

1. Una configurazione PC desktop necessita delle seguenti componenti fondamentali per un corretto funzionamento:
   * CPU
   * GPU
   * Motherboard
   * Storage
   * Memoria
   * Case
2. Una Motherboard si dice incompatibilecon una CPU se il socket della motherboard non riesce ad ospitare la CPU (ad esempio una CPU “AMD Ryzen 2400” è compatibile solo con motherboard con un socket “AM4”).
3. Una Memoria è incompatibile con una Motherboard se quest’ultima non è in grado di supportare la tipologia di RAM (DDR2, DDR3, DDR4)
4. Una configurazione PC desktop necessita di un Power Supply (alimentatore) la cui potenza fornita deve essere uguale o inferiore alla somma dei consumi energetici di tutte le componenti.
5. Coming soon.

## Glossario

|  |  |
| --- | --- |
| **CPU** | **Central Processing Unit**, è il “cuore pulsante” di un qualsiasi PC Desktop in quanto coordina in maniera centralizzata tutte le altre unità di elaborazione presenti, si occupa della gestione delle varie periferiche interne o schede elettroniche come schede audio, schede di rete o schede video |
| **GPU** | **Graphic Processing Unit**, è l'unità di elaborazione grafica conosciuta anche come unità di elaborazione visiva (VPU, in inglese visual processing unit) o processore grafico o processore visivo.  È una tipologia particolare di coprocessore che si contraddistingue per essere specializzato nel rendering di immagini grafiche. |
| **APU** | **Accelerated processing unit** (unità d'elaborazione accelerata) o APU, nel suo uso più comune si riferisce ad un singolo die che combina componenti CPU, GPU e PCIe: difatti in gergo comune si indica con APU un “processore con scheda grafica integrata”.  La GPU integrata in tale APU è solitamente meno performante rispetto ad una generica GPU integrata. |
| **Motherboard** | Una **scheda madre** detta anche scheda di sistema, in lingua inglese motherboard ("scheda madre") mainboard ("scheda principale") o in gergo “mobo”, è un tipo di scheda elettronica principale che raccoglie in sé tutta la circuiteria elettronica e i collegamenti di interfaccia tra i vari componenti interni principali di un personal computer come memoria e le altre schede elettroniche montate o alloggiate sopra, comprendendo anche i bus di espansione e le interfacce verso le periferiche esterne.  Le motherboard “consumer” per PC desktop sono disponibili in diversi formati, tra cui i più comuni sono:   * **ATX**: dimensione (305×244) * **micro ATX**: dimensione (244×244) * **mini ATX**: dimensione (150×150) |
| **PSU** | Un alimentatore, meglio conosciuto con il suo nome inglese **power supply unit (PSU)**, è un dispositivo elettrico che fornisce energia elettrica a un carico elettrico.  La funzione principale di un alimentatore è quella di convertire l'alimentazione CA in corrente continua regolata a bassa tensione per i componenti interni di un computer.  I PSU sono divisi per categorie di efficienza (dal meno al più efficiente) secondo il programma di efficienza 80 Plus:   * 80 Plus “white” * 80 Plus Bronze * 80 Plus Silver * 80 Plus Gold * 80 Plus Platinum |
| **Efficienza di un PSU** | Per spiegare cosa vuol dire efficienza per un PSU è necessario introdurre un breve esempio: si ipotizzi che un PSU eroghi 300W al sistema, ma in realtà assorba 375W dalla presa a muro.  Questo significa che l'efficienza è 375W/300W, che equivale allo 0,8 o all'80%: quei 75W non producono altro se non **calore**.  Più efficiente è un alimentatore, meno calore produce internamente e certamente minore è il suo consumo energetico. La quantità di calore prodotta è cruciale in quanto impatta sulle prestazioni dell'alimentatore e tutti i suoi componenti interni. Inoltre, oltre a questo, maggiore è il carico termico, più difficile è il compito del sistema di raffreddamento dell'alimentatore. |
| **Memoria RAM** | In elettronica e informatica la RAM (**Random Access Memory** ovvero memoria ad accesso casuale) è un tipo di memoria volatile caratterizzata dal permettere l'accesso diretto a qualunque indirizzo di memoria con lo stesso tempo di accesso.  Nella memoria RAM vengono copiati (caricati) i programmi che la CPU deve eseguire. |
| **Storage SSD** | Una unità di **memoria a stato solido** (in acronimo SSD dal corrispondente termine inglese solid-state drive), in elettronica e informatica, è un dispositivo di memoria di massa basato su semiconduttori, che utilizza memoria allo stato solido, in particolare memoria flash, per l'archiviazione dei dati.  A differenza di una memoria, un SSD in quanto esente da parti meccaniche comporta una serie di vantaggi:   * Rumorosità assente * Minore possibilità di guasto * Non necessitano di deframmentazione * Maggiore velocità di trasferimento dati * Minore produzione di calore |
| **Storage HDD** | Un disco rigido, detto anche **disco fisso** (abbreviato comunemente in hard disk con la sigla HDD), in elettronica e informatica indica un dispositivo di memoria di massa di tipo magnetico che utilizza uno o più dischi magnetizzati per l'archiviazione di dati e applicazioni (file, programmi e sistemi operativi). |
| **Configurazione** | Nel contesto in questione, per **configurazione** si intende un insieme di componenti tale da poter costituire un PC pienamente funzionante. |
| **Socket CPU** | Nell'hardware del computer, un **socket CPU** o uno slot CPU, contiene uno o più componenti meccanici che forniscono connessioni meccaniche ed elettriche tra un microprocessore (CPU) e la scheda madre stessa.  Ciò consente di posizionare e sostituire l'unità di elaborazione centrale (CPU) senza saldatura. |
| **Bundle** | PC desktop già assemblato, formato da una combinazione di più componenti, messo in vendita ad un prezzo vantaggioso. |

# **Documento di visione**

## Introduzione

L’obiettivo di questo elaborato è la creazione di un applicativo di nome “PC Ready”, che consenta agli acquirenti di un negozio di informatica di acquistare componenti per PC desktop, PC desktop pre-assemblati o di creare ed acquistare un PC con le componenti da questi desiderate.

## Scopo

Lo scopo del documento di visione è quello di definire, sottolineare e analizzare le peculiarità, le necessità e i servizi offerti da “PC Ready”.

In tale documento si possono osservare tutte le parti interessate e le loro necessità, ricordando che maggiori dettagli riguardo al soddisfacimento di queste ultime sono contenuti nel Modello dei casi d’uso.

## Portata

Il documento di visione ivi presente è relativo all’elaborato “PC Ready”, stilato e revisionato dagli studenti B.Caruso, G.Fallica e G.Costanzo.

L’obiettivo principale del gruppo è quello di soddisfare tutte le richieste del committente, così come tutti i requisiti non funzionali individuati nel corso dell’elaborato.

## Definizioni, acronimi e abbreviazioni

Per tutte le informazioni relative a definizioni, sigle, abbreviazioni e altro ancora, si faccia riferimento al Glossario posto sopra al documento di visione.

## Posizionamento

**Opportunità di business**: l’applicativo che verrà sviluppato consentirà al proprietario del negozio di informatica di raggiungere più clienti, fornendo loro un servizio di e-shop con la possibilità di ricevere comodamente a casa i prodotti acquistati.

Inoltre, tale applicativo permetterà al proprietario del negozio in questione di applicare promozioni ai prodotti inseriti mediante una semplice gestione di tutti i componenti presenti in magazzino.

La tabella posta in seguito mostra la **formulazione del problema:**

|  |  |
| --- | --- |
| **Descrizione del problema** | Il sistema precedentemente usato dal proprietario del negozio di informatica non permette a questi di poter acquisire una maggiore clientela.  Quando un cliente commissiona un PC Desktop, in quanto non vi è un meccanismo di controllo automatico sulla compatibilità delle componenti, si ha una probabilità elevata di comporre una configurazione non funzionante. |
| **Attori coinvolti** | Amministratore del sistema PC Ready, Clienti generici |
| **Impatto** | Processo di creazione di un PC desktop soggetto ad errori di compatibilità o insufficienza energetica, con conseguente danneggiamento delle componenti nei casi più gravi e insoddisfazione dei clienti. |
| **Benefici in caso di successo** | Informatizzazione della creazione di un PC desktop, con conseguente riduzione del tasso degli errori, e aumento della clientela raggiungibile garantito dal processo di vendita online. |

La tabella seguente mostra la **formulazione della posizione del prodotto:**

|  |  |
| --- | --- |
| **Destinatari** | Il prodotto è rivolto ad un negozio di informatica specializzato nella vendita di componenti per PC desktop |
| **Obbiettivi** | Il committente ha la necessità di un applicativo che consenta di effettuare delle vendite online, e che controlli la compatibilità delle componenti nel caso in cui si voglia assemblare un PC Desktop, azzerando la probabilità di commettere errori in tale fase. |
| **Tipologia** | Prodotto software personalizzato per la vendita, ma generale riguardo i controlli di compatibilità delle componenti PC. |
| **Funzione** | “PC Ready” offre una soluzione di gestione delle vendite online, integrata ad un servizio di controllo compatibilità delle componenti. |
| **Soluzioni alternative attuali** | Una soluzione simile per il controllo della compatibilità delle componenti risulta essere “PC Part Picker”, tuttavia questo applicativo web non consente di gestire le vendite o il magazzino di un negozio di informatica. |
| **Caratteristiche del prodotto** | Il prodotto in questione offre un livello di personalizzazione adatto ai prodotti venduti dal committente, aggiungendo delle funzionalità di controllo che agevolano il lavoro del committente e garantiscono ai clienti PC Desktop pronti all’uso. |

## Parti interessate e descrizioni utente

Ecco una breve lista delle parti interessate:

* **Committente:** è colui che ha commissionato lo sviluppo del prodotto.

Il suo compito è quello di fornire agli sviluppatori quante più informazioni possibili (requisiti) sul prodotto richiesto e feedback riguardo ad ogni versione dell’applicativo.

* **Amministratore del sistema:** è colui che si occupa della gestione del magazzino, della gestione delle promozioni e dei bundle acquistabili dai clienti.

Il suo compito è quello di utilizzare in prima persona l’applicativo sviluppato, fornendo feedback dettagliati agli sviluppatori.

* **Sviluppatore software:** è il responsabile dello sviluppo dell’applicativo.

Il suo compito è quello di progettare il prodotto software, cercando di soddisfare tutte le richieste del committente.

# **Sviluppo di PC Ready**

## Riepilogo veloce delle iterazioni

Lo sviluppo di PC Ready è stato svolto in modo iterativo, in modo da apportare delle modifiche graduali e ben controllate, ridurre gli errori di sviluppo, ed eventualmente permettere una più facile correzione di quest’ultimi.

La realizzazione dell’applicazione è stata articolata in quattro iterazione che hanno permesso di coprire la totalità dei casi d’uso mostrati nella parte iniziale di questo documento.

Ecco un breve dettaglio dei contenuti delle iterazioni:

* Iterazione 1
  + Sviluppo dei casi d’uso fondamentali che consentono ad amministratore ed utente di creare una configurazione (senza effettuare alcun controllo di compatibilità sulle componenti inserite), ovvero di UC1 e UC2
  + Sviluppo dei casi d’uso che consentono ad un amministratore di inserire nuove componenti e copie di quest’ultime, ovvero di UC3 e UC4
  + Sviluppo dei diagrammi di sequenza di sistema e dei diagrammi di sequenza per UC1,UC2,UC3,UC4
  + Implementazione dei casi d’uso sopra riportati
* Iterazione 2
  + Sviluppo della logica di compatibilità delle componenti di una configurazione così come previsto dalle estensioni di UC1 e UC2.
  + Riduzione delle responsabilità della classe SistemaPCReady mediante l’uso di controller di caso d’uso (by pattern Controller).
  + Rimozione della classe Categoria e introduzione della classe Catalogo, con conseguente introduzione dell’attributo “categoria” per la classe Componente.
  + Applicazione del pattern Composite per la gestione della Configurazione e delle specializzazioni della classe Componente.
  + Correzioni e migliorie del codice dell’iterazione precedente.
  + Implementazione della console comandi
    - Per maggiori informazioni riguardo a tale iterazioni si consiglia di consultare il documento “Discussione Iterazione 2”.
* Iterazione 3
* Sviluppo dei casi d’uso UC5, UC6 e UC7 in forma completa
* Aggiornamento del modello di dominio e diagramma delle classi di progetto con le nuove classi introdotte dai casi d’uso sopra citati
* Sviluppo dei diagrammi di sequenza di sistema e dei diagrammi di sequenza per UC5,UC6,UC7
* Implementazione dei casi d’uso UC5, UC6, UC7 con le rispettive estensioni
* Migliorie dell’interfaccia utente e aggiunta dei nuovi comandi di login, registrazione e acquisto
  + - Per maggiori informazioni riguardo a tale iterazioni si consiglia di consultare il documento “Discussione Iterazione 3”.
* Iterazione 4
  + Sviluppo dei casi d’uso UC8, UC9 e UC10 in forma completa
  + Sviluppo dei diagrammi di sequenza di sistema e dei diagrammi di sequenza per UC8,UC9,UC10
  + Implementazione di UC8,UC9,UC10 con le relative estensioni
  + Migliorie all’interfaccia utente, con aggiunta di nuovi comandi
  + Raffinamento dei test case con il comando: “@FixMethodOrder(MethodSorters.***NAME\_ASCENDING***)”

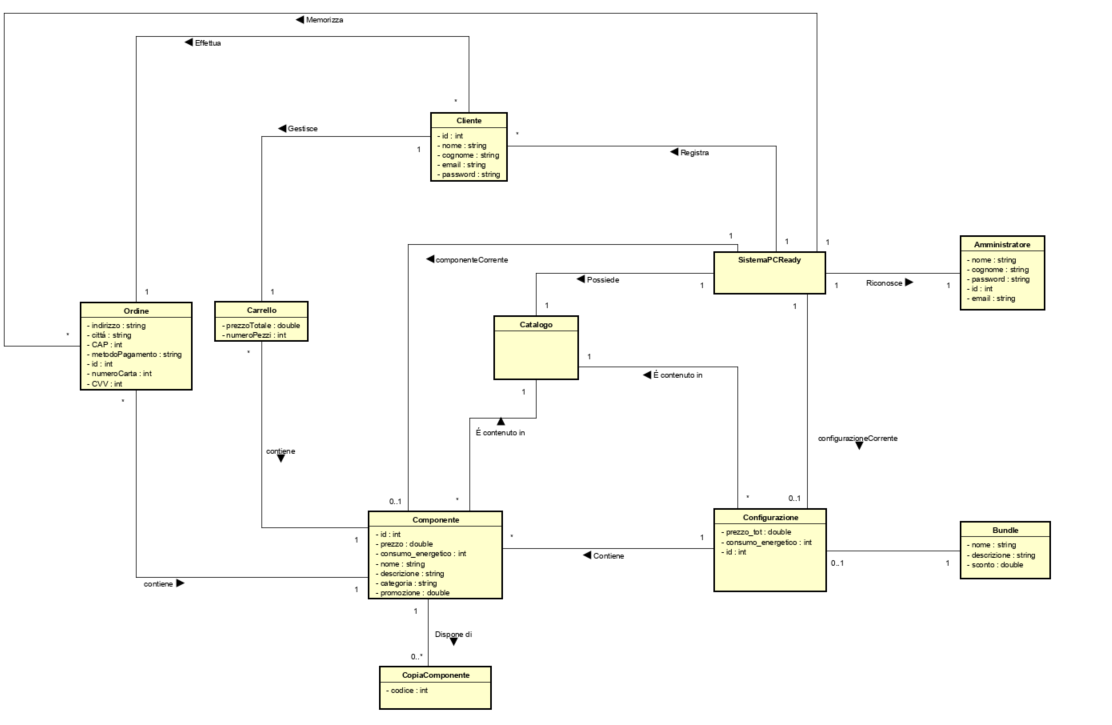
## Modello di dominio

Il modello di dominio è un diagramma che consente di visualizzare tutte le classi concettuali individuate durante l’analisi del problema e lo sviluppo del progetto.

Mediante il suddetto diagramma, che risulta essere un elaborato fondamentale per una buona analisi orientata agli oggetti, mette in evidenzia tutti gli attributi delle classi concettuali, e le relazioni di maggior rilievo che si instaurano tra quest’ultime.

Ovviamente, anche il modello di dominio è stato sviluppato in modo iterativo, per cui di volta in volta sono state applicate ad esso tutte le novitá o le modifiche previste dall’iterazione corrente.

A seguire viene mostrato il modello di domino “finale”, ovvero il modello di dominio che si è ottenuto dopo l’ultima iterazione sviluppata:



Per apprezzare meglio il suddetto diagramma, si consiglia di consultare il file Astah in allegato alla documentazione.

Dunque, mediante tale diagramma sono state individuate le seguenti classi concettuali:

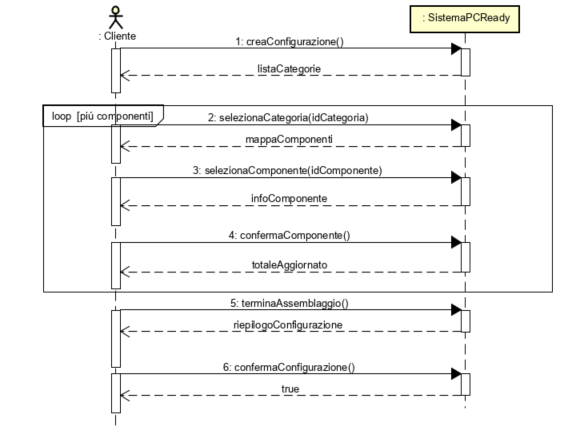
* **SistemaPCReady:** rappresenta il sistema PC Ready;
* **Cliente:** rappresenta un cliente generico, che intende creare configurazioni personalizzate ed acquistarle, o acquistare delle componenti singole o bundle;
* **Amministratore:** rappresenta un amministratore del sistema;
* **Catalogo:** rappresenta il catalogo di componenti disponibili, configurazioni create dai clienti, e bundle pronti per l’acquisto;
* **Carrello:** rappresenta una lista di articoli che il cliente ha selezionato per l’acquisto, difatti il nome di tale classe è un’analogia con un carrello reale;
* **Componente:** rappresenta un componente informatico, come una RAM, una GPU e altro ancora;
* **CopiaComponente:** rappresenta una copia di un componente esistente, attualmente contenuta nel magazzino del negozio;
* **Ordine:** rappresenta un ordine effettuato da un cliente, e contiene tutte le informazioni necessarie per la spedizione dei prodotti acquistati dal cliente;
* **Configurazione:** rappresenta una configurazione assemblata virtualmente da un cliente;
* **Bundle:** rappresenta una configurazione realmente assemblata da un amministratore, alla quale viene assegnato un ulteriore fattore di promozione.

# **SSD e Contratti**

In questo capitolo verranno mostrati tutti i diagrammi di sequenza di sistema SSD e i contratti delle operazioni,(i quali descrivono con un maggiore grado di dettaglio le operazioni presenti nei diagrammi SSD), che sono stati ottenuti durante l’analisi orientata agli oggetti.

Per facilitare la lettura di tali SSD e contratti, si suddivide tale capitolo in casi d’uso.

## UC1



|  |  |
| --- | --- |
| **Nome del contratto** | **CO1: Crea Configurazione** |
| **Operazione** | creaConfigurazione() |
| **Riferimenti** | Caso d’uso UC1: Crea configurazione cliente |
| **Pre-condizioni** | * Il cliente è stato autenticato |
| **Post-condizioni** | * È stata creata un’istanza Conf di Configurazione * Gli attributi di Conf (prezzo, consumo energetico) sono stati inizializzati ai valori di default) * È stata inizializzata una lista di “Componente” denominata “listaComponenti” (attributo che indica i componenti presenti nella configurazione) |

|  |  |
| --- | --- |
| **Nome del contratto** | **CO2: Seleziona Categoria** |
| **Operazione** | selezionaCategoria() |
| **Riferimenti** | Caso d’uso UC1: Crea configurazione cliente |
| **Pre-condizioni** | * È stata inizializzata un’istanza Conf di Configurazione, e una lista “listaComponenti” di Componente * Il cliente ha visualizzato tutte le categorie di Componente |
| **Post-condizioni** | * Viene inizializzata una mappa idComponente-Componente denominata “mappaCorrente” * Il contenuto di mappaCorrente è stato aggiornato in base alla Categoria scelta |

|  |  |
| --- | --- |
| **Nome del contratto** | **CO3: Seleziona Componente** |
| **Operazione** | selezionaComponente() |
| **Riferimenti** | Caso d’uso UC1: Crea configurazione cliente |
| **Pre-condizioni** | * È stata inizializzata una lista di Componente “listaCorrente” |
| **Post-condizioni** | * È stata inizializzata un’istanza componenteCorrente di Componente |

|  |  |
| --- | --- |
| **Nome del contratto** | **CO4: Conferma Componente** |
| **Operazione** | confermaComponente() |
| **Riferimenti** | Caso d’uso UC1: Crea configurazione cliente |
| **Pre-condizioni** | * È stata inizializzata un’istanza Comp di Componente. |
| **Post-condizioni** | * L’istanza Comp viene associata a Conf mediante l’associazione “Contiene” * Gli attributi prezzo e consumo energetico di Conf vengono aggiornati in base alle specifiche di Comp |

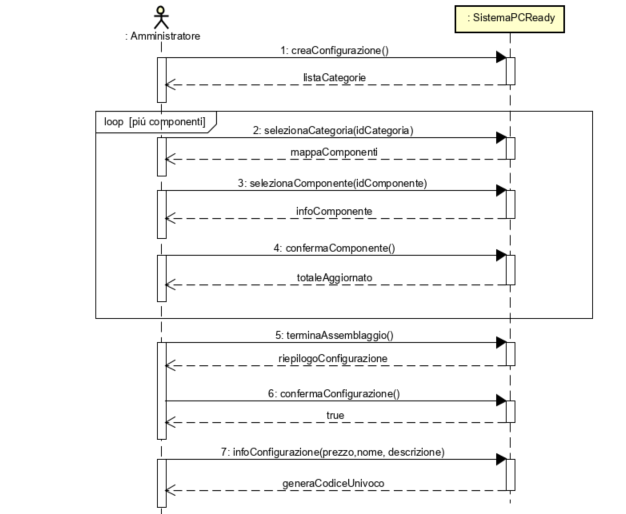
|  |  |
| --- | --- |
| **Nome del contratto** | **CO5: Termina Assemblaggio** |
| **Operazione** | confermaAssemblaggio() |
| **Riferimenti** | Caso d’uso UC1: Crea configurazione cliente |
| **Pre-condizioni** | * Il cliente ha selezionato tutti i componenti per la sua configurazione |
| **Post-condizioni** |  |

|  |  |
| --- | --- |
| **Nome del contratto** | **CO6: Conferma Configurazione** |
| **Operazione** | confermaConfigurazione() |
| **Riferimenti** | Caso d’uso UC1: Crea configurazione cliente |
| **Pre-condizioni** | * Il cliente ha verificato il riepilogo di tutte le scelte che ha effettuato e le specifiche (prezzo e consumo) della configurazione creata |
| **Post-condizioni** | * Il Sistema mantiene in memoria la Configurazione appena creata |

## UC2

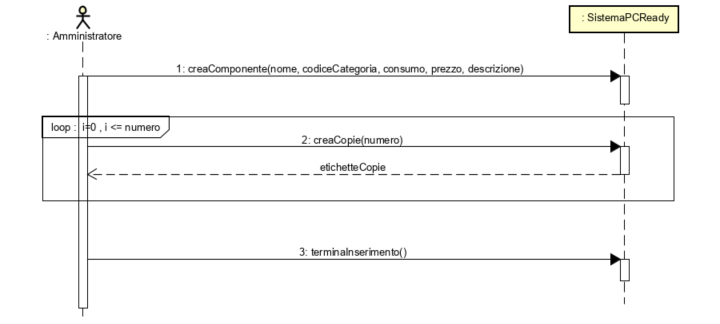
Prima di mostrare SSD e contratti, è d’obbligo sottolineare che molte delle operazioni presenti in tale caso d’uso sono uguali a quello precedente, con l’unica differenza che si sta trattando di un Bundle: dunque, verranno mostrati solo i contratti relativi alle “funzioni aggiuntive”.

Dal punto di vista del codice non vi sono modifiche alle funzioni “condivise” di UC1 per adattarle a UC2 in quanto Bundle risulta essere una estensione della classe Configurazione.



|  |  |
| --- | --- |
| **Nome del contratto** | **CO1: Info Configurazione** |
| **Operazione** | infoConfigurazione() |
| **Riferimenti** | Caso d’uso UC2: Crea Bundle acquistabile |
| **Pre-condizioni** | * È stata inizializzata un’istanza Conf di Configurazione |
| **Post-condizioni** | * È stata inizializzata un’istanza B di Bundle. * Gli attributi di B sono modificati sulla base dei parametri forniti. * L’attributo id di B è stato inizializzato con una sequenza unica. |

## UC3

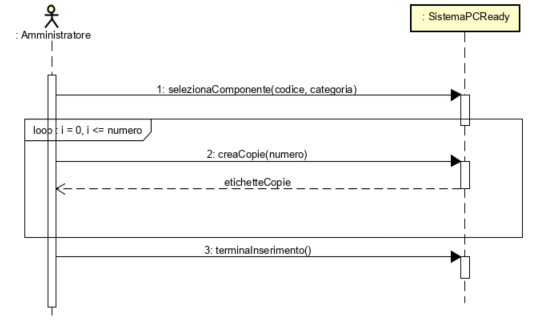


|  |  |
| --- | --- |
| **Nome del contratto** | **CO1: Crea Componente** |
| **Operazione** | creaComponente() |
| **Riferimenti** | Caso d’uso UC3: Inserisci Nuovo Componente |
| **Pre-condizioni** | * L’amministratore è autenticato |
| **Post-condizioni** | * È stata creata un’istanza C di Componente * L’istanza C di Componente è stata associata a Catalogo mediante l’associazione “È contenuto in” * Gli attributi di C sono stati inizializzati con i parametri forniti * È stata inizializzata una lista di CopiaComponente denominata “listaCopie” come attributo di C. |

|  |  |
| --- | --- |
| **Nome del contratto** | **CO2: Crea Copie** |
| **Operazione** | creaCopie() |
| **Riferimenti** | Caso d’uso UC3: Inserisci Nuovo Componente |
| **Pre-condizioni** | * È in corso l’inserimento di un nuovo componente * È stata inizializzata una lista di CopiaComponente detta “listaCopie” |
| **Post-condizioni** | * Sono state create N istanze di Copia, dove N è parametro della funzione * Le N istanze di copia sono associate a C mediante l’associazione “Dispone di” * Un’etichetta è stata stampata per ogni copia * Il valore dell’attributo di C denominato “listaCopie” è stato aggiornato in base alle copie inizializzate |

|  |  |
| --- | --- |
| **Nome del contratto** | **CO3: Termina Inserimento** |
| **Operazione** | terminaInserimento() |
| **Riferimenti** | Caso d’uso UC3: Inserisci Nuovo Componente |
| **Pre-condizioni** | * L’amministratore ha verificato la validità dell’inserimento delle copie del nuovo componente |
| **Post-condizioni** | * Il Sistema mantiene in memoria il Componente e le sue Copie appena generate |

## UC4

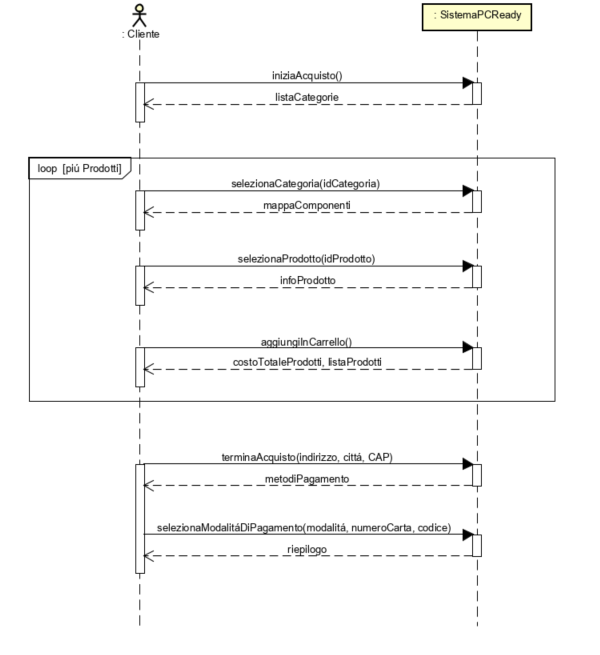


|  |  |
| --- | --- |
| **Nome del contratto** | **CO1: Seleziona Componente** |
| **Operazione** | selezionaComponente() |
| **Riferimenti** | Caso d’uso UC4: Inserisci Copia Componente |
| **Pre-condizioni** | * L’amministratore deve essere autenticato |
| **Post-condizioni** | * Viene recuperata l’istanza C dello specifico Componente cercato |

|  |  |
| --- | --- |
| **Nome del contratto** | **CO2: Crea Copie** |
| **Operazione** | creaCopie() |
| **Riferimenti** | Caso d’uso UC4: Inserisci Copia Componente |
| **Pre-condizioni** | * È stata recuperata un’istanza C di Componente * È presente una lista di CopiaComponente detta “listaCopie” come attributo dell’istanza C |
| **Post-condizioni** | * Sono state create N istanze di Copia, dove N è parametro della funzione * Le N istanze di CopiaComponente sono associate a C mediante l’associazione “Dispone Di” * Un’etichetta è stata stampata per ogni copia * Il valore dell’attributo di C denominato “listaCopie” è stato aggiornato in base alle copie inizializzate |

|  |  |
| --- | --- |
| **Nome del contratto** | **CO3: Termina Inserimento** |
| **Operazione** | terminaInserimento() |
| **Riferimenti** | Caso d’uso UC4: Inserisci Copia Componente |
| **Pre-condizioni** | * L’amministratore ha verificato la validità dell’inserimento delle copie del nuovo componente |
| **Post-condizioni** | * Il Sistema mantiene in memoria le N istanze di CopiaComponente appena generate |

## UC5



|  |  |
| --- | --- |
| **Nome del contratto** | **CO1: iniziaAcquisto** |
| **Operazione** | iniziaAcquisto() |
| **Riferimenti** | Caso d’uso UC5: Effettua Acquisto |
| **Pre-condizioni** | * Il cliente deve essere autenticato |
| **Post-condizioni** | * È stata inizializzata un’istanza di Carrello denominata “carrello” con i valori di default * “carrello” è stato associato ad Utente mediante la relazione “gestisce” * È stata inizializzata una lista “listaComponentiCarrello” di Componente, in quanto attributo di carrello |

|  |  |
| --- | --- |
| **Nome del contratto** | **CO2: selezionaCategoria** |
| **Operazione** | selezionaCategoria() |
| **Riferimenti** | Caso d’uso UC5: Effettua Acquisto |
| **Pre-condizioni** | * Un’istanza di Carrello denominata “carrello” deve essere stata inizializzata * Il cliente ha visualizzato tutte le categorie di Componente |
| **Post-condizioni** | * Viene inizializzata una mappa idComponente-Componente denominata “mappaCorrente” * Il contenuto di mappaCorrente è stato aggiornato in base alla Categoria scelta |

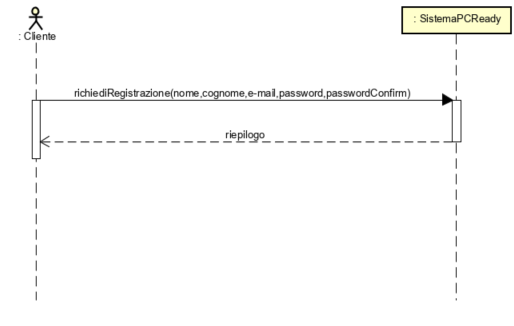
|  |  |
| --- | --- |
| **Nome del contratto** | **CO3: selezionaProdotto** |
| **Operazione** | selezionaProdotto() |
| **Riferimenti** | Caso d’uso UC5: Effettua Acquisto |
| **Pre-condizioni** | * È stata inizializzata una mappa idComponente-Componente denominata “mappaCorrente” * Un’istanza di Carrello denominata “carrello” è stata inizializzata |
| **Post-condizioni** | * È stata inizializzata un’istanza “componenteCorrente” di Componente |

|  |  |
| --- | --- |
| **Nome del contratto** | **CO4: aggiungiInCarrello** |
| **Operazione** | aggiungiInCarrello() |
| **Riferimenti** | Caso d’uso UC5: Effettua Acquisto |
| **Pre-condizioni** | * È stata inizializzata un’istanza “componenteCorrente” di Componente |
| **Post-condizioni** | * L’attributo di “carrello” denominato “listaComponentiCarrello” è stato aggiornato con i valori dell’istanza “componente corrente”, mediante la relazione “contiene” tra Carrello e Componente * I restanti attributi di “carrello” sono stati aggiornati |

|  |  |
| --- | --- |
| **Nome del contratto** | **CO5: terminaAcquisto** |
| **Operazione** | terminaAcquisto() |
| **Riferimenti** | Caso d’uso UC5: Effettua Acquisto |
| **Pre-condizioni** | * Il cliente ha selezionato tutti i componenti da acquistare |
| **Post-condizioni** | * È stata inizializzata un’istanza di Ordine denominata “ordineCorrente” * L’istanza “ordineCorrente” è stata associata a Cliente mediante la relazione “effettua” * Gli attributi di “ordineCorrente” sono stati modificati mediante i valori forniti dall’utente. * L’attributo di “ordine” denominato “listaComponentiOrdine” è stato aggiornato con i valori di “listaComponentiCarrello”. |

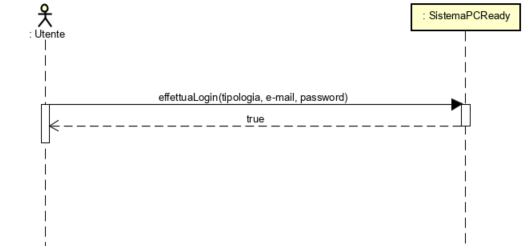
|  |  |
| --- | --- |
| **Nome del contratto** | **CO6: selezionaModalitáDiPagamento** |
| **Operazione** | selezionaModalitáDiPagamento() |
| **Riferimenti** | Caso d’uso UC5: Effettua Acquisto |
| **Pre-condizioni** | * Il cliente ha visualizzato i possibili metodi di pagamento |
| **Post-condizioni** | * Gli attributi di “ordineCorrente” sono stati modificati mediante i valori forniti dall’utente. * Il Sistema mantiene in memoria l’istanza “ordineCorrente” |

## UC6



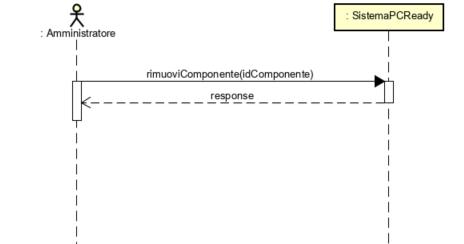
|  |  |
| --- | --- |
| **Nome del contratto** | **CO2: richiediRegistrazione** |
| **Operazione** | richiediRegistrazione() |
| **Riferimenti** | Caso d’uso UC6: Registrazione cliente |
| **Pre-condizioni** |  |
| **Post-condizioni** | * Viene inizializzata un’istanza di Cliente detta “clienteR” mediante l’associazione di Cliente con SistemaPCReady chiamata “registra” * Gli attributi di clienteR sono stati aggiornati mediante i dati forniti dall’utente * Il Sistema mantiene in memoria l’istanza “clienteR” |

## UC7



|  |  |
| --- | --- |
| **Nome del contratto** | **CO1: effettuaLogin** |
| **Operazione** | effettuaLogin() |
| **Riferimenti** | Caso d’uso UC7: Login |
| **Pre-condizioni** | * Esiste in memoria un’istanza di Utente o Amministratore che sta richiedendo il login |
| **Post-condizioni** | * È stata inizializzata un’istanza di Cliente “clienteCorrente” o di Amministratore “amministratoreCorrente”, in base alla corrispondenza trovata con i valori forniti. |

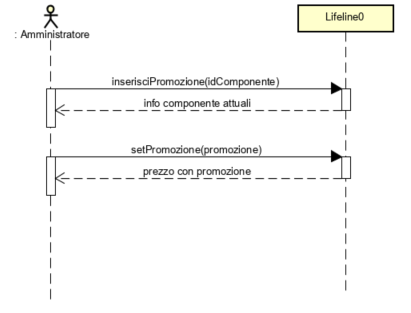
## UC8



|  |  |
| --- | --- |
| **Nome del contratto** | **CO1: rimuoviComponente** |
| **Operazione** | rimuoviComponente() |
| **Riferimenti** | Caso d’uso UC8: Rimozione componente |
| **Pre-condizioni** | * È stata inizializzata un’istanza “amministratoreCorrente” di classe “Amministratore”, ovvero un amministratore deve essere stato autenticato |
| **Post-condizioni** | * È stata eliminata un’istanza di Componente in base alla corrispondenza con l’identificativo fornito * È stata aggiornato l’attributo “mappaComponenti” della classe Catalogo |

|  |  |
| --- | --- |
| **Nome del contratto** | **CO1: rimuoviConfigurazioni** |
| **Operazione** | rimuoviConfigurazioni() |
| **Riferimenti** | Caso d’uso UC8: Rimozione componente |
| **Pre-condizioni** | * È stata eliminata un’istanza di Componente di dato identificativo |
| **Post-condizioni** | * È stata eliminata un’istanza di Configurazione, il cui attributo “listaComponenti” conteneva una corrispondenza con l’istanza di Componente eliminata in precedenza * È stata eliminata un’istanza di Bundle, il cui attributo “listaComponenti” conteneva una corrispondenza con l’istanza di Componente eliminata in precedenza * È stata aggiornato l’attributo “mappaComponenti” della classe Catalogo |

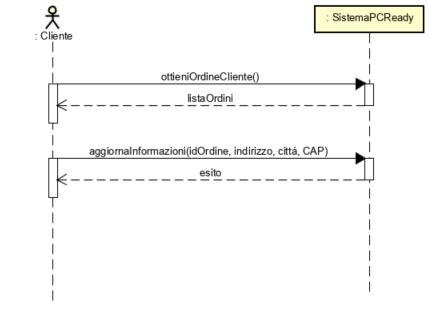
## UC9



|  |  |
| --- | --- |
| **Nome del contratto** | **CO1: selezionaComponente** |
| **Operazione** | selezionaComponente() |
| **Riferimenti** | Caso d’uso UC9: Crea promozione |
| **Pre-condizioni** | * È stata inizializzata un’istanza “amministratoreCorrente” di classe “Amministratore”, ovvero un amministratore deve essere stato autenticato |
| **Post-condizioni** | * È stata recuperata un’istanza di Componente “comp” in base alla corrispondenza con l’identificativo fornito |

|  |  |
| --- | --- |
| **Nome del contratto** | **CO2: setPromozione** |
| **Operazione** | setPromozione() |
| **Riferimenti** | Caso d’uso UC9: Crea promozione |
| **Pre-condizioni** | * È stata recuperata un’istanza “comp” di Componente |
| **Post-condizioni** | * È stato aggiornato l’attributo “promozione” di “comp” in base ai dati forniti dall’amministratore |

## UC10



|  |  |
| --- | --- |
| **Nome del contratto** | **CO1: ottieniOrdineCliente** |
| **Operazione** | ottieniOrdineCliente() |
| **Riferimenti** | Caso d’uso UC10: Modifica ordine |
| **Pre-condizioni** | * È stata inizializzata un’istanza “clienteCorrente” di classe “Cliente”, ovvero un cliente deve essere stato autenticato |
| **Post-condizioni** | * È stata inizializzata una lista di “Ordine” detta “listaOrdini” in base all’attributo id di clienteCorrente |

|  |  |
| --- | --- |
| **Nome del contratto** | **CO2: aggiornaInformazioni** |
| **Operazione** | aggiornaInformazioni() |
| **Riferimenti** | Caso d’uso UC10: Modifica ordine |
| **Pre-condizioni** | * È stata inizializzata una lista di “Ordine” detta “listaOrdini” |
| **Post-condizioni** | * Gli attributi di un’istanza di ordine (individuata in base ai dati forniti dal cliente) sono stati modificati sulla base dei dati forniti, mediante l’associazione “Memorizza” presente tra le classi “SistemaPCReady” e “Ordine” |

# **Progettazione**

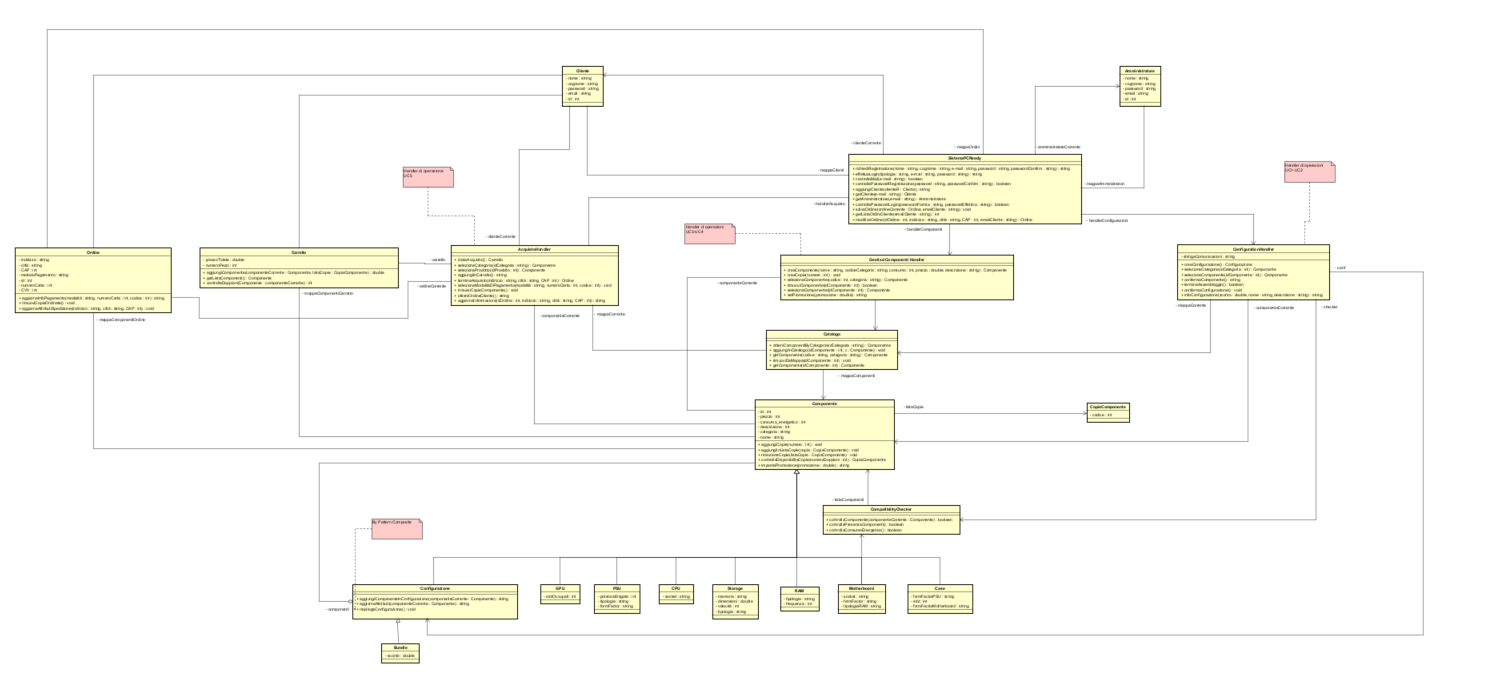
La progettazione orientata agli oggetti consente di definire e caratterizzare degli oggetti software, enfatizzando come questi collaborano per soddisfare tutti i requisiti individuati.

La progettazione è di fondamentale importanza, in quanto tutte le classi scelte vanno poi implementate durante la fase di programmazione orientata agli oggetti.

In tale capitolo verranno mostrati due elementi fondamentali per la progettazione:

* Diagramma delle classi di progetto: permette di descrivere il tipo degli oggetti che fanno parte di un sistema, marcando le tipologie di relazioni che vi sono tra di essi. [Punto di vista statico]
* Diagrammi di sequenza: mostrano l’interazione tra un insieme di oggetti documentando il comportamento dettagliato di un singolo scenario mediante un “formato a steccato”. [Punto di vista dinamico]

## Diagramma delle classi di progetto



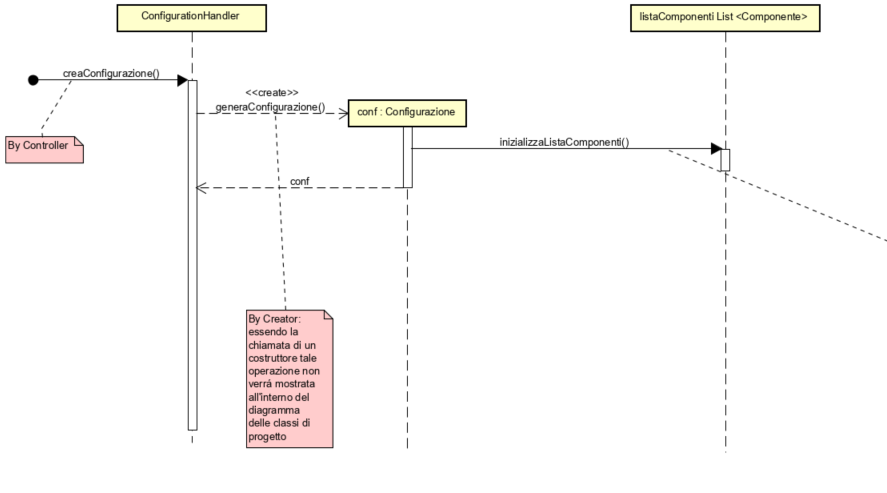
Si invita a visionare il file Astah in allegato alla documentazione per meglio apprezzare i dettagli di tale diagramma.

## Diagrammi di sequenza di sistema

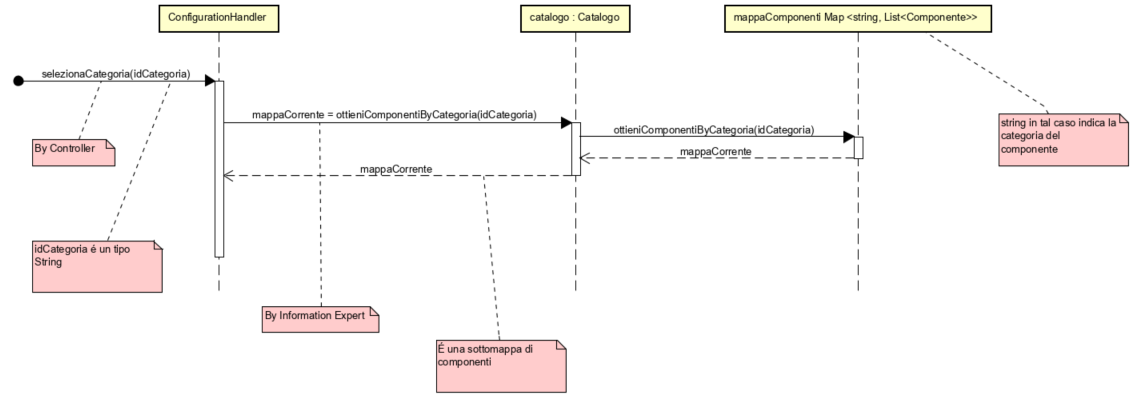
A seguito verranno mostrati i diagrammi di sequenza di sistema, raggruppati per casi d’uso: inoltre ciascun diagramma di sequenza di sistema verrà accompagnato da una breve descrizione

### SSD - UC1

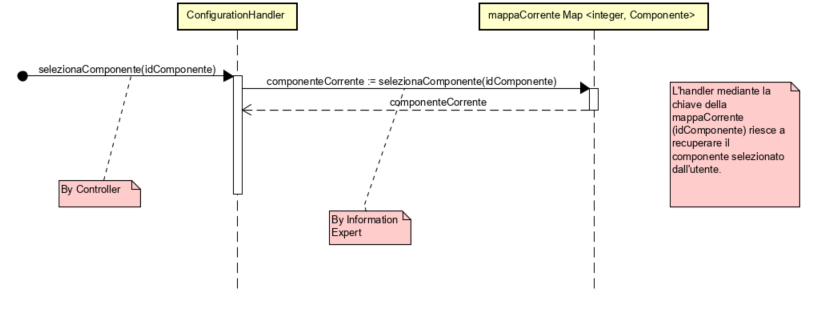
* **creaConfigurazione:** questa operazione si occupa di creare una configurazione, inizializzandone una listaComponenti, e impostando gli altri attributi ai valori di default.



* **selezionaCategoria:** tale funzione si occupa di prendere in ingresso la categoria selezionata da un utente, restituendo tutte le componenti contenute in tale categoria mediante una mappa con chiave intera (id del componente) e valore componente.

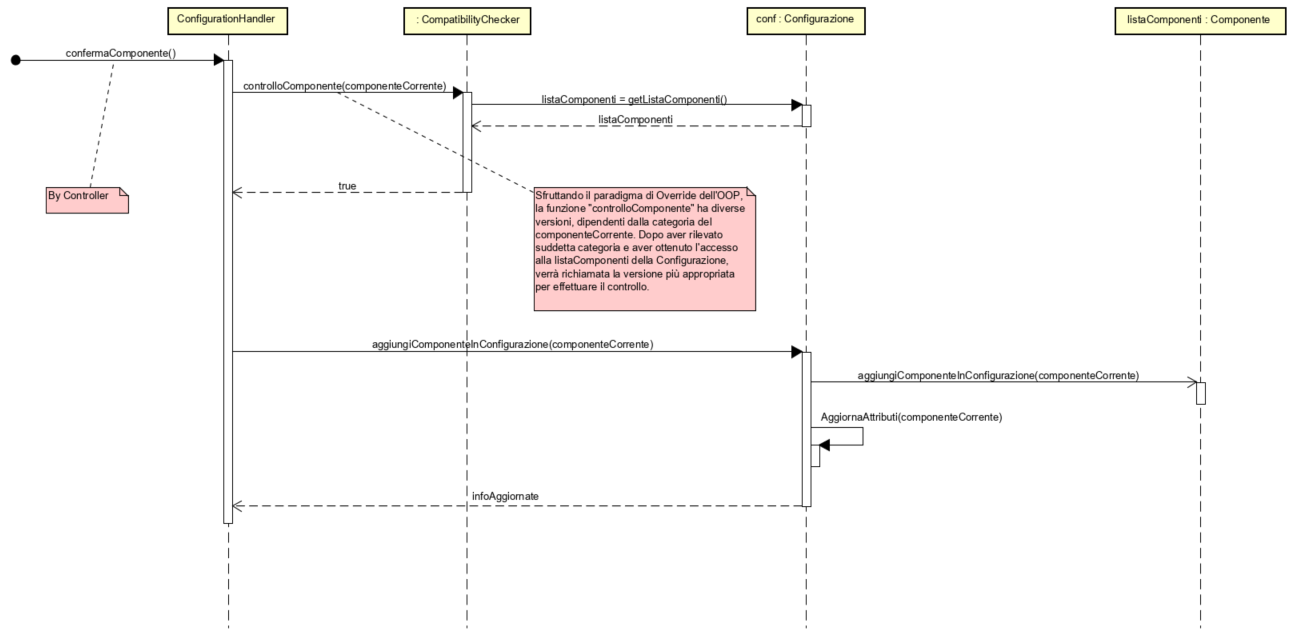


* **selezionaComponente:** tale funzione si occupa di restituire i dettagli del componente selezionato da un utente



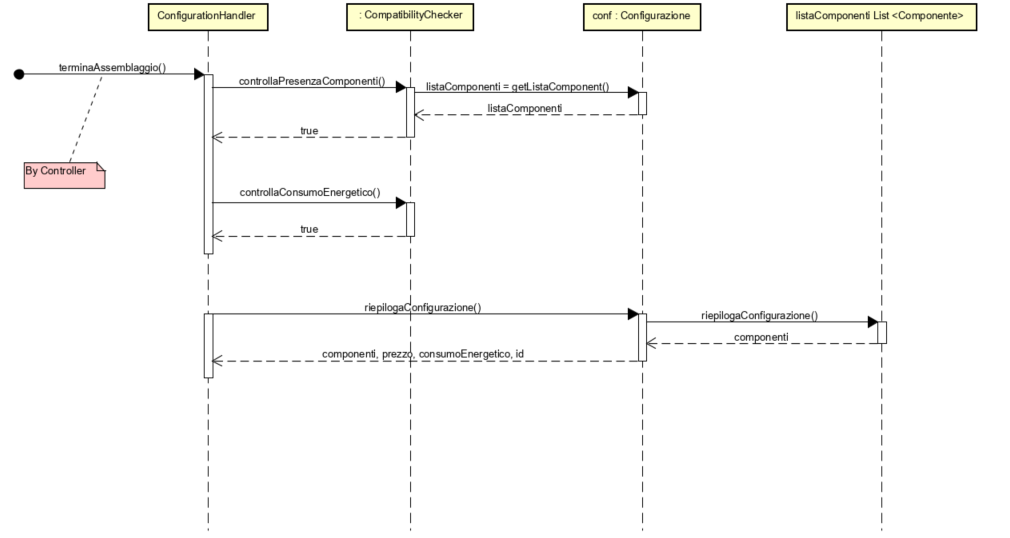
* **confermaComponente:** tale funzione permette di confermare il componente visionato in precedenza, e quindi di inserire quest’ultimo nella configurazione, se e soltanto se tale componente risulta essere compatibile con tutti i componenti già presenti nella “lista componenti” della configurazione.

Si osserva che la responsabilità dei controlli è stata affidata ad una classe apposita chiamata “CompatibilityChecker”.

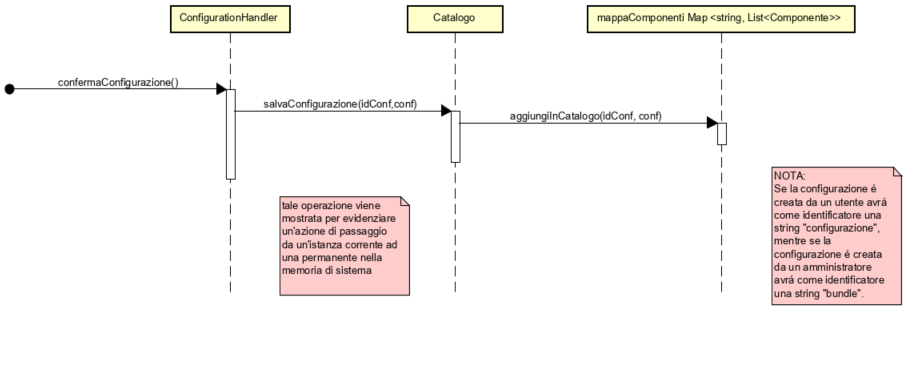


* **terminaAssemblaggio:** tale funzione consente di terminare l’assemblaggio della configurazione ed eseguire i controlli per il corretto funzionamento di quest’ultima.

Se i controlli vanno a buon fine verrà restituito un riepilogo della configurazione appena assemblata.

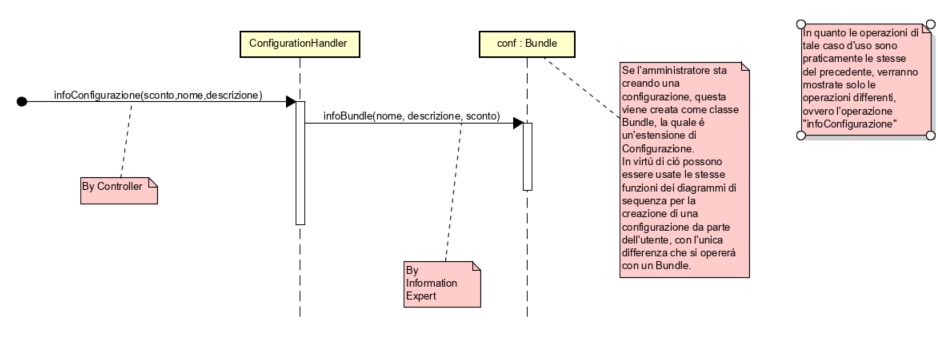


* **confermaConfigurazione:** tale funzione ha il compito di salvare nella memoria del sistema la configurazione appena creata.



### SSD – UC2

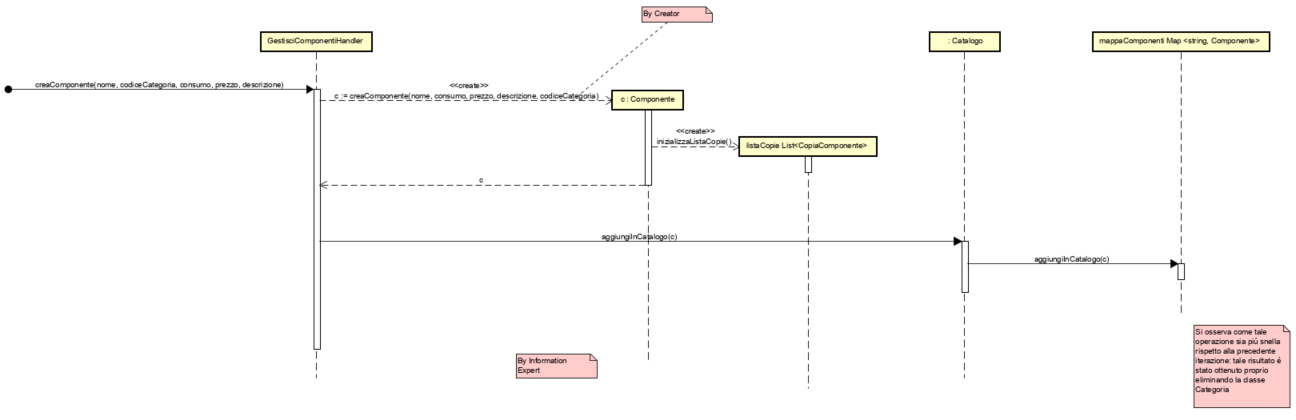
* **infoConfigurazione:** la funzione in questione ha il compito di impostare il nome, lo sconto e la descrizione del bundle che un amministratore ha assemblato.



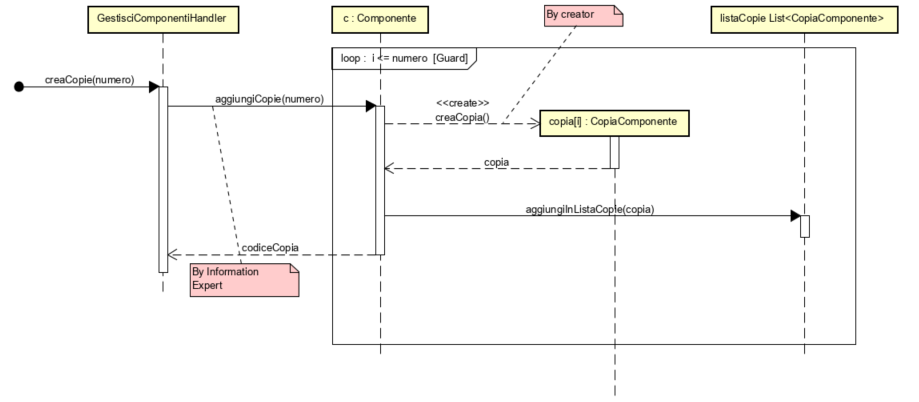
### SSD – UC3

* **creaComponente:** tale funzione ha il compito di creare un nuovo componente, prendendo in ingresso tutte le informazioni necessarie alla creazione.

Il componente creato verrá quindi aggiunto al catalogo.

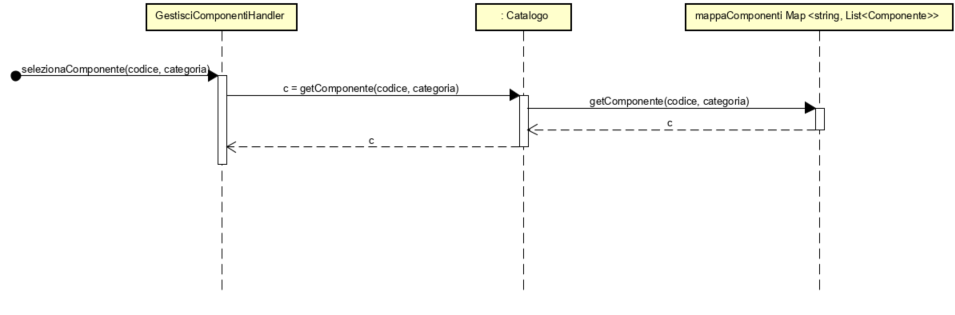


* **creaCopie:** la suddetta funzione ha il compito di inserire un determinato numero di copie per il componente [componenteCorrente] inserito mediante la funzione “creaComponente”.



### SSD – UC4

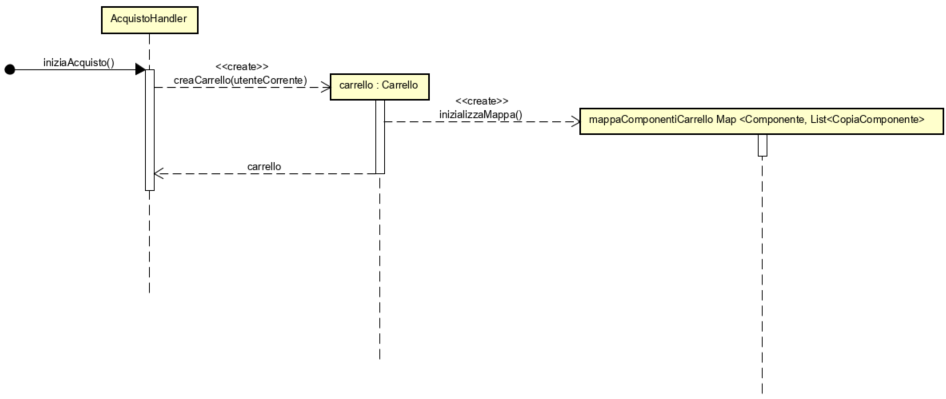
* **selezionaComponente:** tale funzione ha il compito di recuperare un componente già esistente in catalogo, per poter successivamente aggiungere delle copie di quest’ultimo mediante la funzione “creaCopie”.



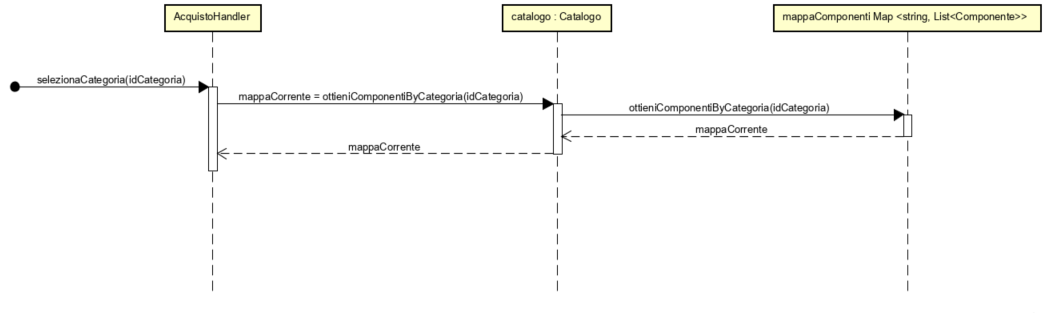
### SSD – UC5

* **iniziaAcquisto:** tale funzione ha il compito di inizializzare un “carrello” per l’utente che ha intenzione di effettuare un acquisto.

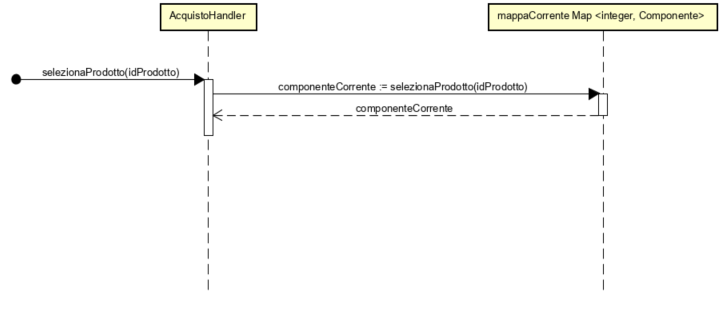
Il carrello in questione conterrà tutti i prodotti che l’utente ha intenzione di acquistare.



* **selezionaCategoria:** mediante tale funzione Viene selezionata una categoria dal cliente, e in base alla categoria scelta viene ritornata una mappa di id-componente/componente

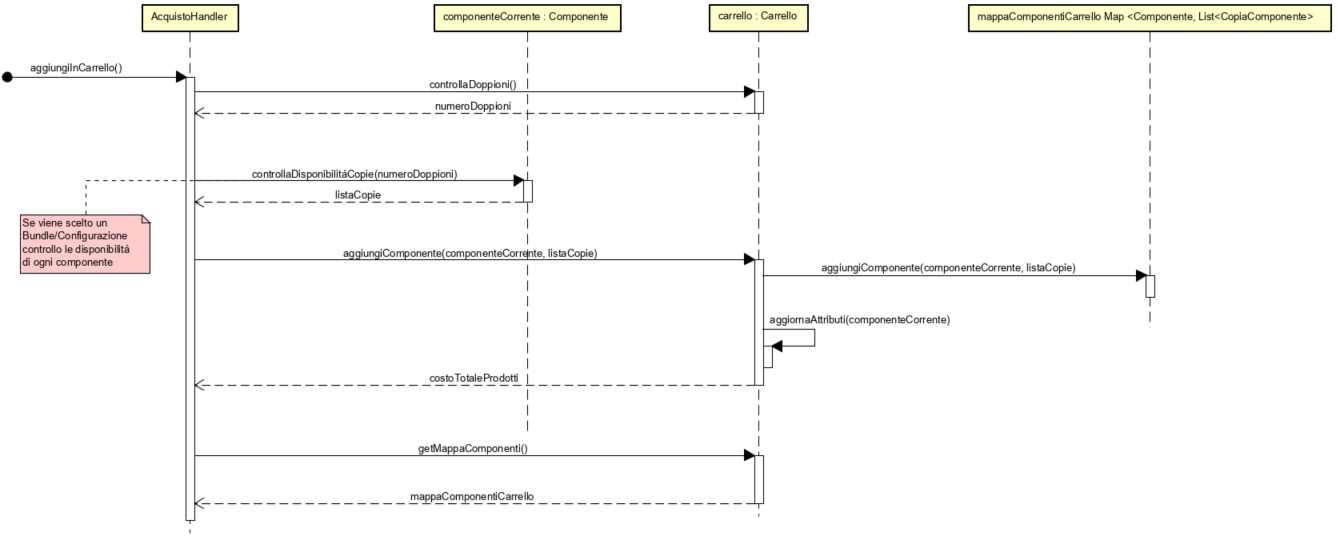


* **selezionaProdotto:** per mezzo di tale funzione viene selezionato dalla mappa ottenuta in precedenza (funzione selezionaCategoria) un prodotto, mediante l’id fornito dal cliente.

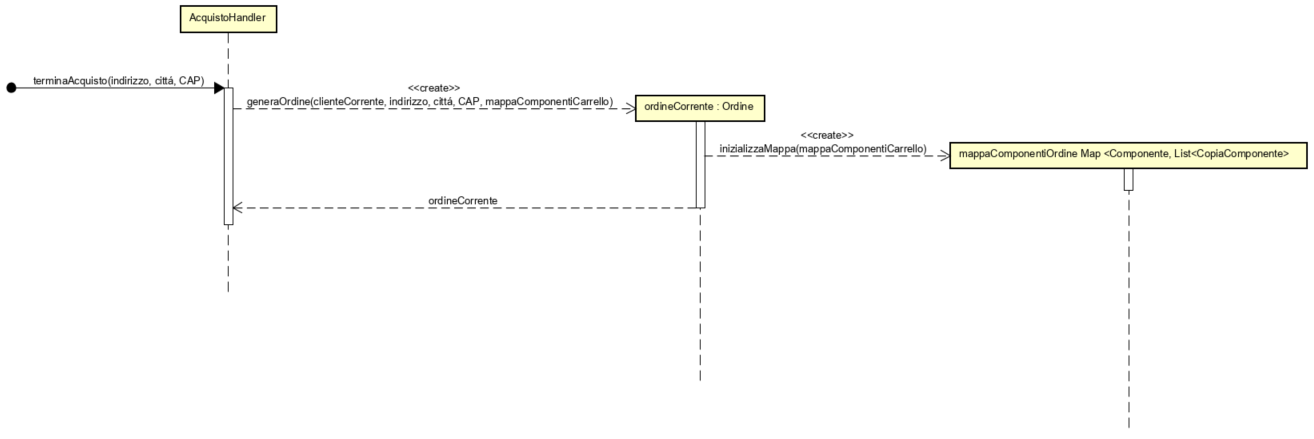


* **aggiungiInCarrello:** mediante tale funzione il prodotto selezionato in precedenza dal cliente viene aggiunto al carrello, dopo un controllo sulla disponibilità delle copie del componente

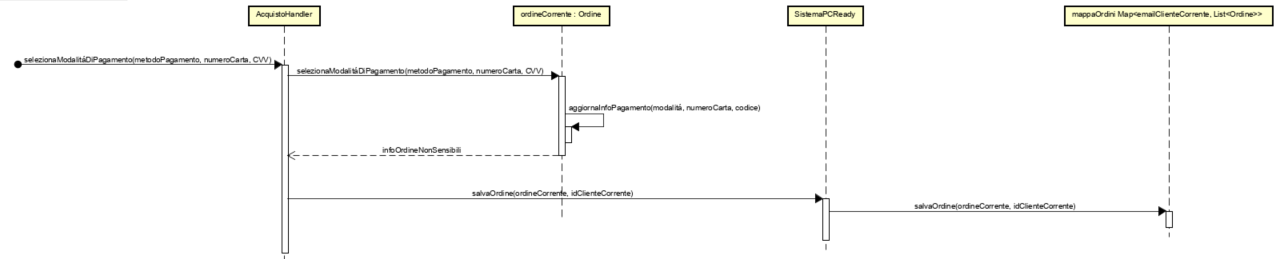
Per osservare in dettaglio la logica di controllo si consulti il documento “Discussione Iterazione 3”.



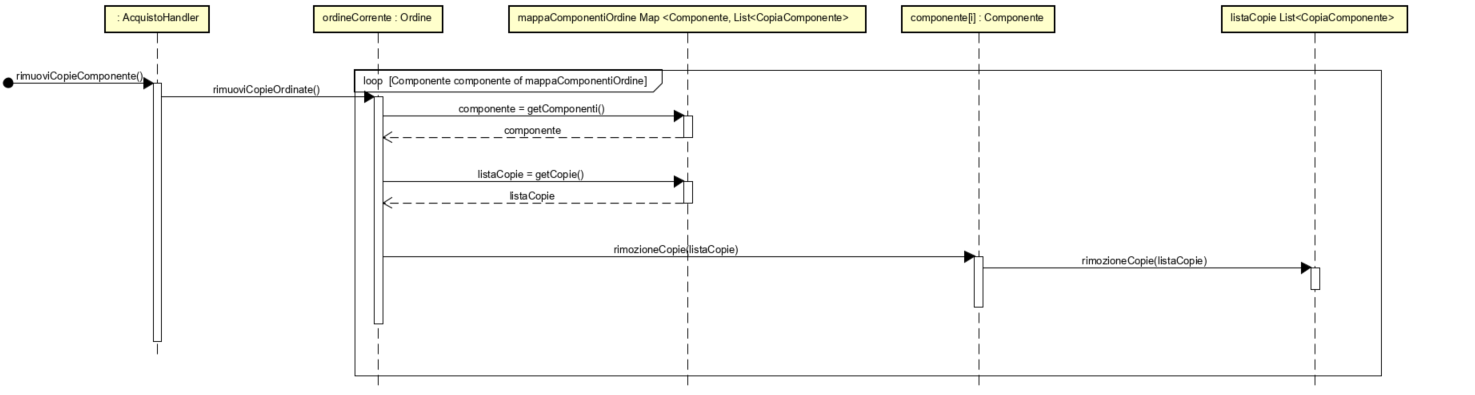
* **terminaAcquisto:** tale funzione consente di inizializzare un’istanza di ordine, la quale possiede una mappa di componenti che l’utente sta acquistando, associata ad un insieme di attributi relativi al domicilio del cliente.



* **selezionaModialitáDiPagamento**

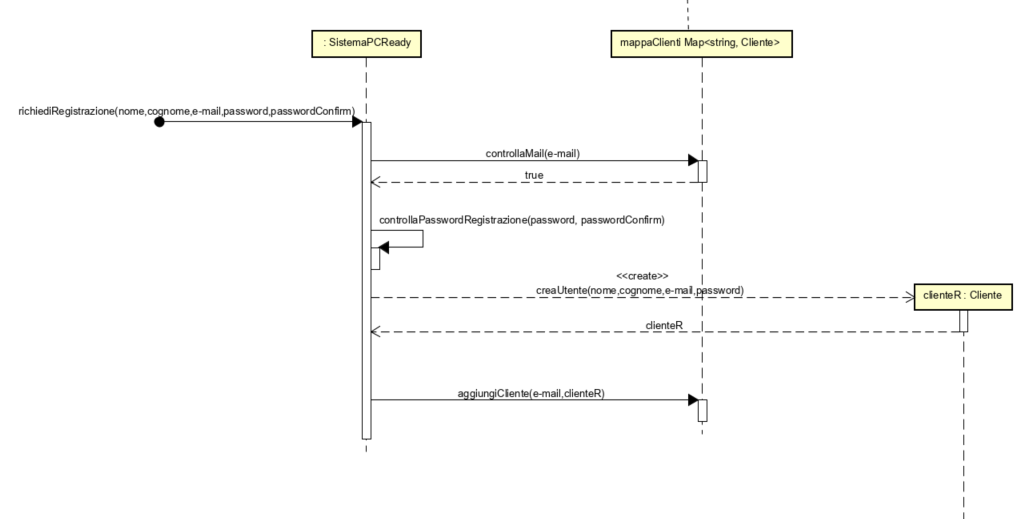


* **rimuoviCopieComponente**



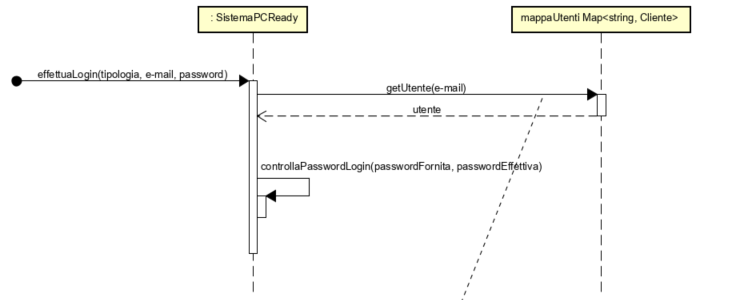
### SSD – UC6

* **richiediRegistrazione:**

****

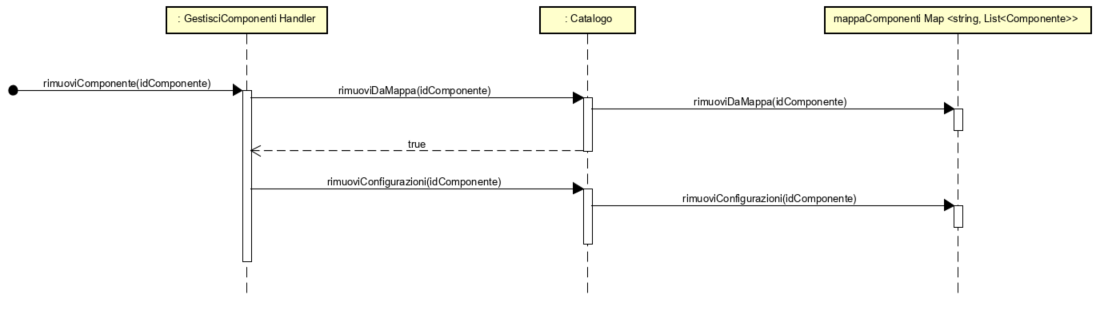
### SSD – UC7

* **effettuaLogin:**



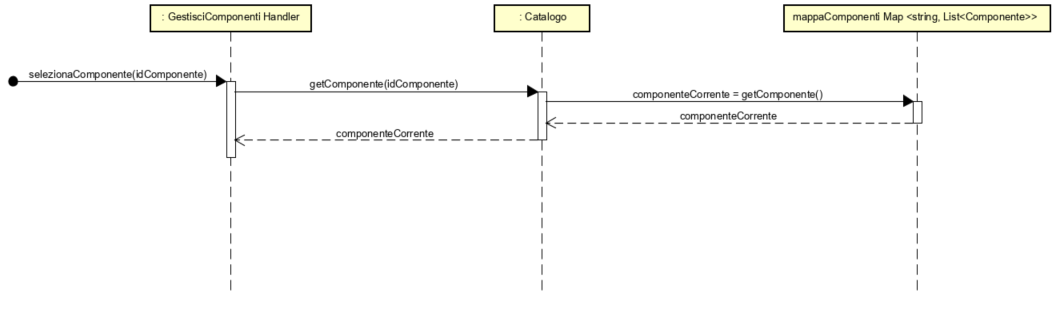
### SSD – UC8

* **rimuoviComponente**

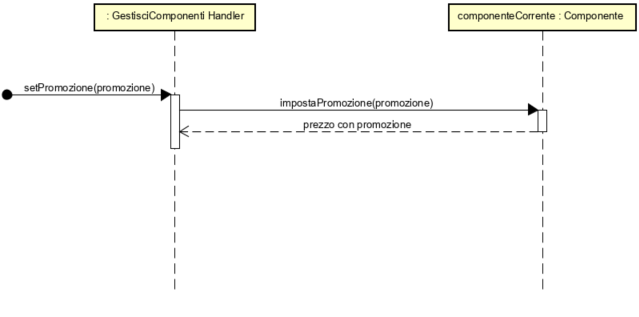


### SSD – UC9

* **selezionaComponente**

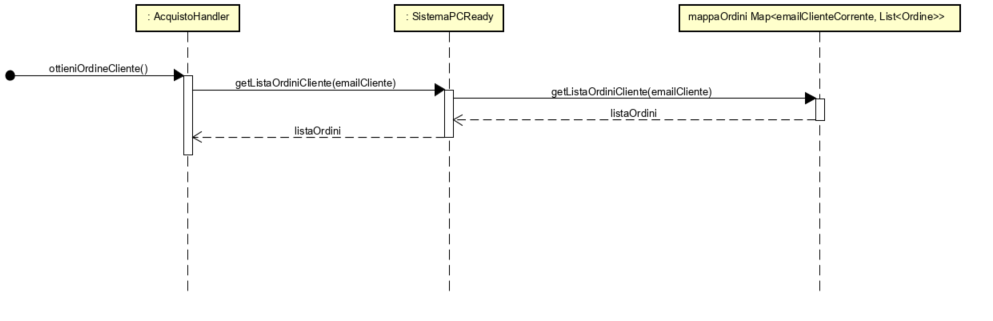


* **setPromozione**



### SSD - UC10

* **ottieniOrdineCliente**



* **aggiornaInformazioni**

