|  |
| --- |
| **Università degli Studi di Salerno Corso di Ingegneria del Software** |

**Nome Progetto: Gametop  
Titolo Documento: Test Case Specification DAO**

****

**Test Case Specification DAO**

Manager:

|  |
| --- |
| **Professori** |
| Prof. De Lucia Andrea |

Partecipanti:

|  |  |
| --- | --- |
| **Nome** | **Matricola** |
| Vincenzo Tarantino | 0512108025 |
| Luigi Sodano | 0512107590 |

# Revision History:

|  |  |  |  |
| --- | --- | --- | --- |
| **Data** | **Versione** | **Descrizione** | **Autore** |
| 18/10/2023 | 1.0 | Prima stesura del document problem statement | Team members |
| 30/10/2023 | 2.0 | Stesura del document RAD | Team members |
| 15/11/2023 | 3.0 | Revisione del document RAD | Team members |
| 03/01/2024 | 3.1 | Ultima revisione del document | Team members |
| 20/11/2023 | 3.2 | Stesura document System Design | Team members |
| 23/01/2024 | 3.3 | Revisione document System Design | Team members |
| 03/12/2023 | 4 | Stesura Object design document | Team members |
| 23/01/2024 | 4.1 | Ultima revisione del Object design document | Team members |
| 16/12/2023 | 5 | Stesura Test Plan e Test Plan DAO | Team members |
| 23/01/2024 | 5.1 | Ultima revisione del Test Plan e Test Plan DAO | Team members |
| 02/01/2024 | 6 | Stesura Test Case Specidication Test Case Specification DAO | Team members |

1. **Pacchetto GestioneAccountTest**
   1. **Test Gestione Personale**

|  |  |
| --- | --- |
| Identificativo : | TestFindPersonaleByEmailExisting() |
| Codice: | 1.1.1 |
| Dati necessari | PersonaleBean personaleEsistente |
| Output | L’account “personale”(Gestore ordini, gestore prodotti o ruoli) essendo esistente nel database restituisce un PersonaleBean che coincide con la tupla avente email uguale a quelle del personale passatagli come parametro. |
| Oracolo | L’istanza che l’oggetto restituisce e uguale a quella dell’oggetto passata come input. |

|  |  |
| --- | --- |
| Identificativo : | TestDeletePersonale() |
| Codice : | 1.1.2 |
| Dati necessari | PersonaleBean personaleEsistente |
| Output | L’account “personale” (Gestore ordini, gestore prodotti o ruoli) è esistente nel database, lo si elimina ed il test ha successo se non trovo nessun account che ha come email quella dell’account eliminato |
| Oracolo | L’istanza dell’oggetto passata come parametro verrà eliminata dalla tabella Personale del data base. |

|  |  |
| --- | --- |
| Identificativo : | TestUpdatePersonale() |
| Codice : | 1.1.3 |
| Dati necessari | PersonaleBean personaleEsistente |
| Output | Visto che il personale è esistente passandogli come parametro un personaleEsistente con i valori modificati aggiornerà la tupla nel data base avente email uguale a quella passata tramite personaleEsistente. |
| Oracolo | L’istanza dell’oggetto che restituisce deve essere uguale all’oggetto passato come parametro con le modifiche. |

|  |  |
| --- | --- |
| Identificativo : | TestUpdatePersonaleNonEsistente() |
| Codice : | 1.1.4 |
| Dati necessari | PersonaleBean personaleNonEsistente |
| Output | Visto che l’account “personale” non è esistente, facendo l’update di un personaleNonEsistente non trovero la tupla in Personale da aggiornare in quanto non esiste e ritorna false |
| Oracolo | L’istanza dell’oggetto che restituisce deve essere diversa all’oggetto passato come parametro con le modifiche e ritornerà false perché no lo trova. |

|  |  |
| --- | --- |
| Identificativo : | TestDeletePersonaleNonEsistente() |
| Codice : | 1.1.5 |
| Dati necessari | PersonaleBean personaleNonEsistente |
| Output | L’account “personale”(Gestore ordini, gestore prodotti o ruoli) essendo non esistente nel database non permette di risalire alla tupla avente email uguale a quelle del personale passatagli come parametro in questo modo il data base rimane invariato senza eliminare nessun personale |
| Oracolo | L’istanza che l’oggetto passata come parametro non verrà eliminata dalla tabella personale del data base e per tanto il risultato sarà false. |

|  |  |
| --- | --- |
| Identificativo : | TestDeletePersonaleNull() |
| Codice : | 1.1.6 |
| Dati necessari | null |
| Output | Il personale(Gestore ordini, gestore prosotti o ruoli) essendo non esistente nel database non possibile risalire alla tupla avente email uguale a quelle del personale passatagli come parametro(null) in questo modo il data base rimane invariato senza eliminare nessun personale e lancera un errore |
| Oracolo | Viene generato un errore in quanto il metodo non può essere eseguito perché non abbiamo fornito l’email null (Null Pointer Exception) |

|  |  |
| --- | --- |
| Identificativo : | TestFindTuttoPersonale() |
| Codice : | 1.1.7 |
| Dati necessari | null |
| Output | Restituira tutti i personali all’interno della tabella personale |
| Oracolo | Restituirà una collection di PersonaleBean all’interno della tabella personale |

|  |  |
| --- | --- |
| Identificativo : | TestFindPersonaleByEmailNull() |
| Codice : | 1.1.8 |
| Dati necessari | null |
| Output | Il personale(Gestore ordini, gestore prosotti o ruoli) essendo uguale a null non è possibile risalire alla tupla avente email uguale a quelle del personale passatagli come parametro. |
| Oracolo | L’istanza che l’oggetto restituisce e uguale a un oggetto con parametri nulli |

|  |  |
| --- | --- |
| Identificativo : | TestFindPersonaleByEmailNonEsistente() |
| Codice : | 1.1.9 |
| Dati necessari | PersonaleBean personaleNonEsistente |
| Output | Il personale(Gestore ordini, gestore prosotti o ruoli) non essendo esistente nel data base, non è possibile risalire alla tupla avente email uguale a quelle del personale passatagli come parametro. |
| Oracolo | L’istanza che l’oggetto restituisce e uguale a un oggetto con parametri nulli |

|  |  |
| --- | --- |
| Identificativo : | TestUpdatePersonaleNull() |
| Codice : | 1.1.10 |
| Dati necessari | PersonaleBean personaleNonEsistente |
| Output | Il personale(Gestore ordini, gestore prosotti o ruoli) non essendo esistente nel data base, non è possibile risalire alla tupla avente email uguale a quelle del personale passatagli come parametro. Quindi il data base rimarrà invariato |
| Oracolo | L’istanza che l’oggetto restituisce e uguale a un oggetto con parametri nulli, e per tanto restituirà false perché non è riuscito a completare l’aggiornamento. |

**1.2) TestRegistrazione**

|  |  |
| --- | --- |
| Identificativo : | TestVerificaPresenzaUtenteNonEsistente () |
| Codice : | 1.2.1 |
| Dati necessari | UtenteBean utenteNonEsistente |
| Output | L’utente essendo non esistente nel database restituisce un UtenteBean che ha tutti i suoi campi a null, visto che non trova nessuna tupla con l’email passata come parametro. |
| Oracolo | L’istanza che l’oggetto restituisce non ha nessun dato al suoi interno in quanto non è stata trovata nessuna assoiazione alla sua email. (untenteNonEsistente). |

|  |  |
| --- | --- |
| Identificativo : | TestTrovaUtenteEsistente() |
| Codice : | 1.2.2 |
| Dati necessari | UtenteBean utenteEsistente |
| Output | Visto che l’utente si trova nel data base verrà restituita l’istanza dell’oggetto passata come input con tutti i suoi dati. |
| Oracolo | L’istanza che l’oggetto restituisce deve essere uguale a quella dell’oggetto passato come input con tutti i suoi dati |

|  |  |
| --- | --- |
| Identificativo : | TestTrovaUtenteNull() |
| Codice : | 1.2.3 |
| Dati necessari | null |
| Output | Visto che l’utente passatogli e uguale a null non potrà restituire nessuno utente. |
| Oracolo | Viene generato un errore in quanto il metodo non può essere eseguito perché non abbiamo fornito l’email null Pointer exception |

|  |  |
| --- | --- |
| Identificativo : | TestInserisciNuovoUtente() |
| Codice : | 1.2.4 |
| Dati necessari | UtenteBean utenteNonEsistente |
| Output | Visto che l’utente non esistente non è ancora presente nel data base verrà inserito all’interno della tabella clienti con tutti i suoi dati |
| Oracolo | L’istanza dell’oggetto che andiamo a cercare deve essere uguale a quella dell’oggetto passatogli come parametro(untenteNonEsistente). |

|  |  |
| --- | --- |
| Identificativo : | TestInserisciNuovoUtenteNull() |
| Codice : | 1.2.5 |
| Dati necessari | null |
| Output | Visto che l’utente che gli passiamo è null non riesce ad inserirlo |
| Oracolo | Viene generato un errore in quanto il metodo non può essere eseguito perché non abbiamo fornito l’email.(null pointer exception) |

|  |  |
| --- | --- |
| Identificativo : | TestInserisciNuovoUtenteEsistente() |
| Codice : | 1.2.6 |
| Dati necessari | UtenteBean utenteEsistente |
| Output | Visto che l’utente esistente è ancora presente nel data base non verrà inserito all’interno della tabella clienti |
| Oracolo | Viene generato un errore sql di chiave duplicata. |

|  |  |
| --- | --- |
| Identificativo : | TestModificaPasswordUtenteEsistente() |
| Codice : | 1.2.7 |
| Dati necessari | UtenteBean utenteEsistente, String nuovaPsssword |
| Output | Visto che l’utente è esistente passandogli il come parametro utenteEsistente con password modificata con nuova password l’aggiornamento aggiornerà la tupla nel data base |
| Oracolo | L’istanza dell’oggetto che restituisce deve essere uguale all’oggetto passato come parametro con la password cambiata. |

|  |  |
| --- | --- |
| Identificativo : | TestModificaPasswordUtenteEsistenteNull() |
| Codice : | 1.2.8 |
| Dati necessari | null |
| Output | Visto che l’utente è esistente passandogli il come parametro nullo non aggiornerà nessuna tupla nel data base. |
| Oracolo | Viene generato un errore in quanto il metodo non può essere eseguito(null pointer exception) |

|  |  |
| --- | --- |
| Identificativo : | TestModificaPasswordUtenteNonEsistente() |
| Codice : | 1.2.9 |
| Dati necessari | UtenteBean utenteNonEsistente, String nuovaPassword |
| Output | Visto che l’utente non esistente passandogli il come parametro utente non Esistente e modificandogli la password con nuova password l’aggiornamento non aggiornerà la tupla nel data base perchè non trova il cliente |
| Oracolo | L’istanza dell’oggetto che restituisce deve essere diversa all’oggetto passato come parametro cambiata di passsword. |

|  |  |
| --- | --- |
| Identificativo : | TestModificaUtenteEsistente() |
| Codice : | 1.2.10 |
| Dati necessari | UtenteBean utenteEsistente |
| Output | Visto che l’utente è esistente passandogli come parametro utente Esistente con i dati modificati, l’aggiornamento aggiornerà la tupla nel data base relativa a quel cliente. |
| Oracolo | L’istanza dell’oggetto che restituisce deve essere uguale all’oggetto passato come parametro ovvero appena modificato. |

|  |  |
| --- | --- |
| Identificativo : | TestModificaUtenteNull () |
| Codice : | 1.2.11 |
| Dati necessari | null |
| Output | Visto che l’utente passatogli è null passandogli il parametro null l’aggiornamento non aggiornerà la tupla nel data base perche non trova il cliente in quanto null |
| Oracolo | Viene generato un errore in quanto il metodo non può essere eseguito(null pointer exception) |

|  |  |
| --- | --- |
| Identificativo : | TestModificaUtenteNonEsistente() |
| Codice : | 1.2.12 |
| Dati necessari | UtenteBean utenteNonEsistente |
| Output | Visto che l’utente non esistente passandogli come parametro un utente non Esistente con i dati modificati l’aggiornamento non aggiornerà la tupla nel data base perchè non trova il cliente visto che non e presente e non vi è nessuna tupla con quella email |
| Oracolo | L’istanza dell’oggetto che restituisce deve essere diversa all’oggetto passato come parametro in quanto non lo trova. |

|  |  |
| --- | --- |
| Identificativo : | TestTrovaTuttiUtenti () |
| Codice : | 1.2.13 |
| Dati necessari | null |
| Output | Vengono restituiti tutti i clienti all’interno della tabella Clienti. |
| Oracolo | Vengono restituiti una Collection tutti i clienti all’interno della tabella Clienti. |

**1.3) TestLogin**

|  |  |
| --- | --- |
| Identificativo : | TestTrovaUtenteEsistente() |
| Codice : | 1.3.1 |
| Dati necessari | UtenteBean UtenteEsistente |
| Output | Visto che l’utente è presente, verrà restituito l’utente con i sui dati che coincidono alla tupla con l’email dell’utente passatogli come parametro. |
| Oracolo | Viene restituita una istanza di utente uguale a quella passatagli come parametro. |

|  |  |
| --- | --- |
| Identificativo : | TestTrovaUtenteNonEsistente() |
| Codice : | 1.3.2 |
| Dati necessari | UtenteBean UtenteNonEsistente |
| Output | Visto che l’utente non è presente, verrà restituito l’utente con i sui dati null in quanto non coincidono alla tupla con l’email dell’utente passatogli come parametro. |
| Oracolo | Viene restituita una istanza di utente diversa da quella passatagli come parametro. |

|  |  |
| --- | --- |
| Identificativo : | TestTrovaUtenteNull () |
| Codice : | 1.3.3 |
| Dati necessari | null |
| Output | Visto che l’utente passatogli è null, non possiamo risalire alla sua presenza nel database. |
| Oracolo | Viene generato un errore in quanto il metodo non può essere eseguito(null pointer exception). |

|  |  |
| --- | --- |
| Identificativo : | TestTrovaPersonaleEsistente() |
| Codice : | 1.3.4 |
| Dati necessari | PersonaleBean PersonleEsistente |
| Output | Visto che il personale è presente, verrà restituito il personale con i sui dati che coincidono alla tupla con l’email del personale passatogli come parametro. |
| Oracolo | Viene restituita una istanza di personale uguale a quella passatagli come parametro. |

|  |  |
| --- | --- |
| Identificativo : | TestTrovaPersonaleNonEsistente() |
| Codice : | 1.3.5 |
| Dati necessari | PersonaleBean PersonleNonEsistente |
| Output | Visto che il personale non è presente, verrà restituito il personale con i sui dati null in quanto non coincidono alla tupla con l’email del personle passatogli come parametro. |
| Oracolo | Viene restituita una istanza di personale diversa da quella passatagli come parametro. |

|  |  |
| --- | --- |
| Identificativo : | TestTrovaPersonaleNull () |
| Codice : | 1.3.6 |
| Dati necessari | null |
| Output | Visto che il personale passatogli è null, non possiamo risalire alla sua presenza nel database. |
| Oracolo | Viene generato un errore in quanto il metodo non può essere eseguito (null pointer exception). |

**Pacchetto GestioneAcquistiTest**

**TestGestioneOrdini**

|  |  |
| --- | --- |
| Identificativo : | TestFindOrdineByNumeroOrdine() |
| Codice : | 2.1.1 |
| Dati necessari | OrdineBean ordineEsistente |
| Output | Viene restituita un’istanza di OrdineBean che contiene i dati di un ordine presente nella tabella Ordine del database che ha quella determinato numero ordine. |
| Oracolo | Viene restituita un’istanza di OrdineBean che contiene i dati di un ordine presente nella tabella Ordine del database che ha quella determinato numero ordine. |

|  |  |
| --- | --- |
| Identificativo : | TestFindOrdineByNumeroNonOrdine() |
| Codice : | 2.1.2 |
| Dati necessari | OrdineBean ordineNonEsistente |
| Output | Viene restituito un’istanza di OrdineBean con riferimento a null perché non esiste nessun ordine con quel numeroOrdine nel database. |
| Oracolo | Viene restituito un’istanza di OrdineBean con riferimento a tutti i parametri a null perché non esiste nessun ordine con quel numeroOrdine nel database. |

|  |  |
| --- | --- |
| Identificativo : | TestFindOrdineByNumeroNonOrdineNull() |
| Codice : | 2.1.3 |
| Dati necessari | null |
| Output | Viene restituito un’istanza di OrdineBean con riferimento a null perché non esiste nessun ordine con quel numeroOrdine nel database. |
| Oracolo | Viene generato un errore in quanto il metodo non può essere eseguito perché non abbiamo fornito il numero ordine dell’ordine che stiamo cercando. (Null Pointer Exception) |

|  |  |
| --- | --- |
| Identificativo : | TestFindTuttiOrdini() |
| Codice : | 2.1.4 |
| Dati necessari | null |
| Output | vengono restituita una collection di tutti gli ordini non ancora confermati all’interno della tabella ordini del data base |
| Oracolo | vengono restituita una collection di tutti gli ordini non ancora confermati all’interno della tabella ordini del data base |

|  |  |
| --- | --- |
| Identificativo : | TestUpdateOrdine() |
| Codice : | 2.1.5 |
| Dati necessari | OrdineBean ordineEsistenteDaModificare |
| Output | Vengono aggiornati i dati dell’ordine che ha quel determinato numero ordine dell’item specificata nell’oggetto item e ritorna true |
| Oracolo | Vengono aggiornati i dati dell’ordine che ha quel determinato numero ordine dell’item specificata nell’oggetto item e ritorna true |

|  |  |
| --- | --- |
| Identificativo : | TestUpdateOrdineNonEsistente() |
| Codice : | 2.1.6 |
| Dati necessari | OrdineBean ordineNonEsistenteDaModificare |
| Output | Ritorna false in quanto stiamo cercando di aggiornare i dati di un ordine che NON esistente nel database e quindi il database rimane invariato |
| Oracolo | Ritorna false in quanto stiamo cercando di aggiornare i dati di un ordine che NON esistente nel database e quindi il database rimane invariato perché non trova la tupal con quel numeroOrdine. |

|  |  |
| --- | --- |
| Identificativo : | TestUpdateOrdineNull() |
| Codice : | 2.1.7 |
| Dati necessari | null |
| Output | Viene generato un errore in quanto il metodo non può essere eseguito perché non abbiamo fornito il numero ordine del GestoreOrdiniBean che stiamo cercando. |
| Oracolo | Viene generato un errore in quanto il metodo non può essere eseguito perché non abbiamo fornito il numero ordine del GestoreOrdiniBean che stiamo cercando. (NullPointerException) |

|  |  |
| --- | --- |
| Identificativo : | TestDeleteOrdineEsistente() |
| Codice : | 2.1.8 |
| Dati necessari | GestoreOrdineBean ordineEsistente |
| Output | Viene eliminato l’ordine del Gestore Ordini che ha quella determinata email dell’item specificata nell’oggetto item e ritorna true. |
| Oracolo | Viene eliminato l’ordine del Gestore Ordini che ha quella determinata email dell’item specificata nell’oggetto item e ritorna true. |

|  |  |
| --- | --- |
| Identificativo : | TestUpdateOrdineNonEsistente() |
| Codice : | 2.1.9 |
| Dati necessari | OrdineBean ordineNonEsistenteDaModificare |
| Output | Ritorna false in quanto stiamo cercando di eliminare un ordine del Gestore Ordini che NON esistente nel database e quindi il database rimane invariato. |
| Oracolo | Ritorna false in quanto stiamo cercando di eliminare un ordine del Gestore Ordini che NON esistente nel database e quindi il database rimane invariato, perché non trova la tupal con quel numeroOrdine. |

|  |  |
| --- | --- |
| Identificativo : | TestUpdateOrdineNull() |
| Codice : | 2.1.10 |
| Dati necessari | null |
| Output | Viene generato un errore in quanto il metodo non può essere eseguito perché non abbiamo fornito il numero ordine dell’ordine che stiamo cercando. |
| Oracolo | Viene generato un errore in quanto il metodo non può essere eseguito perché non abbiamo fornito il numero ordine dell’ordine che stiamo cercando. (NullPointerException) |
|  |  |
| Identificativo : | TestConfermaOrdineEsistente() |
| Codice : | 2.1.11 |
| Dati necessari | OrdineBean ordineEsistente |
| Output | Viene confermato l’ordine che ha quel determinato numero ordine dell’item specificata nell’oggetto item e ritorna true |
| Oracolo | Viene confermato l’ordine che ha quel determinato numero ordine dell’item specificata nell’oggetto item e ritorna true |

|  |  |
| --- | --- |
| Identificativo : | TesConfermaOrdineNonEsistente() |
| Codice : | 2.1.12 |
| Dati necessari | OrdineBean ordineNonEsistente |
| Output | Ritorna un errore in quanto stiamo cercando di confermare l’ordine di un ordine che NON esistente nel database e quindi il database rimane invariato. |
| Oracolo | Ritorna un errore in quanto stiamo cercando di confermare l’ordine di un ordine che NON esistente nel database e quindi il database rimane invariato. |

|  |  |
| --- | --- |
| Identificativo : | TestConfermaOrdineNull() |
| Codice : | 2.1.13 |
| Dati necessari | null |
| Output | Viene generato un errore in quanto il metodo non può essere eseguito perché non abbiamo fornito il numero ordine dell’ OrdineBean che stiamo cercando. |
| Oracolo | Viene generato un errore in quanto il metodo non può essere eseguito perché non abbiamo fornito il numero ordine dell’ OrdineBean che stiamo cercando. (NullPointerException) |

**Pacchetto GestioneCarrelloTest**

**TestGestioneCarrello**

|  |  |
| --- | --- |
| Identificativo : | TestAggiungiAlCarrello() |
| Codice : | 3.1.1 |
| Dati necessari | SessionCarrelloBean carrello |
| Output | Viene inserita una nuova tupla nella tabella carrello del database contente i dati nel parametro item e restituisce true |
| Oracolo | Viene inserita una nuova tupla nella tabella carrello del database contente i dati nel parametro item e restituisce true. |

|  |  |
| --- | --- |
| Identificativo : | TestAggiungiAlCarrelloUtenteNonesistente() |
| Codice : | 3.1.2 |
| Dati necessari | SessionCarrelloBean carrello |
| Output | Non viene inserita una nuova tupla nella tabella carrello del database contente i dati nel parametro item perché non è presente l’utente a cui associare tale aggiunta e restituisce false. |
| Oracolo | Non viene inserita una nuova tupla nella tabella carrello del database contente i dati nel parametro item perché non è presente l’utente a cui associare tale aggiunta e restituisce false. |

|  |  |
| --- | --- |
| Identificativo : | TestEliminaTuttoIlCarrello() |
| Codice : | 3.1.3 |
| Dati necessari | SessionCarrelloBean carrello |
| Output | Viene eliminato il carello che ha quella determinata email dell’item specificata nell’oggetto item e ritorna true. |
| Oracolo | Viene eliminato il carello che ha quella determinata email dell’item specificata nell’oggetto item e ritorna ture. |

|  |  |
| --- | --- |
| Identificativo : | TestRitornaCarrello() |
| Codice : | 3.1.4 |
| Dati necessari | SessionCarrelloBean carrello |
| Output | vengono restituiti tutti i prodotti all’interno della tabella carrello del data base. (per recuperare il carrello). |
| Oracolo | vengono restituita una collection di tutti i prodotti all’interno della tabella carrello del data base. (per recuperare il carrello). |

|  |  |
| --- | --- |
| Identificativo : | TestRitornaCarrelloUtenteNonRegistrato() |
| Codice : | 3.1.5 |
| Dati necessari | SessionCarrelloBean carrello |
| Output | La collection che restituisce sarà vuota perché non è stata trovata nessuna email. |
| Oracolo | La collection che restituisce sarà vuota perché non è stata trovata nessuna email. |

|  |  |
| --- | --- |
| Identificativo : | TestRitornaCarrelloUtenteNonRegistrato() |
| Codice : | 3.1.5 |
| Dati necessari | null |
| Output | La collection che restituisce sarà nulla perché non è stata trovata nessuna email. |
| Oracolo | La collection che restituisce sarà nulla perché non è stata trovata nessuna email. |

**Pacchetto Gestione Prodotto**

**TestGestioneProdotti**

|  |  |
| --- | --- |
| Identificativo : | TestFindAllProductAccessori() |
| Codice : | 4.1.1 |
| Dati necessari | null |
| Output | vengono restituiti tutti i prodotti della categoria Accessori presenti all’interno della tabella prodotto del data base. |
| Oracolo | vengono restituita una collection di tutti i prodotti della categoria Accessori presenti all’interno della tabella prodotto del data base. |

|  |  |
| --- | --- |
| Identificativo : | TestFindAllProductConsole() |
| Codice : | 4.1.2 |
| Dati necessari | null |
| Output | vengono restituiti tutti i prodotti della categoria Console presenti all’interno della tabella prodotto del data base. |
| Oracolo | vengono restituita una collection di tutti i prodotti della categoria Console presenti all’interno della tabella prodotto del data base. |

|  |  |
| --- | --- |
| Identificativo : | TestFindAllProductVideogiochi() |
| Codice : | 4.1.3 |
| Dati necessari | null |
| Output | vengono restituiti tutti i prodotti della categoria Videogiochi presenti all’interno della tabella prodotto del data base. |
| Oracolo | vengono restituita una collection di tutti i prodotti della categoria Videogiochi presenti all’interno della tabella prodotto del data base. |

|  |  |
| --- | --- |
| Identificativo : | TestdoRetrieveByKeyPresente() |
| Codice : | 4.1.4 |
| Dati necessari | String codice |
| Output | Viene restituita un’istanza di ShopBean che contiene i dati di un prodotto presente nella tabella prodotto del database che ha quella determinato numero prodotto. |
| Oracolo | Viene restituita un’istanza di ShopBean che contiene i dati di un prodotto presente nella tabella prodotto del database che ha quella determinato numero prodotto. |

|  |  |
| --- | --- |
| Identificativo : | TestdoRetrieveByKeyNonPresente() |
| Codice : | 4.1.5 |
| Dati necessari | String codice |
| Output | Viene restituito un’istanza di ShopBean con riferimento ai dati a null perché non esiste nessun prodotto con quel codice prodotto nel database. |
| Oracolo | Viene restituito un’istanza di ShopBean con riferimento ai dati a null perché non esiste nessun prodotto con quel codice prodotto nel database. |

|  |  |
| --- | --- |
| Identificativo : | TestdoRetrieveByKeyNull() |
| Codice : | 4.1.6 |
| Dati necessari | String codice |
| Output | Viene restituito un’istanza di ShopBean con riferimento ai dati a null perché non esiste nessun prodotto con quel codice prodotto nel database. |
| Oracolo | Viene restituito un’istanza di ShopBean con riferimento ai dati a null perché non esiste nessun prodotto con quel codice prodotto nel database. |

|  |  |
| --- | --- |
| Identificativo : | TestFindAllProduct() |
| Codice : | 4.1.7 |
| Dati necessari | null |
| Output | vengono restituiti tutti i prodotti all’interno della tabella peodotto del data base. |
| Oracolo | vengono restituita una collection con tutti i prodotti all’interno della tabella prodotto del data base. |

|  |  |
| --- | --- |
| Identificativo : | TestDoUpdatePresente() |
| Codice : | 4.1.8 |
| Dati necessari | ShopBean expected |
| Output | Vengono aggiornati i dati del prodotto che ha quel determinato codice prodotto dell’item specificata nell’oggetto item e ritorna true. |
| Oracolo | Vengono aggiornati i dati del prodotto che ha quel determinato codice prodotto dell’item specificata nell’oggetto item e ritorna true. |

|  |  |
| --- | --- |
| Identificativo : | TestDoUpdateNonPresente() |
| Codice : | 4.1.9 |
| Dati necessari | ShopBean expected |
| Output | l’aggiornamento non va a buon fine quanto stiamo cercando di aggiornare i dati di un prodotto che NON è esistente nel database e quindi il database rimane invariato. |
| Oracolo | l’aggiornamento non va a buon fine quanto stiamo cercando di aggiornare i dati di un prodotto che NON esistente nel database e quindi il database rimane invariato. |

|  |  |
| --- | --- |
| Identificativo : | TestDoUpdateNull() |
| Codice : | 4.1.10 |
| Dati necessari | null |
| Output | l’aggiornamento non va a buon fine quanto stiamo cercando di aggiornare i dati di un prodotto che NON esistente nel database e quindi il database rimane invariato. |
| Oracolo | l’aggiornamento non va a buon fine quanto stiamo cercando di aggiornare i dati di un prodotto che NON è esistente(in quanto null) nel database e quindi il database rimane invariato. |

|  |  |
| --- | --- |
| Identificativo : | TestDoSave () |
| Codice : | 4.1.11 |
| Dati necessari | ShopBean prodotto |
| Output | Viene inserita una nuova tupla nella tabella prodotto del database contente i dati nel parametro item. |
| Oracolo | Viene inserita una nuova tupla nella tabella prodotto del database contente i dati nel parametro item. |

|  |  |
| --- | --- |
| Identificativo : | TestDoSaveEsistente() |
| Codice : | 4.1.12 |
| Dati necessari | ShopBean prodotto |
| Output | Viene generato un errore dovuto alla chiave primaria(codice prodotto) è già presente nel data base. |
| Oracolo | Viene generato un errore dovuto alla chiave primaria(codice prodotto) è già presente nel data base. .(Chiave duplicata) |

|  |  |
| --- | --- |
| Identificativo : | TestDoSaveNull() |
| Codice : | 4.1.13 |
| Dati necessari | ShopBean prodotto |
| Output | Viene generato un errore in quanto il metodo non può essere eseguito perché non abbiamo fornito il codice prodotto che stiamo cercando. |
| Oracolo | Viene generato un errore in quanto il metodo non può essere eseguito perché non abbiamo fornito il codice prodotto che stiamo cercando.(Null Pointer Exception) |

|  |  |
| --- | --- |
| Identificativo : | TestDoDeleteEsistente() |
| Codice : | 4.1.15 |
| Dati necessari | ShopBean prodotto |
| Output | Viene eliminata la tupla nel data base che ha quel determinato codice prodotto dell’item specificata nell’oggetto item. |
| Oracolo | Viene eliminata la tupla nel data base che ha quel determinato codice prodotto dell’item specificata nell’oggetto item. |

|  |  |
| --- | --- |
| Identificativo : | TestDoDeleteNull() |
| Codice : | 4.1.16 |
| Dati necessari | null |
| Output | Viene generato un errore in quanto il metodo non può essere eseguito perché non abbiamo fornito il codice prodotto che stiamo cercando. |
| Oracolo | Viene generato un errore in quanto il metodo non può essere eseguito perché non abbiamo fornito il codice prodotto che stiamo cercando. |

|  |  |
| --- | --- |
| Identificativo : | TestDoDeleteNonEsistente() |
| Codice: | 4.1.17 |
| Dati necessari | ShopBean prodottoNonEsistente |
| Output | l’eliminazione non va a buon fine in quanto stiamo cercando di eliminare un prodotto del che NON esistente nel database e quindi il database rimane invariato. |
| Oracolo | l’eliminazione non va a buon fine in quanto stiamo cercando di eliminare un prodotto del che NON esistente nel database e quindi il database rimane invariato. |