

Università degli Studi di Napoli Federico II

**Corso di Laurea in Ingegneria Informatica**

**Corso di Ingegneria del Software**

**Prof. R. Pietrantuono- A.A. 2023 – 24**

***Progetto***

NOLEGGIO ARTICOLI

SPORTIVI INVERNALI

Studenti: Giuseppe Maglione N46006085

Manuel Parasmo N46005706

Simone Imperatore M46005458

Salvatore Marra N46005945

Versione ?? del *04/06/2024*

**Indice**

1. Specifiche informali 1

2. Analisi e specifica dei requisiti 2

2.1 Revisione dei requisiti 2

2.2 Glossario dei termini 2

2.3 Classificazione dei requisiti 3

2.3.1 Requisiti funzionali 3

2.3.2 Requisiti sui dati 3

2.3.3 Vincoli / Altri requisiti 4

2.4 Modellazione dei casi d’uso 4

2.4.1 Attori e casi d’uso 4

2.4.2 Diagramma dei casi d’uso 2

2.4.3 Scenari 2

2.5 Diagramma delle classi 2

2.6 Diagrammi di sequenza 2

3. Stima dei costi 3

4. Piano di test funzionale 4

5. Progettazione 6

5.1 Diagramma delle classi 6

5.2 Diagrammi di sequenza 6

6. Implementazione 7

7. Testing 8

7.1 Test funzionale 8

# Specifiche informali

*Si vuole realizzare un sistema informativo per un negozio che noleggia attrezzature sportive invernali. Con il sistema interagiscono i dipendenti del negozio e i clienti che vogliono noleggiare attrezzature.*

*Il negozio noleggia sci con bastoncini, snowboard, caschi e scarponi. Gli sci sono caratterizzati dalla lunghezza e sono suddivisi in sci di base, medi e avanzati a seconda del materiale di fabbricazione e della forma. I bastoncini sono corti, medi o lunghi. Gli snowboard sono di un unico tipo. I caschi sono di dimensione M, L o XL. Infine, gli scarponi hanno una misura (alla stregua delle scarpe, es. 44). Il costo giornaliero di noleggio cambia secondo la tipologia di attrezzature (è uguale per tutti gli sci, per tutti i caschi, etc.)*

*L’attrezzatura può essere: disponibile, se è presente nel magazzino; noleggiata, se è stata noleggiata da un cliente; o guasta, se è presente nel magazzino ma non può essere utilizzata a causa di un danno che la rende inutilizzabile.*

*I clienti prenotano tramite il sistema le attrezzature da noleggiare per il giorno dopo: inseriscono nome, cognome, e-mail, residenza, ed estremi di un documento, selezionano le attrezzature. I magazzinieri del negozio consultano le prenotazioni e preparano le attrezzature per il giorno dopo; quando tutte le attrezzature di una prenotazione sono disponibili, il magazziniere conferma l’accettazione della prenotazione, e il sistema invia al cliente un messaggio di conferma; il cliente effettua il pagamento, e riceve un codice di prenotazione. Se una prenotazione non può essere soddisfatta, perché una attrezzatura non è disponibile, il magazziniere non accetta la prenotazione, e il sistema invia un messaggio al cliente. Per il ritiro e la riconsegna, il cliente deve comunicare il nr di prenotazione al banco del negozio, e il dipendente registra la consegna/restituzione.*

*A fine giornata, il sistema produce un report dei i noleggi che terminano in quella giornata che ancora devono essere consegnati.*

# Analisi e specifica dei requisiti

## Revisione dei requisiti

*1. Ciascuno scii ha una lunghezza e una categoria (base, medio, avanzato) in base a materiale e forma.*

*2. I bastoncini hanno una dimensione (corti, medi, lunghi).*

*3. Gli snowboard sono di unico tipo.*

*4. I caschi hanno una taglia (M, L, XL).*

*5. Gli scarponi hanno una misura.*

*6. Il costo giornaliero di noleggio è fissato per ogni categoria di attrezzatura.*

*7. Un'attrezzatura può essere disponibile, noleggiata o guasta.*

*8. Il sistema deve permettere al cliente di effettuare una richiesta di prenotazione per le attrezzature da noleggiare il giorno seguente.*

*9. Per ciascuna richiesta di prenotazione il cliente deve specificare: nome, cognome, email, residenza, estremi di un documento e telefono.*

*10. Il sistema deve permettere al dipendente di selezionare le attrezzature che intende richiedere.*

*11. Il sistema deve permettere al dipendente (magazziniere) di consultare le prenotazioni.*

*12. Il sistema deve permettere al dipendente (magazziniere) di confermare o rifiutare una richiesta di prenotazione.*

*13. Il sistema deve inviare al cliente un messaggio tramite mail contenente l’esito stabilito della richiesta di prenotazione.*

*14. Il sistema deve permettere al cliente di effettuare il pagamento a seguito della recezione della mail di conferma.*

*15. Il sistema, a seguito della conclusione del pagamento, deve inviare al cliente un messaggio tramite mail di riepilogo prenotazione.*

*16. Il riepilogo prenotazione contiene: numero di prenotazione, e articoli noleggiati.*

*17. Per poter ritirare o riconsegnare le attrezzature noleggiata il cliente deve comunicare il numero di prenotazione fornito.*

*18. Il sistema deve permettere al dipendente (banco) di registrare le consegne e le restituzioni in base al numero di prenotazione fornito dal cliente.*

*19. Il sistema deve produrre ogni giorno un report dei noleggi in scadenza per il giorno stesso ma che non sono stati riconsegnati.*

*20. Il report deve contenere per ogni prenotazione presente: numero di prenotazione, nome, cognome, email e telefono del cliente che ha effettuato questa ultima.*

*21. Per lo scambio di messaggi tra il sistema e il cliente deve essere disponibile un server di posta elettronica esterno al sistema.*

*22. Il report giornaliero deve essere generato ogni giorno all’orario di chiusura del magazzino.*

## Glossario dei termini

|  |  |  |
| --- | --- | --- |
| Termine | Descrizione | Sinonimi |
| Articolo | Una delle attrezzature sportive presenti in magazzino. | Attrezzatura |
| Report | Recap giornaliero dell’andamento dei noleggi non restituiti in scadenza il giorno stesso. |  |
| Esito | Risultato (conferma o rifiuto) della richiesta di prenotazione inviata dall’utente. |  |
| Dipendente | Uno dei lavoratori in servizio nel magazzino. | Impiegato |
| Riepilogo | Riassunto della prenotazione effettuata dall’utente. |  |

## Classificazione dei requisiti

## Requisiti funzionali

|  |  |  |
| --- | --- | --- |
| ID | Requisito | Origine (n. frase dei requisiti revisionati) |
| RF01 | *Il sistema deve permettere al cliente di effettuare una richiesta di prenotazione per le attrezzature da noleggiare il giorno seguente.* | 8 |
| RF02 | Il sistema deve permettere al dipendente di selezionare le attrezzature che intende richiedere. | 10 |
| RF03 | Il sistema deve permettere al dipendente (magazziniere) di consultare le prenotazioni. | 11 |
| RF04 | Il sistema deve permettere al dipendente (magazziniere) di confermare o rifiutare una richiesta di prenotazione. | 12 |
| RF05 | Il sistema deve inviare al cliente un messaggio tramite mail contenente l’esito stabilito della richiesta di prenotazione. | 13 |
| RF06 | Il sistema deve permettere al cliente di effettuare il pagamento a seguito della recezione della mail di conferma. | 14 |
| RF07 | Il sistema, a seguito della conclusione del pagamento, deve inviare al cliente un messaggio tramite mail di riepilogo prenotazione. | 15 |
| RF08 | Il sistema deve permettere al dipendente (banco) di registrare le consegne e le restituzioni in base al numero di prenotazione fornito dal cliente. | 18 |
| RF9 | Il sistema deve produrre ogni giorno un report dei noleggi in scadenza per il giorno stesso ma che non sono stati riconsegnati. | 19 |

## Requisiti sui dati

|  |  |  |
| --- | --- | --- |
| ID | Requisito | Origine (n. frase dei requisiti revisionati) |
| RD01 | Ciascuno scii ha una lunghezza e una categoria (base, medio, avanzato) in base a materiale e forma. | 1 |
| RD02 | I bastoncini hanno una dimensione (corti, medi, lunghi). | 2 |
| RD03 | Gli snowboard sono di unico tipo. | 3 |
| RD04 | I caschi hanno una taglia (M, L, XL). | 4 |
| RD05 | Gli scarponi hanno una misura. | 5 |
| RD06 | Il costo giornaliero di noleggio è fissato per ogni categoria di attrezzatura. | 6 |
| RD07 | Un'attrezzatura può essere disponibile, noleggiata o guasta. | 7 |
| RD08 | Per ciascuna richiesta di prenotazione il cliente deve specificare: nome, cognome, email, residenza, estremi di un documento e telefono. | 9 |
| RD09 | Il riepilogo prenotazione contiene: numero di prenotazione, e articoli noleggiati. | 16 |
| RD10 | Il report deve contenere per ogni prenotazione presente: numero di prenotazione, nome, cognome, email e telefono del cliente che ha effettuato questa ultima. | 20 |

## Vincoli / Altri requisiti

RNF01: Per lo scambio di messaggi tra il sistema e il cliente deve essere disponibile un server di posta elettronica esterno al sistema.

RNF02: Per poter ritirare o riconsegnare le attrezzature noleggiata il cliente deve comunicare il numero di prenotazione fornito.

RNF03: Il report giornaliero deve essere generato ogni giorno all’orario di chiusura del magazzino.

## Modellazione dei casi d’uso

## Attori e casi d’uso

***Attori Primari:***

* Cliente
* Dipendente
* Tempo

***Attori Secondari:***

* Server di posta elettronica
* ...
* ...

***Casi d’uso:***

* UC01: Effettua prenotazione
* UC02: Effettua pagamento
* UC03: Genera report
* UC04: Consulta prenotazioni
* UC05: Registra consegna
* UC06: Registra restituzione
* UC07: Evadi prenotazione

***Casi d’ uso di inclusione:***

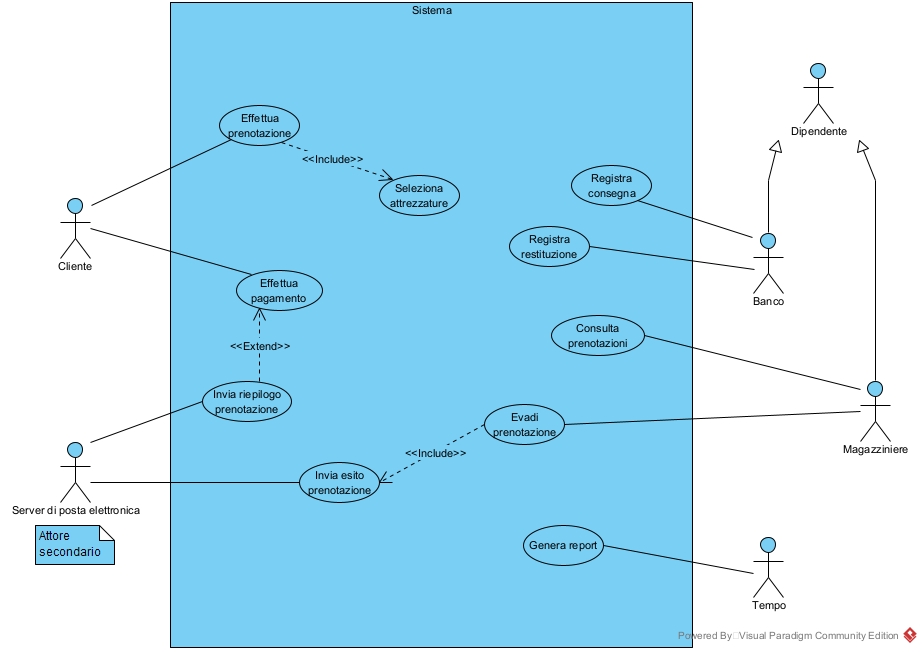
* UC08: Seleziona attrezzature
* UC09: Invia esito prenotazione

***Casi d’ uso di estensione:***

* UC10: Invia riepilogo prenotazione
* ...

|  |  |  |  |
| --- | --- | --- | --- |
| Caso d’uso | Attori Primari | Attori Secondari | Incl. / Ext. |
| UC01:  Effettua prenotazione | Cliente | ... | Seleziona attrezzature |
| UC02:  Effettua pagamento | Cliente | Server posta  elettronica | Invia riepilogo prenotazione |
| UC03:  Genera report | Tempo | ... | ... |
| UC04:  Consulta prenotazioni | Dipendente  (Magazziniere) | ... | ... |
| UC05:  Registra consegna | Dipendente  (Banco) | ... | ... |
| UC06:  Registra restituzione | Dipendente  (Banco) | ... | ... |
| UC07:  Evadi prenotazione | Dipendente  (Magazziniere) | Server posta  elettronica | Invia esito prenotazione |
| UC08:  Seleziona attrezzature | Cliente | ... | Incluso in  Effettua prenotazione |
| UC09:  Invia esito prenotazione | ... | Server posta  elettronica | Incluso in  Evadi prenotazione |
| UC10:  Invia riepilogo prenotazione | ... | Server posta  elettronica | Estende  Effettua pagamento |

## Diagramma dei casi d’uso



## Scenari

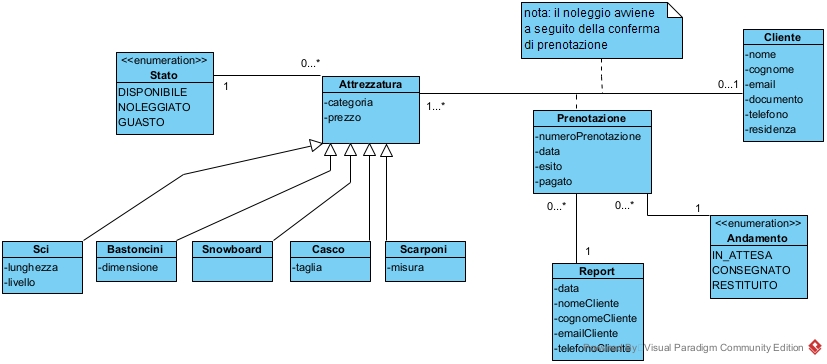
|  |  |
| --- | --- |
| Caso d’uso: | Effettua prenotazione |
| Attore primario | Cliente |
| Attore secondario | *nessuno* |
| Descrizione | Il cliente invia una richiesta di prenotazione per noleggiare una o più attrezzature per il giorno seguente. |
| Pre-Condizioni | *nessuna* |
| Sequenza di eventi  principale | 1. Il caso d’uso inizia quando il cliente avvia la funzionalità di “Effettua prenotazione”. 2. <<Include>> Seleziona attrezzature *(Il cliente seleziona le attrezzature che intende noleggiare).* 3. Il sistema calcola il costo totale per il noleggio. 4. Il sistema chiede una conferma al cliente. 5. Il cliente inserisce i dati richiesti per la prenotazione (nome, cognome, email, documento, residenza e telefono). 6. Il sistema effettua un controllo sui dati. 7. Il sistema mostra al cliente un messaggio di CONFERMA. |
| Post-Condizioni | La richiesta di prenotazione viene creata. |
| Casi d’uso correlati | Seleziona attrezzature |
| Sequenza di eventi  alternativi | nessuna |

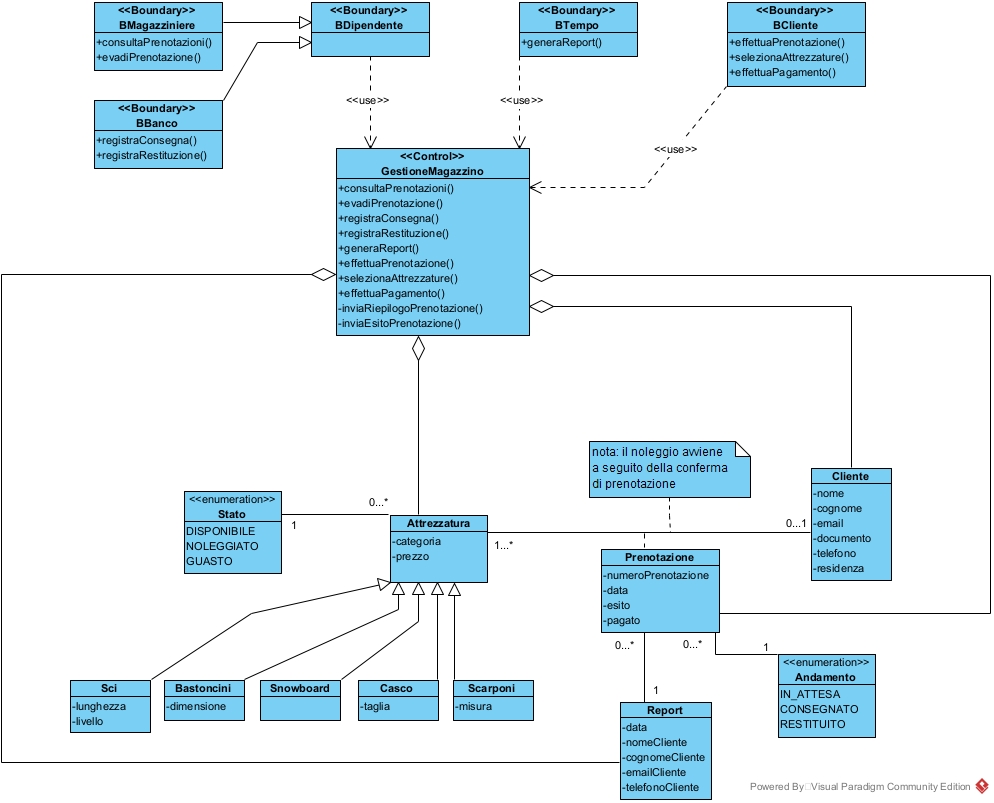
|  |  |
| --- | --- |
| Caso d’uso: | Evadi prenotazione |
| Attore primario | Dipendente (Magazziniere) |
| Attore secondario | Server posta elettronica |
| Descrizione | Il dipendente evade la richiesta di prenotazione ricevuta dal cliente. |
| Pre-Condizioni | Il dipendente ha verificato la disponibilità delle attrezzature richieste nella prenotazione. |
| Sequenza di eventi  principale | 1. Il caso d’uso inizia quando il dipendente avvia la funzionalità di “Evadi prenotazione”. 2. Il sistema richiede il numero di prenotazione e l’esito della richiesta che il dipendente intende evadere. 3. Il sistema effettua una ricerca della prenotazione associata.    1. Se la ricerca non va a buon fine o la prenotazione è già stata gestita il sistema restituisce un messaggio di ERRORE. 4. Il sistema modifica opportunamente l’esito della prenotazione.    1. Se la prenotazione è stata confermata il sistema modifica lo stato di tutte le attrezzature presenti nella richiesta in “noleggiato”. 5. <<Include>> Invia esito prenotazione *(L’esito stabilito dal dipendente viene notificato al cliente tramite mail)*. 6. Il sistema restituisce al dipendente un messaggio di CONFERMA. |
| Post-Condizioni | La prenotazione viene gestita. |
| Casi d’uso correlati | Invia esito prenotazione |
| Sequenza di eventi  alternativi | nessuna |

|  |  |
| --- | --- |
| Caso d’uso: | Registra consegna |
| Attore primario | Dipendente (Banco) |
| Attore secondario | *nessuno* |
| Descrizione | Il dipendente registra l’avvenuta consegna delle attrezzature al cliente che le ha noleggiate. |
| Pre-Condizioni | Il dipendente ha acquisito il numero di prenotazione dal cliente. |
| Sequenza di eventi  principale | 1. Il caso d’uso inizia quando il dipendente avvia la funzionalità di “Registra consegna”. 2. Il sistema richiede il numero di prenotazione che il dipendente intende registrare. 3. Il sistema effettua una ricerca della prenotazione associata.    1. Se la ricerca non va a buon fine o la prenotazione è già stata registrata il sistema restituisce un messaggio di ERRORE. 4. Il sistema modifica l'andamento della prenotazione in “consegnato”. 5. Il sistema restituisce al dipendente un messaggio di CONFERMA. |
| Post-Condizioni | L’avvenuta consegna viene registrata. |
| Casi d’uso correlati | *nessuno* |
| Sequenza di eventi alternativi | *nessuna* |

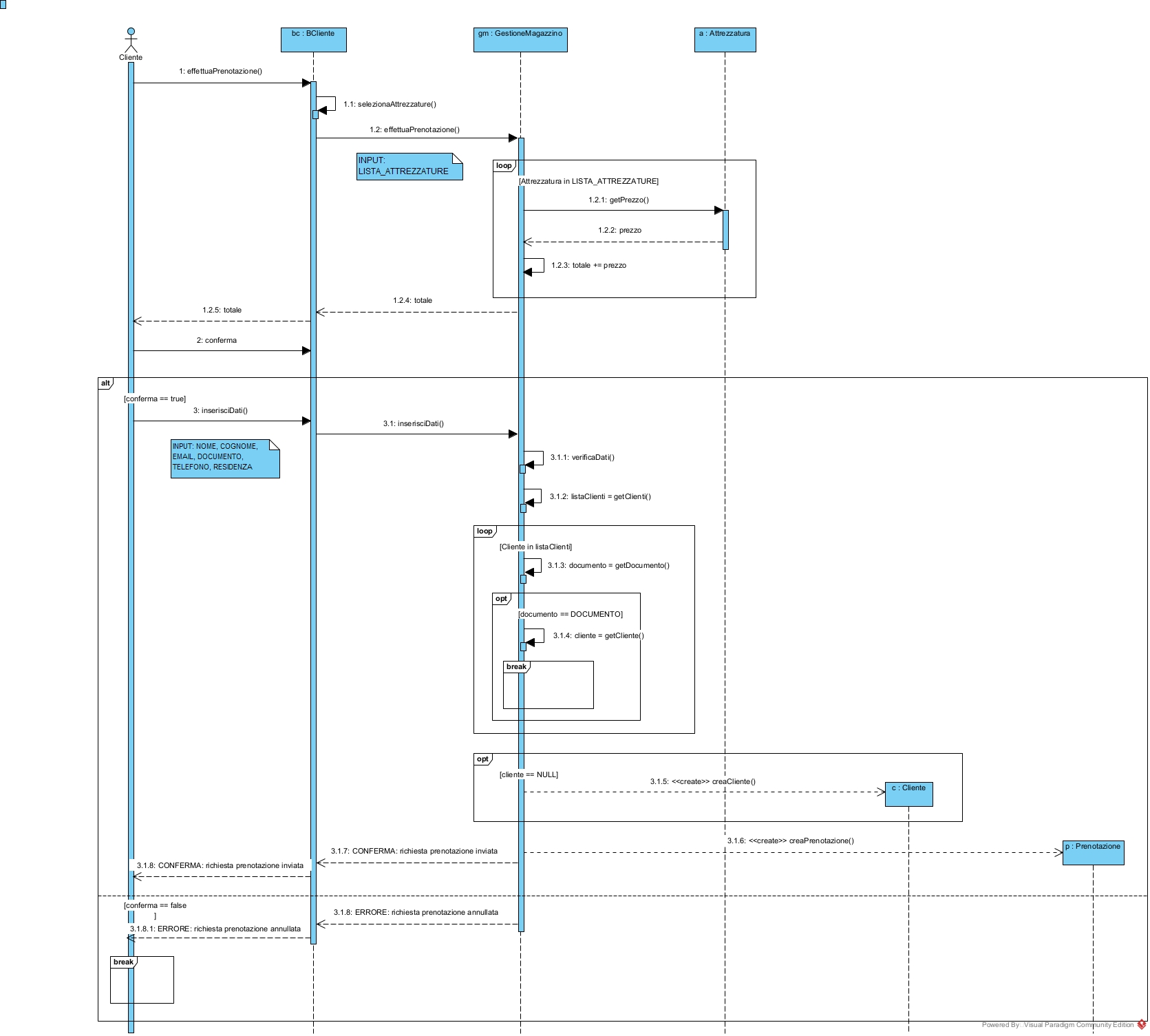
|  |  |
| --- | --- |
| Caso d’uso: | Genera report |
| Attore primario | Tempo |
| Attore secondario | *nessuno* |
| Descrizione | Viene prodotto un report giornaliero in cui sono presenti i noleggi in scadenza per la giornata stessa ma che non sono stati riconsegnati. |
| Pre-Condizioni | *nessuna* |
| Sequenza di eventi  principale | 1. Il caso d’uso inizia ogni giorno all’orario di chiusura del magazzino. 2. Il sistema crea un nuovo report. 3. Il sistema individua i noleggi in scadenza per il giorno stesso che non sono stati riconsegnati. 4. Per ciascun noleggio trovato il sistema ricava le informazioni sul cliente che lo ha effettuato. 5. Il sistema aggiunge al report i noleggi e i dati dei rispettivi clienti individuati. |
| Post-Condizioni | Il report viene memorizzato all’interno del sistema. |
| Casi d’uso correlati | *nessuno* |
| Sequenza di eventi  alternativi | nessuna |

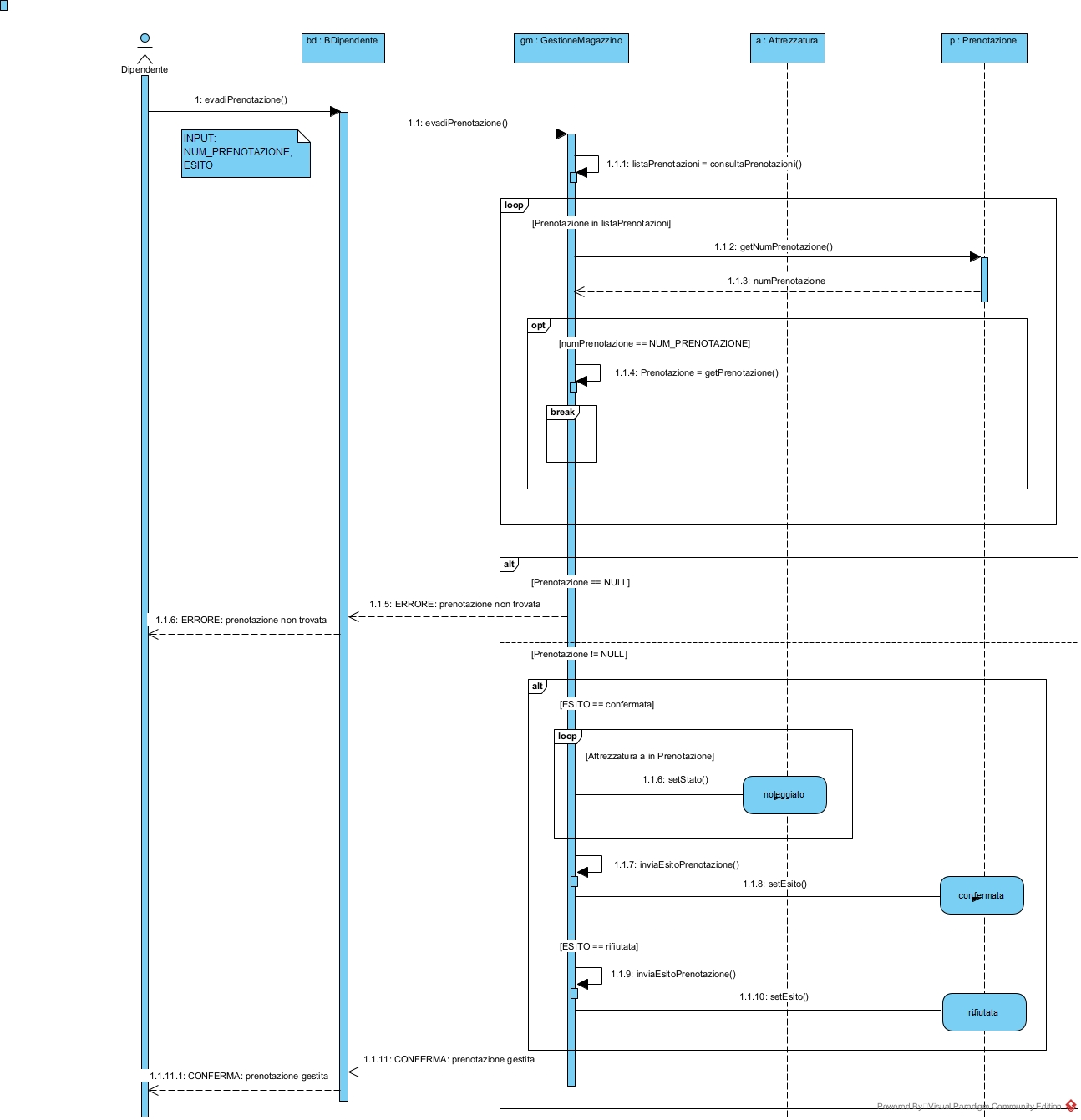
## Diagramma delle classi

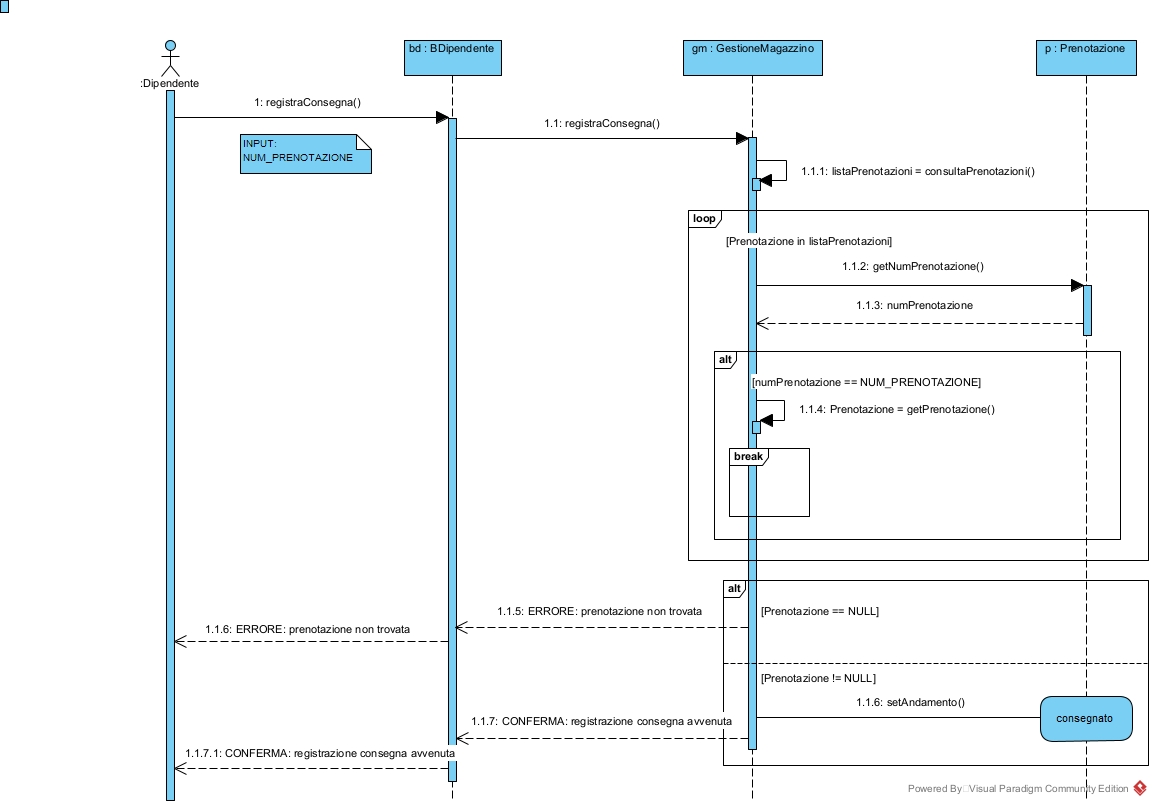


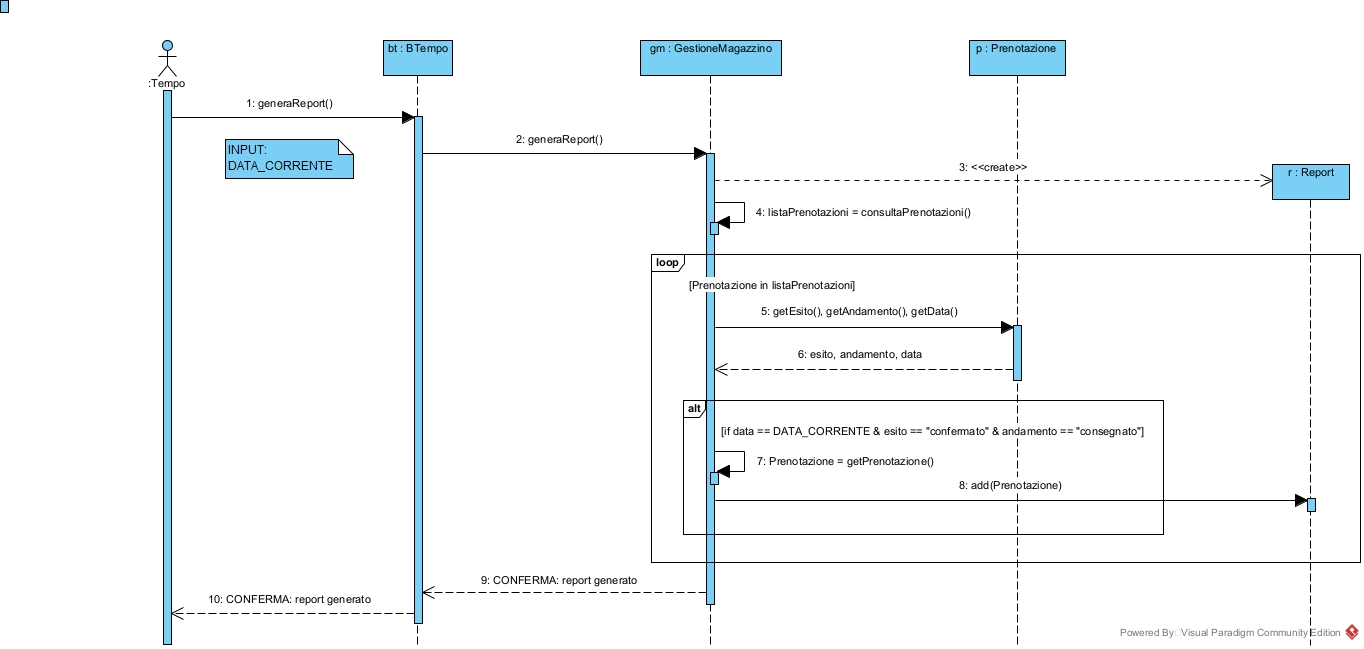


## Diagrammi di sequenza









# Stima dei costi

* Tabelle di riferimento utilizzate per le complessità di dati e transazioni

|  |  |  |  |
| --- | --- | --- | --- |
| ILF DET  RET | 1-19 | 20-50 | >50 |
| 1 | **Low(7)** | **Low(7)** | **Avg(10)** |
| 2-5 | **Low(7)** | **Avg(10)** | **High(15)** |
| >5 | **Avg(10)** | **High(15)** | **High(15)** |

|  |  |  |  |
| --- | --- | --- | --- |
| EQ DET  RET | 1-5 | 6-19 | >19 |
| <2 | **Low(3)** | **Low(3)** | **Avg(4)** |
| 2-3 | **Low(3)** | **Avg(4)** | **High(6)** |
| >3 | **Avg(4)** | **High(6)** | **High(6)** |

|  |  |  |  |
| --- | --- | --- | --- |
| EI DET  RET | 1-4 | 5-15 | >15 |
| <2 | **Low(3)** | **Low(3)** | **Avg(4)** |
| 2 | **Low(3)** | **Avg(4)** | **High(6)** |
| >2 | **Avg(4)** | **High(6)** | **High(6)** |

|  |  |  |  |
| --- | --- | --- | --- |
| EO DET  RET | 1-5 | 6-19 | >19 |
| <2 | **Low(4)** | **Low(4)** | **Avg(5)** |
| 2-3 | **Low(4)** | **Avg(5)** | **High(7)** |
| >3 | **Avg(5)** | **High(7)** | **High(7)** |

**DATABASE (ILF)**

DET: 7 Attributi cliente + 5 Attributi prenotazione + 1 Attributi report + 7 Attributi attrezzatura= 21 DET

RET: 1 Tabella cliente + 1 Tabella prenotazione + 1 Tabella report + 1 Tabella attrezzatura = 4 RET

PESO: AVG(10)

**EFFETTUA PRENOTAZIONE (EI)**

DET: 6 Attributi cliente + 5 Attributi prenotazione + 1 COMANDO + 1 CONFERMA + 1 ERRORE = 14 DET

RET: 1 Tabella cliente + 1 Tabella prenotazione = 2 RET

PESO: AVG(4)

**EVADI PRENOTAZIONE (EI)**

DET: 2 Attributi prenotazione + 2 Attributi attrezzatura + 1 COMANDO + 1 CONFERMA + 1 ERRORE = 7 DET

RET: 1 Tabella prenotazione + 1 Tabella attrezzatura = 2 RET

PESO: AVG(4)

**REGISTRA CONSEGNA (EI)**

DET: 2 Attributi prenotazione + 1 COMANDO + 1 CONFERMA + 1 ERRORE = 5 DET

RET: 1 Tabella prenotazione = 1 RET

PESO: LOW(3)

**GENERA REPORT (EO)**

DET: 5 Attributi prenotazione + 4 Attributi cliente + 1 CONFERMA = 10 DET

RET: 1 Tabella prenotazione + 1 Tabella cliente = 2 RET

PESO: AVG(5)

**SELEZIONA ATTREZZATURE (EQ)**

DET: 9 Attributi attrezzatura + 1 COMANDO + 1 CONFERMA + 1 ERRORE = 12 DET

RET: 1 Tabella attrezzatura = 1 RET

PESO: LOW(3)

**CONSULTA PRENOTAZIONI (EQ)**

DET: 9 Attributi attrezzatura + 5 Attributi prenotazione + 1 COMANDO + 1 CONFERMA = 16 DET

RET: 1 Tabella prenotazione+ 1 Tabella attrezzatura = 2 RET

PESO: AVG(4)

**INVIA ESITO PRENOTAZIONE (EO)**

DET: 2 Attributi prenotazione + 1 COMANDO + 1 CONFERMA = 4 DET

RET: 1 Tabella prenotazione = 1 RET

PESO: LOW(4)

**INVIA RIEPILOGO PRENOTAZIONE (EO)**

DET: 3 Attributi prenotazione + 7 Attributi attrezzatura + 1 CONFERMA = 11DET

RET: 1 Tabella prenotazione + 1 Tabella attrezzatura = 2 RET

PESO: AVG(5)

**REGISTRA CONSEGNA (EI)**

DET: 2 Attributi prenotazione 2 Attributi attrezzatura + 1 COMANDO + 1 CONFERMA + 1 ERRORE = 7 DET

RET: 1 Tabella prenotazione+ 1 Tabella attrezzatura = 2 RET

PESO: AVG(4)

**EFFETTUA PAGAMENTO (EI)**

DET: 7 Attributi cliente + 1 Attributi prenotazione + 1 COMANDO + 1 CONFERMA + 1 ERRORE = 11 DET

RET: 1 Tabella cliente + 1 Tabella prenotazione = 2 RET

PESO: AVG(4)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | VALORE | SEMPLICE | MEDIO | COMPLESSO | TOT |
| NILF | 1 | **7** | **10** | **15** | **10** |
| NEIF | 0 | **5** | **7** | **10** | **0** |
| NEI | 5 | **3** | **4** | **6** | **19** |
| NEO | 3 | **4** | **5** | **7** | **14** |
| NEQ | 2 | **3** | **4** | **6** | **7** |

**UFP = 50**

* Tabella elenco dei fattori correttivi (il cui valore è compreso tra 0 e 5)

|  |  |  |  |
| --- | --- | --- | --- |
| FATTORI CORRETTIVI | | | |
| COMUNICAZIONE DATI | | | 4 |
| DISTRIBUZIONE ELABORAZIONE | | | 2 |
| PRESTAZIONI | | | 2 |
| UTILIZZO INTENSIVO CONFIGURAZIONE | | | 0 |
| FREQUENZA DELLE TRANSAZIONI | | | 2 |
| INSERIMENTO DATI INTERATTIVO | | | 2 |
| EFFICIENZA PER L’UTENTE FINALE | | | 4 |
| AGGIORNAMENTO INTERATTIVO | | | 4 |
| COMPLESSITA’ ELABORATIVA | | | 1 |
| RIUSABILITA’ | | | 5 |
| FACILITA’ INSTALLAZIONE | | | 2 |
| FACILITA’ GESTIONE OPERATIVA |  |  | 5 |
| MOLTEPLICITA’ DI SITI | | | 0 |
| FACILITA’ DI MODIFICA | | | 4 |
| TOT | | | 37 |

**AFP = 0.65 + 0.1 \* 37 = 1.02**

**FP = AFP \* UFP = 51**

# Piano di test funzionale

**PIANO DI TEST UTILIZZANDO IL METODO DEL *CATEGORY-PARTITION TESTING* PER LA FUNZIONALITÀ “Effettua prenotazione”.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **NOME** | **COGNOME** | **EMAIL** | **DOCUMENTO** | **TELEFONO** | **RESIDENZA** |
| * Stringa di caratteri di lunghezza <= 30. * Stringa di caratteri di lunghezza > 30 [ERROR]. * Stringa che contiene simboli che non sono caratteri [ERROR]. | * Stringa di caratteri di lunghezza <= 30. * Stringa di caratteri di lunghezza > 30 [ERROR]. * Stringa che contiene simboli che non sono caratteri [ERROR]. | * Stringa in cui sono presenti i simboli ‘.’ e ‘@’. * Stringa in cui non sono presenti i simboli ‘.’ e ‘@’ [ERROR]. | * Stringa di caratteri di lunghezza = 9. * Stringa di caratteri di lunghezza > di 9 [ERROR]. * Stringa di caratteri di lunghezza < di 9 [ERROR]. * Stringa che contiene simboli che non sono caratteri o numeri. [ERROR]. | * Stringa di numeri di lunghezza = 10. * Stringa di numeri di lunghezza > di 10 [ERROR]. * Stringa di numeri di lunghezza < di 10 [ERROR]. * Stringa che contiene simboli che non sono numeri [ERROR]. | * Stringa di caratteri di lunghezza <= 30. * Stringa di caratteri di lunghezza > 30 [ERROR]. * Stringa che contiene simboli che non sono caratteri o numeri [ERROR]. |

Il numero di test da effettuare senza vincoli è: 3 \* 3 \* 2 \* 3 \* 3 \* 3 = 486.

Introduciamo i vincoli [ERROR].

Il numero di test da eseguire per testare singolarmente i vincoli è: 13 (2 per nome, 2 per cognome, 1 per email, 3 per documento, 3 per telefono, 2 per residenza).

Il numero di test da effettuare con i vincoli è: (1 \* 1 \* 1 \* 1 \* 1 \* 1) + 13 = 14.

**PIANO DI TEST UTILIZZANDO IL METODO DEL *CATEGORY-PARTITION TESTING* PER LA FUNZIONALITÀ “Effettua prenotazione”.**

|  |  |
| --- | --- |
| **NUMERO PRENOTAZIONE** | **ESITO** |
| * Stringa di numeri. * Stringa che contiene simboli che non sono numeri [ERROR]. | * Singolo carattere. * Stringa di caratteri di lunghezza > 1 [ERROR]. * Singolo carattere che contiene i caratteri ‘C’ o ‘R’. [ERROR]. |

Il numero di test da effettuare senza vincoli è: 2 \* 3 = 6.

Introduciamo i vincoli [ERROR].

Il numero di test da eseguire per testare singolarmente i vincoli è: 3 (1 per numero prenotazione, 2 per esito).

Il numero di test da effettuare con i vincoli è: (1 \* 1) + 3 = 4.

**PIANO DI TEST UTILIZZANDO IL METODO DEL *CATEGORY-PARTITION TESTING* PER LA FUNZIONALITÀ “Registra consegna”.**

|  |
| --- |
| **NUMERO PRENOTAZIONE** |
| * Stringa di numeri. * Stringa che contiene simboli che non sono numeri [ERROR]. |

Il numero di test da effettuare senza vincoli è: 2.

**TEST SUITE PER LA FUNZIONALITÀ “Effettua prenotazione”.**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Descrizione** | **Classi di equivalenza coperte** | **Pre-condizioni** | **Input** | **Output Attesi** | **Post-condizioni Attese** |
| 1 | Tutti gli input validi. | Nome valido, cognome valido, email valido, documento valido, telefono valido, residenza valido. | Il database è inizializzato  correttamente. | {Nome: “Manuel”,  Cognome: “Parasmo”,  Email: “manuel@mail.com”,  Documento: “FO0000000”,  Telefono: “3516789012”,  Residenza: “Formia 46” } | Prenotazione creata. | La prenotazione viene creata con i dati del cliente (se il cliente non era già presente nel database questo viene aggiunto) e il cliente riceve un messaggio di conferma. |
| 2 | Nome stringa > 30 caratteri. | Nome stringa > 30 caratteri [ERROR], cognome valido, email valido, documento valido, telefono valido, residenza valido. | Il cliente ha selezionato la  funzionalità di effettua prenotazione. Il  sistema richiede  all’utente l’inserimento  del nome. | {Nome: “Maaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaanuel”,  Cognome: “Parasmo”,  Email: “manuel@mail.com”,  Documento: “FO0000000”,  Telefono: “3516789012”,  Residenza: “Formia 46” } | Nome troppo lungo. | Il sistema chiede di reinserire il nome. |
| 3 | Nome stringa con simboli. | Nome stringa con simboli [ERROR], cognome valido, email valido, documento valido, telefono valido, residenza valido. | Il cliente ha selezionato la  funzionalità di effettua prenotazione. Il  sistema richiede  all’utente l’inserimento  del nome. | {Nome: “Manuel00-”,  Cognome: “Parasmo”,  Email: “manuel@mail.com”,  Documento: “FO0000000”,  Telefono: “3516789012”,  Residenza: “Formia 46” } | Nome non valido. | Il sistema chiede di reinserire il nome. |
| 4 | Cognome stringa > 30 caratteri. | Cognome stringa > 30 caratteri [ERROR], nome valido, email valido, documento valido, telefono valido, residenza valido. | Il cliente ha selezionato la  funzionalità di effettua prenotazione. Il  sistema richiede  all’utente l’inserimento  del cognome. | {Nome: “Manuel”,  Cognome: “Paaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaarasmo”,  Email: “manuel@mail.com”,  Documento: “FO0000000”,  Telefono: “3516789012”,  Residenza: “Formia 46” } | Cognome troppo lungo. | Il sistema chiede di reinserire il cognome. |
| 5 | Cognome stringa con simboli. | Cognome stringa con simboli [ERROR], nome valido, email valido, documento valido, telefono valido, residenza valido. | Il cliente ha selezionato la  funzionalità di effettua prenotazione. Il  sistema richiede  all’utente l’inserimento  del cognome. | {Nome: “Manuel”,  Cognome: “Parasmo01??”,  Email: “manuel@mail.com”,  Documento: “FO0000000”,  Telefono: “3516789012”,  Residenza: “Formia 46” } | Cognome non valido. | Il sistema chiede di reinserire il cognome. |
| 6 | Email senza simboli ‘.’ e ‘@’. | Email senza simboli ‘.’ e ‘@’ [ERROR], nome valido, cognome valido, documento valido, telefono valido, residenza valido. | Il cliente ha selezionato la  funzionalità di effettua prenotazione. Il  sistema richiede  all’utente l’inserimento  della email. | {Nome: “Manuel”,  Cognome: “Parasmo”,  Email: “manuelmail”,  Documento: “FO0000000”,  Telefono: “3516789012”,  Residenza: “Formia 46” } | Email non valida. | Il sistema chiede di reinserire la email. |
| 7 | Documento stringa > 9 caratteri. | Documento stringa > 9 caratteri [ERROR], nome valido, cognome valido, email valido, telefono valido, residenza valido. | Il cliente ha selezionato la  funzionalità di effettua prenotazione. Il  sistema richiede  all’utente l’inserimento  del documento. | {Nome: “Manuel”,  Cognome: “Parasmo”,  Email: “manuel@mail.com”,  Documento: “FO0000000RM”,  Telefono: “3516789012”,  Residenza: “Formia 46” } | Documento troppo lungo. | Il sistema chiede di reinserire il documento. |
| 8 | Documento stringa < 9 caratteri. | Documento stringa < 9 caratteri [ERROR], nome valido, cognome valido, email valido, telefono valido, residenza valido. | Il cliente ha selezionato la  funzionalità di effettua prenotazione. Il  sistema richiede  all’utente l’inserimento  del documento. | {Nome: “Manuel”,  Cognome: “Parasmo”,  Email: “manuel@mail.com”,  Documento: “FO00”,  Telefono: “3516789012”,  Residenza: “Formia 46” } | Documento troppo corto. | Il sistema chiede di reinserire il documento. |
| 9 | Documenti stringa con simboli. | Documento stringa con simboli [ERROR], nome valido, cognome valido, email valido, telefono valido, residenza valido. | Il cliente ha selezionato la  funzionalità di effettua prenotazione. Il  sistema richiede  all’utente l’inserimento  del documento. | {Nome: “Manuel”,  Cognome: “Parasmo”,  Email: “manuel@mail.com”,  Documento: “FO00000??.-”,  Telefono: “3516789012”,  Residenza: “Formia 46” } | Documento non valido. | Il sistema chiede di reinserire il documento. |
| 10 | Telefono stringa > 10 caratteri. | Telefono stringa > 11 caratteri [ERROR], nome valido, cognome valido, email valido, documento valido, residenza valido. | Il cliente ha selezionato la  funzionalità di effettua prenotazione. Il  sistema richiede  all’utente l’inserimento  del telefono. | {Nome: “Manuel”,  Cognome: “Parasmo”,  Email: “manuel@mail.com”,  Documento: “FO0000000”,  Telefono: “35167890123112”,  Residenza: “Formia 46” } | Telefono troppo lungo. | Il sistema chiede di reinserire il telefono. |
| 11 | Telefono stringa < 10 caratteri. | Telefono stringa < 11 caratteri [ERROR], nome valido, cognome valido, email valido, documento valido, residenza valido. | Il cliente ha selezionato la  funzionalità di effettua prenotazione. Il  sistema richiede  all’utente l’inserimento  del telefono. | {Nome: “Manuel”,  Cognome: “Parasmo”,  Email: “manuel@mail.com”,  Documento: “FO0000000”,  Telefono: “3516783”,  Residenza: “Formia 46” } | Telefono troppo corto. | Il sistema chiede di reinserire il telefono. |
| 12 | Telefono stringa con simboli. | Telefono stringa con simboli [ERROR], nome valido, cognome valido, email valido, documento valido, residenza valido. | Il cliente ha selezionato la  funzionalità di effettua prenotazione. Il  sistema richiede  all’utente l’inserimento  del telefono. | {Nome: “Manuel”,  Cognome: “Parasmo”,  Email: “manuel@mail.com”,  Documento: “FO0000000”,  Telefono: “35168@ab”,  Residenza: “Formia 46” } | Telefono non valido. | Il sistema chiede di reinserire il telefono. |
| 13 | Residenza stringa > 30 caratteri. | Residenza stringa > 30 caratteri [ERROR], nome valido, cognome valido, email valido, telefono valido, documento valido. | Il cliente ha selezionato la  funzionalità di effettua prenotazione. Il  sistema richiede  all’utente l’inserimento  della residenza. | {Nome: “Manuel”,  Cognome: “Parasmo”,  Email: “manuel@mail.com”,  Documento: “FO0000000”,  Telefono: “3516789012”,  Residenza: “Fooooooooooooooooooooooooooooooormia 46” } | Residenza troppo lunga. | Il sistema chiede di reinserire la residenza. |
| 14 | Residenza stringa con simboli. | Residenza stringa con simboli [ERROR], nome valido, cognome valido, email valido, telefono valido, documento valido. | Il cliente ha selezionato la  funzionalità di effettua prenotazione. Il  sistema richiede  all’utente l’inserimento  della residenza. | {Nome: “Manuel”,  Cognome: “Parasmo”,  Email: “manuel@mail.com”,  Documento: “FO0000000”,  Telefono: “3516789012”,  Residenza: “Formia 46!!!!#” } | Residenza non valida. | Il sistema chiede di reinserire la residenza. |

**TEST SUITE PER LA FUNZIONALITÀ “Evadi prenotazione”.**

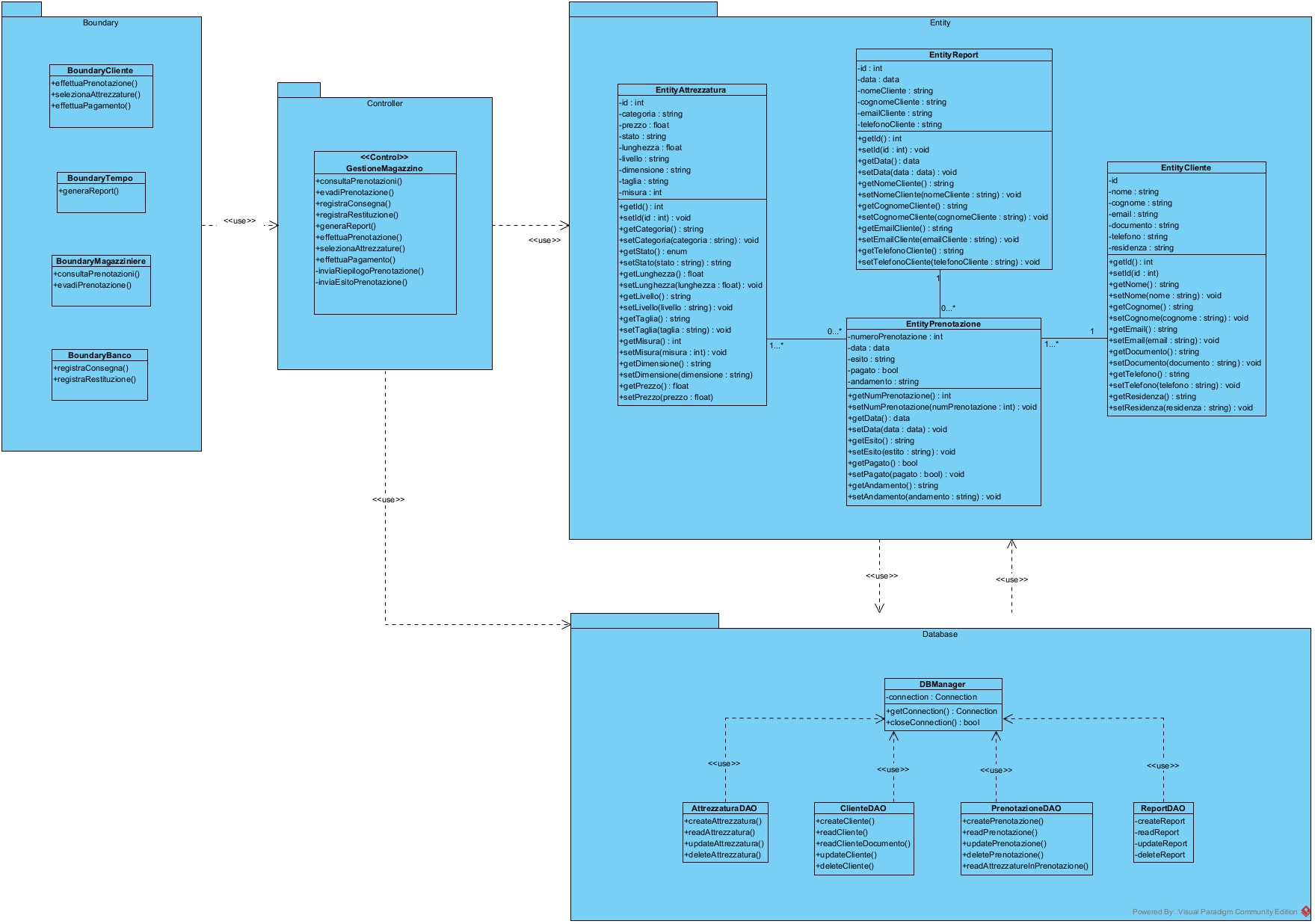
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Descrizione** | **Classi di equivalenza coperte** | **Pre-condizioni** | **Input** | **Output Attesi** | **Post-condizioni Attese** |
| 1 | Tutti gli input validi. | Numero prenotazione valido, esito valido. | Il database è inizializzato  correttamente. | {Numero prenotazione: “716”,  Esito: “C” } | Prenotazione modificata. | Il magazziniere riceve un messaggio di conferma della modifica prenotazione. |
| 2 | Numero prenotazione stringa con simboli o caratteri. | Numero prenotazione stringa con caratteri [ERROR], esito valido. | Il magazziniere ha selezionato la  funzionalità di evadi prenotazione. Il  sistema richiede  al dipendente l’inserimento  del numero prenotazione. | {Numero prenotazione: “11aa@a”,  Esito: “C” } | Numero prenotazione non valido. | Il sistema chiede di reinserire il numero prenotazione. |
| 3 | Esito stringa > 1 carattere. | Esito stringa > 1 carattere [ERROR], numero prenotazione valido. | Il magazziniere ha selezionato la  funzionalità di evadi prenotazione. Il  sistema richiede  al dipendente l’inserimento  del carattere di esito. | {Numero prenotazione: “716”,  Esito: “CO” } | Esito troppo lungo. | Il sistema chiede di reinserire l’esito. |
| 4 | Esito senza caratteri ‘C’ o ‘R’. | Esito senza caratteri ‘C’ o ‘R’ [ERROR], numero prenotazione valido. | Il magazziniere ha selezionato la  funzionalità di evadi prenotazione. Il  sistema richiede  al dipendente l’inserimento  del carattere di esito. | {Numero prenotazione: “716”,  Esito: “NO” } | Esito non valido. | Il sistema chiede di reinserire l’esito. |

**TEST SUITE PER LA FUNZIONALITÀ “Registra consegna”.**

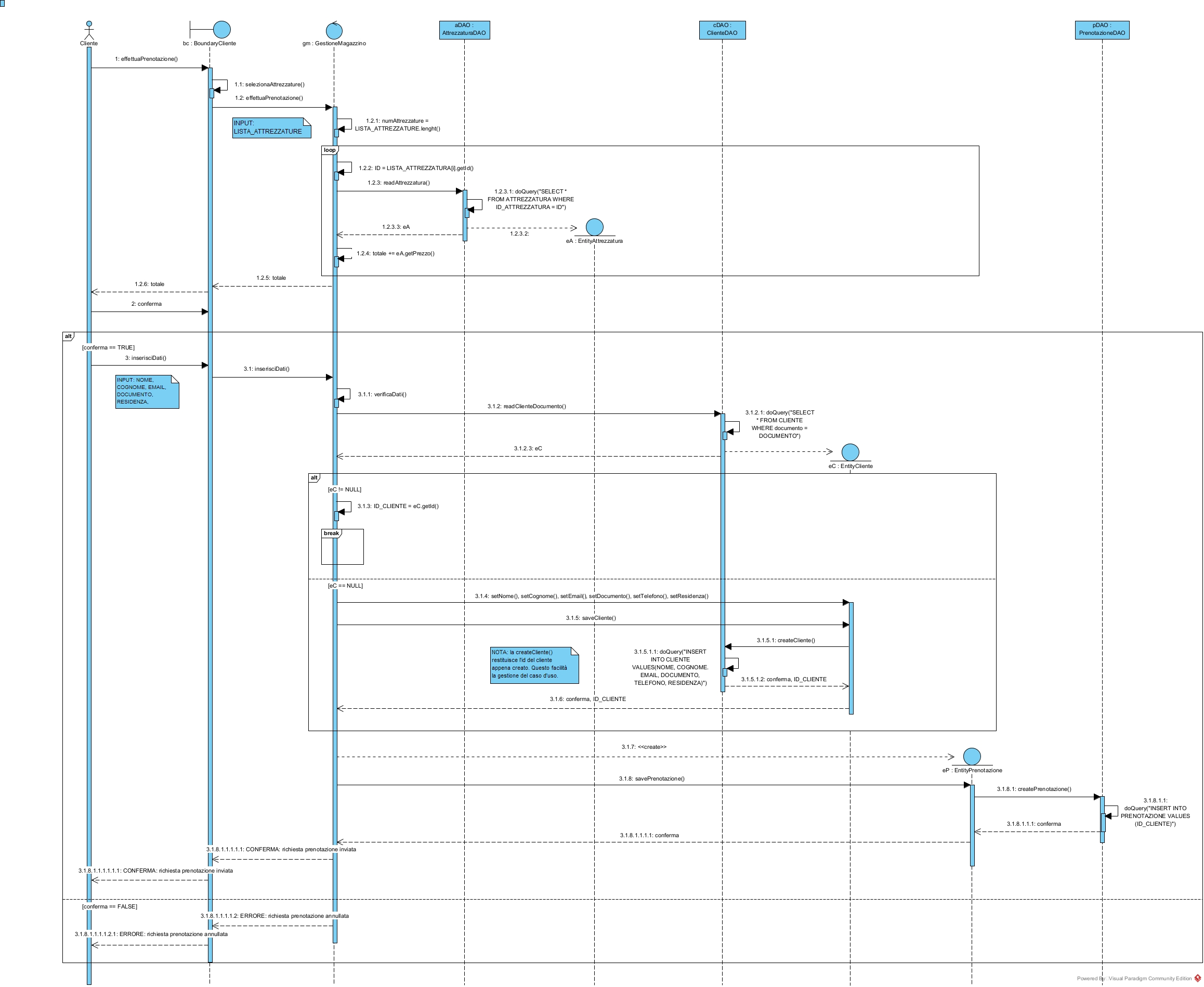
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Descrizione** | **Classi di equivalenza coperte** | **Pre-condizioni** | **Input** | **Output Attesi** | **Post-condizioni Attese** |
| 1 | Tutti gli input validi. | Numero prenotazione valido. | Il database è inizializzato  correttamente. | {Numero prenotazione: “716” } | Prenotazione modificata. | Il banco riceve un messaggio di conferma della modifica prenotazione. |
| 2 | Numero prenotazione stringa con simboli o caratteri. | Numero prenotazione stringa con caratteri [ERROR]. | Il magazziniere ha selezionato la  funzionalità di evadi prenotazione. Il  sistema richiede  al dipendente l’inserimento  del numero prenotazione. | {Numero prenotazione: “11aa@a” } | Numero prenotazione non valido. | Il sistema chiede di reinserire il numero prenotazione. |

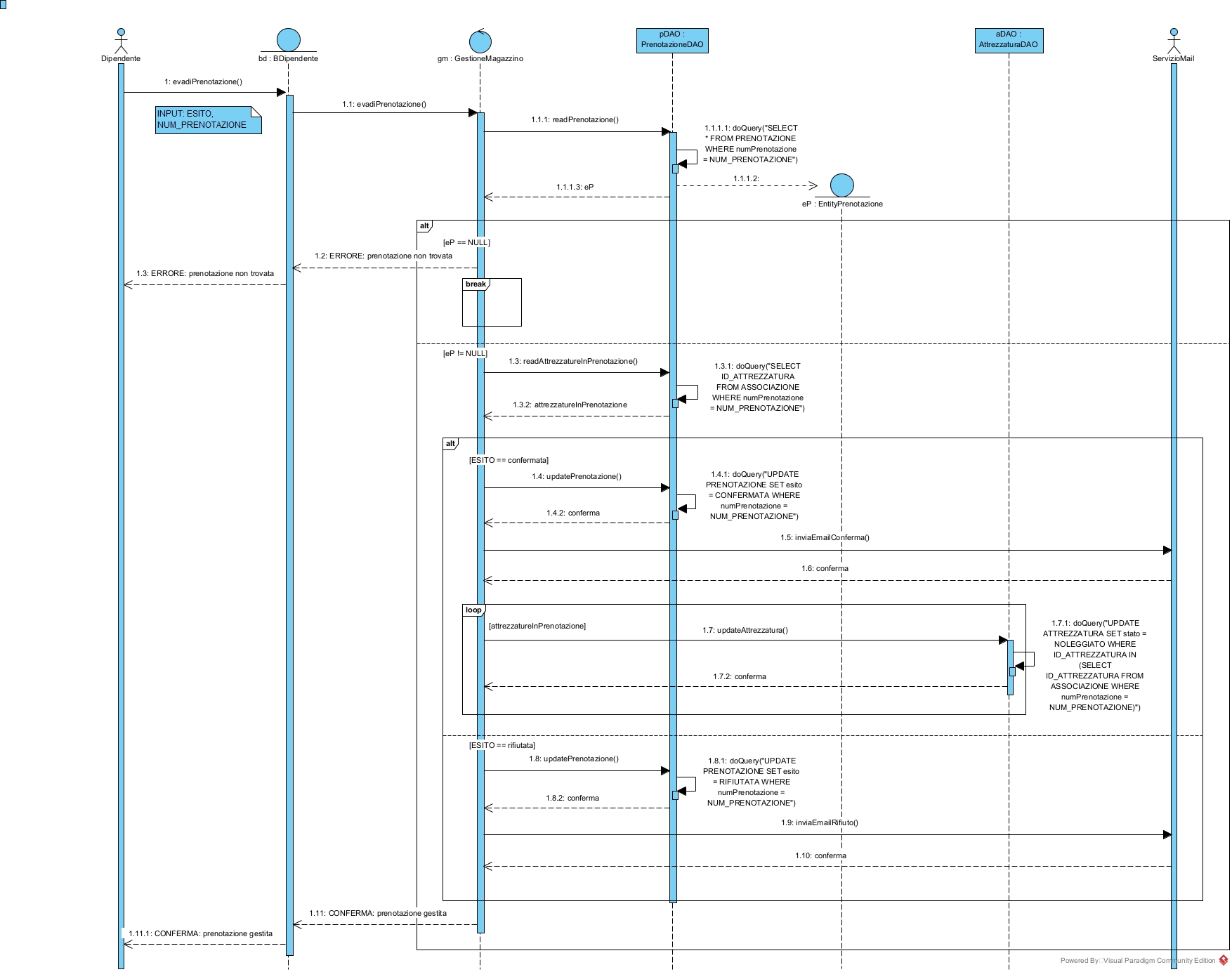
# Progettazione

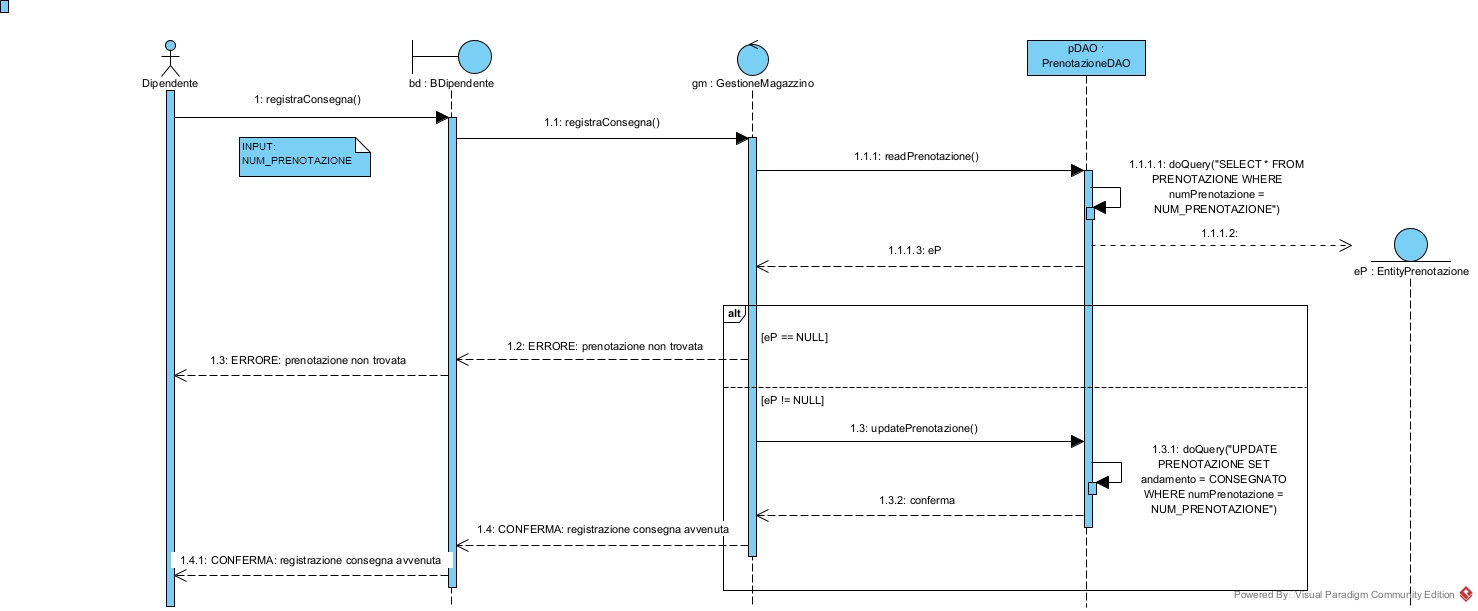
## Diagramma delle classi

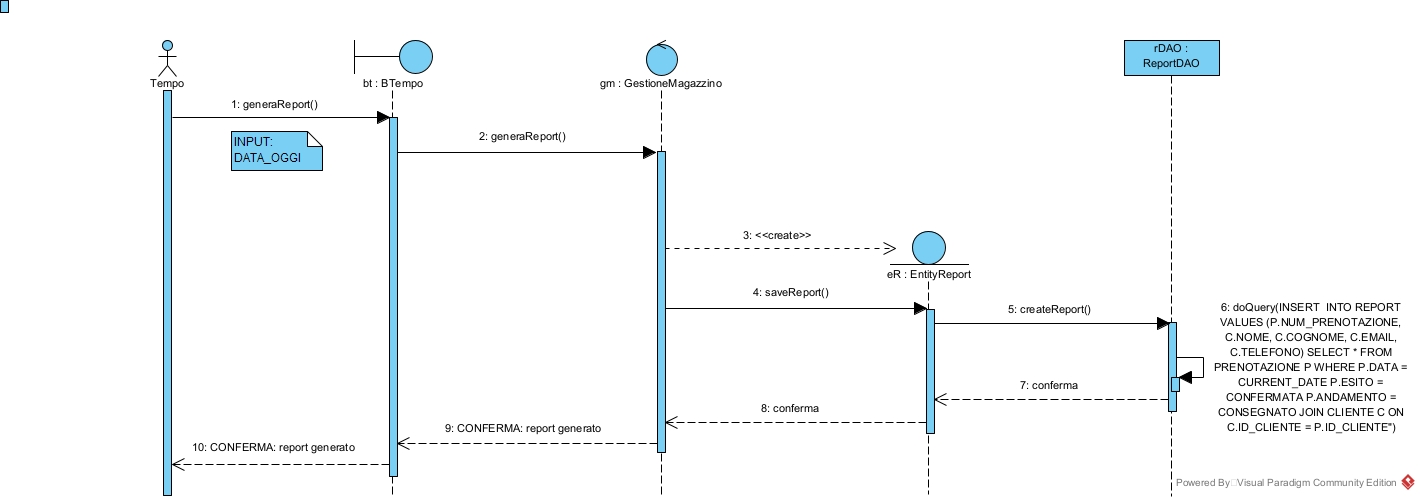


## Diagrammi di sequenza









# Implementazione

**PACKAGE:**

* Boundary
* Control
* Database
* Entity
* Exception

**CLASSI:**

* BoundaryMagazziniere
* BoundaryCliente
* BoundaryBanco
* BoundaryTempo
* GestioneMagazzino
* AttrezzaturaDAO
* PrenotazioneDAO
* ClienteDAO
* ReportDAO
* DBManager
* EntityAttrezzatura
* EntityCliente
* EntityPrenotazione
* EntityReport
* DAOException
* DBConnectionException
* OperationException

**ECCEZIONI:**

* DAOException: eccezioni generate a seguito di una scorretta query eseguita sul database.
* DBConnectionException: eccezioni generate a seguito di un errore nella connessione al database.
* OperationException: eccezioni generate dal control.

**ARTEFATTI:**

* Il database utilizzato è ‘H2’ (il file .jar per il funzionamento del database è incluso nella cartella presente nel repository).
* La versione di java su cui è stato scritto il codice è ‘1.8.0’.

**NOTE PER L'UTILIZZO:**

* Per poter permettere l’utilizzo del software è stata implementata una funzionalità “fittizia”, si tratta di ‘Consulta prenotazioni’. Questa dovrebbe fornire una lista di prenotazioni con il relativo numero e le relative attrezzature richieste dal cliente; tuttavia, per poter permettere l’esecuzione delle funzionalità di ‘Evadi prenotazione’ e ‘Registra consegna' (che richiedono la conoscenza del numero di prenotazione) la funzionalità di ‘Consulta prenotazioni’ è stata sviluppata in maniera tale da restituire il numero di prenotazione dell’ultima prenotazione inserite. Si consiglia pertanto di testare il software nel seguente ordine:

1. Effettua prenotazione (Boundary Cliente)

2. Consulta prenotazioni (Boundary Magazziniere)

3. Evadi prenotazione (Boundary Magazziniere)

4. Registra consegna (Boundary Banco)

# Testing

## Test funzionale

**RISULTATI DELLA TEST EXECUTION PER LA FUNZIONALITÀ “Effettua prenotazione”.**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Descrizione** | **Classi di equivalenza coperte** | **Pre-condizioni** | **Input** | **Output Attesi** | **Post-condizioni Attese** | **Output Ottenuti** | **Post-condizioni Ottenute** | **Esito**  **(*FAIL*, *PASS*)** |
| 1 | Tutti gli input validi. | Nome valido, cognome valido, email valido, documento valido, telefono valido, residenza valido. | Il database è inizializzato correttamente. | {Nome: “Manuel”,Cognome: “Parasmo”,  Email: “manuel@mail.com”,  Documento: “FO0000000”,  Telefono: “3516789012”,  Residenza: “Formia 46” } | Prenotazione creata. | La prenotazione viene creata con i dati del cliente (se il cliente non era già presente nel database questo viene aggiunto) e il cliente riceve un messaggio di conferma. | \\ | \\ | **PASS** |
| 2 | Nome stringa > 30 caratteri. | Nome stringa > 30 caratteri [ERROR], cognome valido, email valido, documento valido, telefono valido, residenza valido. | Il cliente ha selezionato la  funzionalità di effettua prenotazione. Il  sistema richiede  all’utente l’inserimento  del nome. | {Nome: “Maaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaanuel”,  Cognome: “Parasmo”,  Email: “manuel@mail.com”,  Documento: “FO0000000”,  Telefono: “3516789012”,  Residenza: “Formia 46” } | Nome troppo lungo. | Il sistema chiede di reinserire il nome. | \\ | \\ | **PASS** |
| 3 | Nome stringa con simboli. | Nome stringa con simboli [ERROR], cognome valido, email valido, documento valido, telefono valido, residenza valido. | Il cliente ha selezionato la  funzionalità di effettua prenotazione. Il  sistema richiede  all’utente l’inserimento  del nome. | {Nome: “Manuel00-”,  Cognome: “Parasmo”,  Email: “manuel@mail.com”,  Documento: “FO0000000”,  Telefono: “3516789012”,  Residenza: “Formia 46” } | Nome non valido. | Il sistema chiede di reinserire il nome. | \\ | \\ | **PASS** |
| 4 | Cognome stringa > 30 caratteri. | Cognome stringa > 30 caratteri [ERROR], nome valido, email valido, documento valido, telefono valido, residenza valido. | Il cliente ha selezionato la  funzionalità di effettua prenotazione. Il  sistema richiede  all’utente l’inserimento  del cognome. | {Nome: “Manuel”,  Cognome: “Paaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaarasmo”,  Email: “manuel@mail.com”,  Documento: “FO0000000”,  Telefono: “3516789012”,  Residenza: “Formia 46” } | Cognome troppo lungo. | Il sistema chiede di reinserire il cognome. | \\ | \\ | **PASS** |
| 5 | Cognome stringa con simboli. | Cognome stringa con simboli [ERROR], nome valido, email valido, documento valido, telefono valido, residenza valido. | Il cliente ha selezionato la  funzionalità di effettua prenotazione. Il  sistema richiede  all’utente l’inserimento  del cognome. | {Nome: “Manuel”,  Cognome: “Parasmo01??”,  Email: “manuel@mail.com”,  Documento: “FO0000000”,  Telefono: “3516789012”,  Residenza: “Formia 46” } | Cognome non valido. | Il sistema chiede di reinserire il cognome. | \\ | \\ | **PASS** |
| 6 | Email senza simboli ‘.’ e ‘@’. | Email senza simboli ‘.’ e ‘@’ [ERROR], nome valido, cognome valido, documento valido, telefono valido, residenza valido. | Il cliente ha selezionato la  funzionalità di effettua prenotazione. Il  sistema richiede  all’utente l’inserimento  della email. | {Nome: “Manuel”,  Cognome: “Parasmo”,  Email: “manuelmail”,  Documento: “FO0000000”,  Telefono: “3516789012”,  Residenza: “Formia 46” } | Email non valida. | Il sistema chiede di reinserire la email. | \\ | \\ | **PASS** |
| 7 | Documento stringa > 9 caratteri. | Documento stringa > 9 caratteri [ERROR], nome valido, cognome valido, email valido, telefono valido, residenza valido. | Il cliente ha selezionato la  funzionalità di effettua prenotazione. Il  sistema richiede  all’utente l’inserimento  del documento. | {Nome: “Manuel”,  Cognome: “Parasmo”,  Email: “manuel@mail.com”,  Documento: “FO0000000RM”,  Telefono: “3516789012”,  Residenza: “Formia 46” } | Documento troppo lungo. | Il sistema chiede di reinserire il documento. | \\ | \\ | **PASS** |
| 8 | Documento stringa < 9 caratteri. | Documento stringa < 9 caratteri [ERROR], nome valido, cognome valido, email valido, telefono valido, residenza valido. | Il cliente ha selezionato la  funzionalità di effettua prenotazione. Il  sistema richiede  all’utente l’inserimento  del documento. | {Nome: “Manuel”,  Cognome: “Parasmo”,  Email: “manuel@mail.com”,  Documento: “FO00”,  Telefono: “3516789012”,  Residenza: “Formia 46” } | Documento troppo corto. | Il sistema chiede di reinserire il documento. | \\ | \\ | **PASS** |
| 9 | Documenti stringa con simboli. | Documento stringa con simboli [ERROR], nome valido, cognome valido, email valido, telefono valido, residenza valido. | Il cliente ha selezionato la  funzionalità di effettua prenotazione. Il  sistema richiede  all’utente l’inserimento  del documento. | {Nome: “Manuel”,  Cognome: “Parasmo”,  Email: “manuel@mail.com”,  Documento: “FO00000??.-”,  Telefono: “3516789012”,  Residenza: “Formia 46” } | Documento non valido. | Il sistema chiede di reinserire il documento. | \\ | \\ | **PASS** |
| 10 | Telefono stringa > 10 caratteri. | Telefono stringa > 11 caratteri [ERROR], nome valido, cognome valido, email valido, documento valido, residenza valido. | Il cliente ha selezionato la  funzionalità di effettua prenotazione. Il  sistema richiede  all’utente l’inserimento  del telefono. | {Nome: “Manuel”,  Cognome: “Parasmo”,  Email: “manuel@mail.com”,  Documento: “FO0000000”,  Telefono: “35167890123112”,  Residenza: “Formia 46” } | Telefono troppo lungo. | Il sistema chiede di reinserire il telefono. | \\ | \\ | **PASS** |
| 11 | Telefono stringa < 10 caratteri. | Telefono stringa < 11 caratteri [ERROR], nome valido, cognome valido, email valido, documento valido, residenza valido. | Il cliente ha selezionato la  funzionalità di effettua prenotazione. Il  sistema richiede  all’utente l’inserimento  del telefono. | {Nome: “Manuel”,  Cognome: “Parasmo”,  Email: “manuel@mail.com”,  Documento: “FO0000000”,  Telefono: “3516783”,  Residenza: “Formia 46” } | Telefono troppo corto. | Il sistema chiede di reinserire il telefono. | \\ | \\ | **PASS** |
| 12 | Telefono stringa con simboli [ERROR], nome valido, cognome valido, email valido, documento valido, residenza valido. | Telefono stringa con simboli [ERROR], nome valido, cognome valido, email valido, documento valido, residenza valido. | Il cliente ha selezionato la  funzionalità di effettua prenotazione. Il  sistema richiede  all’utente l’inserimento  del telefono. | {Nome: “Manuel”,  Cognome: “Parasmo”,  Email: “manuel@mail.com”,  Documento: “FO0000000”,  Telefono: “35167@ab”,  Residenza: “Formia 46” } | Telefono non valido. | Il sistema chiede di reinserire il telefono. | \\ | \\ | **PASS** |
| 13 | Residenza stringa > 30 caratteri [ERROR], nome valido, cognome valido, email valido, telefono valido, documento valido. | Residenza stringa > 30 caratteri [ERROR], nome valido, cognome valido, email valido, telefono valido, documento valido. | Il cliente ha selezionato la  funzionalità di effettua prenotazione. Il  sistema richiede  all’utente l’inserimento  della residenza. | {Nome: “Manuel”,  Cognome: “Parasmo”,  Email: “manuel@mail.com”,  Documento: “FO0000000”,  Telefono: “3516789012”,  Residenza: “Fooooooooooooooooooooooooooooooormia 46” } | Residenza troppo lunga. | Il sistema chiede di reinserire la residenza. | \\ | \\ | **PASS** |
| 14 | Residenza stringa con simboli [ERROR], nome valido, cognome valido, email valido, telefono valido, documento valido. | Residenza stringa con simboli [ERROR], nome valido, cognome valido, email valido, telefono valido, documento valido. | Il cliente ha selezionato la  funzionalità di effettua prenotazione. Il  sistema richiede  all’utente l’inserimento  della residenza. | {Nome: “Manuel”,  Cognome: “Parasmo”,  Email: “manuel@mail.com”,  Documento: “FO0000000”,  Telefono: “35167890123”,  Residenza: “Formia 46!!!!#” } | Residenza non valida. | Il sistema chiede di reinserire la residenza. | \\ | \\ | **PASS** |

**RISULTATI DELLA TEST EXECUTION PER LA FUNZIONALITÀ “Evadi prenotazione”.**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Descrizione** | **Classi di equivalenza coperte** | **Pre-condizioni** | **Input** | **Output Attesi** | **Post-condizioni Attese** | **Output Ottenuti** | **Post-condizioni Ottenute** | **Esito**  **(*FAIL*, *PASS*)** |
| 1 | Tutti gli input validi. | Numero prenotazione valido, esito valido. | Il database è inizializzato correttamente. | {Numero prenotazione: “716”,Esito: “C” } | Prenotazione modificata. | Il magazziniere riceve un messaggio di conferma della modifica prenotazione. | \\ | \\ | **PASS** |
| 2 | Numero prenotazione stringa con simboli o caratteri. | Numero prenotazione stringa con caratteri [ERROR], esito valido. | Il magazziniere ha selezionato la  funzionalità di evadi prenotazione. Il  sistema richiede  al dipendente l’inserimento  del numero prenotazione. | {Numero prenotazione: “11aa@a”,  Esito: “C” } | Numero prenotazione non valido. | Il sistema chiede di reinserire il numero prenotazione. | \\ | \\ | **PASS** |
| 3 | Esito stringa > 1 carattere. | Esito stringa > 1 carattere [ERROR], numero prenotazione valido. | Il magazziniere ha selezionato la  funzionalità di evadi prenotazione. Il  sistema richiede  al dipendente l’inserimento  del carattere di esito. | {Numero prenotazione: “716”,  Esito: “CO” } | Esito troppo lungo. | Il sistema chiede di reinserire l’esito. | \\ | \\ | **PASS** |
| 4 | Esito senza caratteri ‘C’ o ‘R’. | Esito senza caratteri ‘C’ o ‘R’ [ERROR], numero prenotazione valido. | Il magazziniere ha selezionato la  funzionalità di evadi prenotazione. Il  sistema richiede  al dipendente l’inserimento  del carattere di esito. | {Numero prenotazione: “716”,  Esito: “NO” } | Esito non valido. | Il sistema chiede di reinserire l’esito. | \\ | \\ | **PASS** |

**RISULTATI DELLA TEST EXECUTION PER LA FUNZIONALITÀ “Evadi prenotazione”.**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Descrizione** | **Classi di equivalenza coperte** | **Pre-condizioni** | **Input** | **Output Attesi** | **Post-condizioni Attese** | **Output Ottenuti** | **Post-condizioni Ottenute** | **Esito**  **(*FAIL*, *PASS*)** |
| 1 | Tutti gli input validi. | Numero prenotazione valido. | Il database è inizializzato correttamente. | {Numero prenotazione: “716” } | Prenotazione modificata. | Il banco riceve un messaggio di conferma della modifica prenotazione. | \\ | \\ | **PASS** |
| 2 | Numero prenotazione stringa con simboli o caratteri. | Numero prenotazione stringa con caratteri [ERROR]. | Il magazziniere ha selezionato la  funzionalità di evadi prenotazione. Il  sistema richiede  al dipendente l’inserimento  del numero prenotazione. | {Numero prenotazione: “11aa@a” } | Numero prenotazione non valido. | Il sistema chiede di reinserire il numero prenotazione. | \\ | \\ | **PASS** |