# Contratti delle Operazioni

### **Contratti UC1**

|  |  |
| --- | --- |
| **Nome del contratto** | CO1: registraCliente |
| **Operazione** | registraCliente(nome : String, cognome : String, email : String, password : String, confermaPassword : String, altezza : String, peso : String) |
| **Riferimenti** | Caso d’uso UC1: Registrazione Cliente |
| **Pre-condizioni** |  |
| **Post-condizioni** | * È stata creata un’istanza *newCliente* di *Cliente* * Gli attributi di *newCliente* sono stati inizializzati tramite i dati inseriti dall’utente * *newCliente* è stato associato alla lista *listaUtenti* di *FitnessReady* |

|  |  |
| --- | --- |
| **Nome del contratto** | CO2: assegnaScheda |
| **Operazione** | assegnaScheda(cliente : Cliente) |
| **Riferimenti** | Caso d’uso UC1: Registrazione Cliente |
| **Pre-condizioni** |  |
| **Post-condizioni** | * È stata assegnata un’istanza *scheda* di *Scheda* all’istanza *cliente* di *Cliente* |

### **Contratti UC2**

|  |  |
| --- | --- |
| **Nome del contratto** | CO1: effettuaLogin |
| **Operazione** | effettuaLogin(email : String, password : String) |
| **Riferimenti** | Caso d’uso UC2: Login Utente |
| **Pre-condizioni** | * È presente in memoria un’istanza *Cliente* o *Amministratore* che richiede l’accesso al sistema |
| **Post-condizioni** | * È stata inizializzata un’istanza *utenteAttuale* di *Utente* con i relativi privilegi in base alle credenziali inserite dall’utente |

### **Contratti UC3**

|  |  |
| --- | --- |
| **Nome del contratto** | CO1: aggiungiAlCarrello |
| **Operazione** | aggiungiAlCarrello(quantity : int, price : float) |
| **Riferimenti** | Caso d’uso UC3: Effettua Acquisto |
| **Pre-condizioni** | * Il cliente deve essere autenticato * È presente un’istanza *carrello* di *Carrello* |
| **Post-condizioni** | * È stata creata un’istanza *riga* di *RigaCarrello* * Gli attributi di *riga* sono stati inizializzati tramite i dati sia ricavati dal prodotto selezionato che inseriti dal cliente * *riga* è stata associata alla lista *carrello.righeCarrello* |

|  |  |
| --- | --- |
| **Nome del contratto** | CO2: effettuaAcquisto |
| **Operazione** | effettuaAcquisto(metodoPagamento : String) |
| **Riferimenti** | Caso d’uso UC3: Effettua Acquisto |
| **Pre-condizioni** | * È presente un’istanza *carrello* di *Carrello* avente una lista *righeCarrello* non vuota |
| **Post-condizioni** | * È stata creata un’istanza *newAcquisto* di *Acquisto* * È stata creata un’istanza *rigaAcquisto* di *RigaAcquisto* per ogni elemento di *carrello.righeCarrello* * Una *rigaAcquisto* è stata associata alla lista *newAcquisto.righeAcquisto* per ogni elemento della lista *carrello.righeCarrello* |

|  |  |
| --- | --- |
| **Nome del contratto** | CO3: aggiungiAcquisto |
| **Operazione** | aggiungiAcquisto(newAcquisto : Acquisto, emailClienteAttuale : String, mappaAcquistiClienti : Map<String, List<Acquisto>>) |
| **Riferimenti** | Caso d’uso UC3: Effettua Acquisto |
| **Pre-condizioni** | * Dev’essere presente un’istanza *newAcquisto* di *Acquisto* |
| **Post-condizioni** | * *newAcquisto* è stato associato alla lista *mappaAcquistiClienti.get(emailClienteAttuale).listaAcquistiCliente* |

|  |  |
| --- | --- |
| **Nome del contratto** | CO4: effettuaOrdine |
| **Operazione** | effettuaOrdine(newAcquisto : Acquisto, dataEsecuzione : Date, dataArrivo : Date, indirizzo : String, città : String, cap : String, metodoPagamento : String, numCarta : String, cvv : String) |
| **Riferimenti** | Caso d’uso UC3: Effettua Acquisto |
| **Pre-condizioni** | * Dev’essere presente un’istanza *newAcquisto* di *Acquisto* da cui derivare l’ordine |
| **Post-condizioni** | * È stata creata un’istanza *newOrdine* di *Ordine* * È stata creata un’istanza *rigaOrdine* di *RigaOrdine* per ogni elemento della lista *newAcquisto.righeAcquisto* * Una *rigaOrdine* è stata associata alla lista *newOrdine.righeOrdine* per ogni elemento della lista *newAcquisto.righeAcquisto* |

|  |  |
| --- | --- |
| **Nome del contratto** | CO5: aggiungiOrdine |
| **Operazione** | aggiungiOrdine(emailClienteAttuale : String, newOrdine : Ordine) |
| **Riferimenti** | Caso d’uso UC3: Effettua Acquisto |
| **Pre-condizioni** | * Dev’essere presente un’istanza n*ewOrdine* di *Ordine* |
| **Post-condizioni** | * *newOrdine* è stato associato alla lista *mappaOrdiniClienti.get(emailClienteAttuale).listaOrdiniCliente* |

|  |  |
| --- | --- |
| **Nome del contratto** | CO6: aggiornaCatalogo |
| **Operazione** | aggiornaCatalogo(newAcquisto : Acquisto) |
| **Riferimenti** | Caso d’uso UC3: Effettua Acquisto |
| **Pre-condizioni** | * Dev’essere stata creata un’istanza *newAcquisto* di *Acquisto* |
| **Post-condizioni** | * Il parametro *disp* dei prodotti presenti nella lista *newAcquisto.righeAcquisto* viene aggiornato |

|  |  |
| --- | --- |
| **Nome del contratto** | CO7: buyAndUpdate |
| **Operazione** | buyAndUpdate(dataEsecuzione : Date, dataArrivo : Date, indirizzo : String, città : String, metodoPagamento : String, numCarta : String, cvv : String) |
| **Riferimenti** | Caso d’uso UC3: Effettua Acquisto |
| **Pre-condizioni** | * Dev’essere stato effettuato un corretto inserimento dei dati richiesti per l’acquisto |
| **Post-condizioni** | * *clienteAttuale.gettoni* è diventato *clienteAttuale.gettoni –* *carrello.totaleGettoni* (se *metodoPagamento = “Gettoni”*) * *clienteAttuale.gettoni* è diventato *clienteAttuale.gettoni + 1* (se *metodoPagamento = “Carta di credito”* e *carrello.totaleCarrello* è compreso tra 5€ e 30€) * *clienteAttuale.gettoni* è diventato *clienteAttuale.gettoni + 10* (*se metodoPagamento = “Carta di credito”* e *carrello.totaleCarrello > 30*) * *newAcquisto.totale* è diventato *carrello.totaleCarrello + 10* (se *metodoPagamento = “Carta di credito”* e *carrello.totaleCarrello < 35*) * *mappaAcquistiClienti* è stata aggiornata * l’istanza *gestoreAcquisto* (relativo all’acquisto in corso) è stata cancellata |

### **Contratti UC4**

|  |  |
| --- | --- |
| **Nome del contratto** | CO1: visualizzaAcquisti |
| **Operazione** | visualizzaAcquisti() |
| **Riferimenti** | Caso d’uso UC4: Visualizza Acquisti |
| **Pre-condizioni** |  |
| **Post-condizioni** | * Viene creata un’istanza *stringa* di *String*, contente tutti gli acquisti del cliente |

### **Contratti UC5**

|  |  |
| --- | --- |
| **Nome del contratto** | CO1: modificaOrdine |
| **Operazione** | modificaOrdine(idOrdine : int, emailClienteAttuale : String, indirizzo : String, città : String, cap : String) |
| **Riferimenti** | Caso d’uso UC5: Modifica Ordine |
| **Pre-condizioni** | * Dev’essere presente almeno un’istanza *ordine* di *Ordine* in *mappaOrdiniClienti.get(emailClienteAttuale).listaOrdiniCliente* * Dev’essere stato effettuato un corretto inserimento dei dati richiesti |
| **Post-condizioni** | * *ordine.indirizzo* è diventato *indirizzo* * *ordine.città* è diventato *città* * *ordine.cap* è diventato *cap* |

### **Contratti UC6**

|  |  |
| --- | --- |
| **Nome del contratto** | CO1: modificaDati |
| **Operazione** | modificaDati(nome : String, cognome : String, email : String, password : String, altezza : String, peso : String, obiettivo : String) |
| **Riferimenti** | Caso d’uso UC6: Modifica Dati |
| **Pre-condizioni** | * Dev’essere presente in memoria un’istanza *clienteAttuale* di *Cliente* * Dev’essere stato effettuato un corretto inserimento dei dati richiesti |
| **Post-condizioni** | * *clienteAttuale.nome* è diventato *nome* * *clienteAttuale.cognome* è diventato *cognome* * *clienteAttuale.email* è diventato *email* * *clienteAttuale.password* è diventato *password* * *clienteAttuale.altezza* è diventato *altezza* * *clienteAttuale.peso* è diventato *peso* * *clienteAttuale.obiettivo* è diventato *obiettivo* |

|  |  |
| --- | --- |
| **Nome del contratto** | CO2: assegnaScheda |
| **Operazione** | assegnaScheda(cliente : Cliente) |
| **Riferimenti** | Caso d’uso UC6: Modifica Dati |
| **Pre-condizioni** |  |
| **Post-condizioni** | * È stata assegnata un’istanza *scheda* di *Scheda* all’istanza *cliente* di *Cliente* |

### **Contratti UC7**

|  |  |
| --- | --- |
| **Nome del contratto** | CO1: getSchedaCliente |
| **Operazione** | getSchedaCliente(emailCliente : String) |
| **Riferimenti** | Caso d’uso UC7: Visualizza Scheda |
| **Pre-condizioni** | * Dev’essere presente in memoria un’istanza *clienteAttuale* di *Cliente* |
| **Post-condizioni** | * È stata recuperata un’istanza *scheda* di *Scheda* dalla mappa *mappaSchedeClienti* |

### **Contratti UC8**

|  |  |
| --- | --- |
| **Nome del contratto** | CO1: eseguiEsercizio |
| **Operazione** | eseguiEsercizio(i : int) |
| **Riferimenti** | Caso d’uso UC8: Esegui Esercizi |
| **Pre-condizioni** | * Dev’essere presente in memoria un’istanza *clienteAttuale* di *Cliente* |
| **Post-condizioni** | * È stata recuperata un’istanza *esercizio* di *Esercizio* * *esercizio.stato* è diventato *“ESEGUITO”* |

### **Contratti UC9**

|  |  |
| --- | --- |
| **Nome del contratto** | CO1: effettuaPrenotazione |
| **Operazione** | effettuaPrenotazione(cliente : Cliente, dataPrenotazione : Date, prod : Prodotto, quantità : int, indirizzo : String, città : String, cap : String, metodoPagamento : String, numCarta : String, cvv : String) |
| **Riferimenti** | Caso d’uso UC9: Gestisci Prenotazioni, CRUD |
| **Pre-condizioni** | * Dev’essere presente in memoria un’istanza *clienteAttuale* di *Cliente* * Dev’essere presente un’istanza *prodotto* di *Prodotto* con *prodotto.disp* pari 0 |
| **Post-condizioni** | * È stata creata un’istanza *newPrenotazione* di *Prenotazione* * *newPrenotazione* è stata associata alla lista *mappaPrenotazioniClienti.listaPrenotazioniCliente* |

|  |  |
| --- | --- |
| **Nome del contratto** | CO1.1: getPrenotazione |
| **Operazione** | getPrenotazione(emailCliente : String, idPrenotazione : int) |
| **Riferimenti** | Caso d’uso UC9: Gestisci Prenotazioni, CRUD |
| **Pre-condizioni** | * Dev’essere presente almeno un’istanza *prenotazione* di *Prenotazione* nella lista *mappaPrenotazioniClienti.listaPrenotazioniCliente* |
| **Post-condizioni** | * È stata recuperata un’istanza *prenotazione* di *Prenotazione*. |

|  |  |
| --- | --- |
| **Nome del contratto** | CO1.2: modificaPrenotazione |
| **Operazione** | modificaPrenotazione(oldPrenotazione : Prenotazione, cliente : Cliente, quantità : int, indirizzo : String, città : String, cap : String, metodoPagamento : String, numCarta : String, cvv : String) |
| **Riferimenti** | Caso d’uso UC9: Gestisci Prenotazioni, CRUD |
| **Pre-condizioni** | * Dev’essere presente almeno un’istanza *prenotazione* di *Prenotazione* nella lista *mappaPrenotazioniClienti.listaPrenotazioniCliente* |
| **Post-condizioni** | * *prenotazione.quantità* è diventato *quantità* * *prenotazione.indirizzo* è diventato *indirizzo* * *prenotazione.città* è diventato *città* * *prenotazione.cap* è diventato *cap* * *prenotazione.metodoPagamento* è diventato *metodoPagamento* * *prenotazione.totale* è diventato *totale* * *prenotazione.totalGettoni* è diventato *totaleGettoni* * *prenotazione.numCarta* è diventato *numCarta* * *prenotazione.cvv* è diventiato *cvv* |

|  |  |
| --- | --- |
| **Nome del contratto** | CO1.3: eliminaPrenotazione |
| **Operazione** | eliminaPrenotazion(cliente : Cliente, idPrenotazione : int) |
| **Riferimenti** | Caso d’uso UC9: Gestisci Prenotazioni, CRUD |
| **Pre-condizioni** | * Dev’essere presente almeno un’istanza *prenotazione* di *Prenotazione* nella lista *mappaPrenotazioniClienti.listaPrenotazioniCliente* |
| **Post-condizioni** | * È stata cancellata un’istanza *prenotazione* di *Prenotazione* dalla lista *mappaPrenotazioniClienti.listaPrenotazioniCliente* |