

**Nome Progetto: Gametop**  
**Titolo Documento: Test plan DAO**



# Test Plan 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

## 1) Pacchetto gestioneAccount

### 1.1 LoginModelDS

metodo:	doRetrieveByKeyPersonale(String email)
Parametri:	String email
Oggetti dell'ambiente	Database
Stringa Valida	SPD: stringa con ID presente nel database SNPD: stringa con ID NON presente nel database SV: stringa vuota
TEST FRAME	
SPD	ORACOLO: Viene restituita un'istanza di PersonaleBean che contiene i dati di un account personale presente nella tabella Personale del database che ha quella determinata email.
SNPD	ORACOLO: Viene restituito un'istanza di PersonaleBean con riferimento a null perché non esiste nessun account personale con quell'email nel database.
SV	ORACOLO: Viene generato un errore in quanto il metodo non può essere eseguito perché non abbiamo fornito l'email necessaria per la ricerca nel database di un account personale

metodo:	<b>doRetrieveByKeyUtente(String email)</b>
Parametri:	String email
Oggetti dell'ambiente	Database
Striga Valida	SPD: stringa con ID presente nel database SNPD: stringa con ID NON presente nel database SV: stringa vuota
TEST FRAME	
SPD	ORACOLO: Viene restituita un'istanza di UtenteBean che contiene i dati di un personale presente nella tabella Cliente del database che ha quella determinata email.
SNPD	ORACOLO: Viene restituito un'istanza di UtenteBean con riferimento a null perché non esiste nessun utente con quell'emal nel database.
SV	ORACOLO: Viene generato un errore in quanto il metodo non può essere eseguito perché non abbiamo fornito l'email necessaria per la ricerca nel database di un account utente

## 1.2 RegistrazioneModelDS

metodo:	<b>doSave(UtenteBean item)</b>
Parametri:	UtenteBean item
Oggetti dell'ambiente	Database
Item Valido	<ul style="list-style-type: none"> <li>• UDB: il nuovo Utente che stiamo salvando nel database non è ancora presente nella tabella clienti.</li> <li>• UPDB: il nuovo Utente che stiamo cercando di salvare nel database esiste già.</li> <li>• UNULL: il parametro item è null.</li> </ul>
TEST FRAME	
UDB	ORACOLO: Viene inserita una nuova tupla nella tabella Cliente del database contenente i dati nel parametro item.
UPDB	ORACOLO: Viene generato un errore dovuto alla chiave primaria(email) è già presente nel data base.
UNULL	ORACOLO: Viene generato un errore in quanto il metodo non può essere eseguito perché non abbiamo fornito l'email del personale che stiamo cercando

metodo:	<b>doUpdate(UtenteBean item)</b>
Parametri:	UtenteBean item
Oggetti dell'ambiente	Database
Item Valido	<ul style="list-style-type: none"> <li>• UNDB: L'utente di cui stiamo cercando di fare l'aggiornamento dei suoi dati NON è presente nel database.</li> <li>• UPDB: L'utente di cui stiamo cercando di fare l'aggiornamento dei suoi dati è presente nel database.</li> <li>• UNULL: il parametro item è null.</li> </ul>
TEST FRAME	
UNDB	ORACOLO: Ritorna false in quanto stiamo cercando di aggiornare i dati di un cliente che NON esiste nel database e quindi il database rimane invariato.
UPDB	ORACOLO: Vengono aggiornati i dati del cliente che ha quella determinata email dell'item specificata nell'oggetto item.
UNULL	ORACOLO: Viene generato un errore in quanto il metodo non può essere eseguito perché non abbiamo fornito l'email del personale che stiamo cercando.

metodo:	<b>doRetrieveByKey(String email)</b>
Parametri:	String email
Oggetti dell'ambiente	Database
Item Valido	SPD: stringa con ID presente nel database SNPD: stringa con ID NON presente nel database SV: stringa vuota
TEST FRAME	
UDB	ORACOLO: Viene restituita un'istanza di UtenteBean che contiene i dati di un utente presente nella tabella Cliente del database che ha quella determinata email.
UPDB	ORACOLO: Viene restituito un'istanza di UtenteBean con riferimento a null perché non esiste nessun utente con quell'email nel database.
UNULL	ORACOLO: Viene generato un errore in quanto il metodo non può essere eseguito perché non abbiamo fornito l'email necessaria per la ricerca nel database di un account utente

metodo:	<b>doUpdatePassword (UtenteBean item)</b>
Parametri:	UtenteBean item
Oggetti dell'ambiente	Database
Item Valido	<ul style="list-style-type: none"> <li>• UNDB: L'utente di cui stiamo cercando di fare l'aggiornamento dei suoi dati</li> </ul>

	NON è presente nel database. • UPDB: L'utente di cui stiamo cercando di fare l'aggiornamento dei suoi dati è presente nel database. • UNULL: il parametro item è null.
TEST FRAME	
UNDB	ORACOLO: Ritorna false in quanto stiamo cercando di aggiornare la password di un cliente che NON esistente nel database e quindi il database rimane invariato.
UPDB	ORACOLO: Viene aggiornata la password del cliente che ha quella determinata email dell'item specificata nell'oggetto item.
UNULL	ORACOLO: Viene generato un errore in quanto il metodo non può essere eseguito perché non abbiamo fornito l'email del cliente che stiamo cercando.

### 1.3 GestoreRuoliModelDS

metodo:	<b>doRetrieveByKey(String code)</b>
Parametri:	String code
Oggetti dell'ambiente	Database
Stringa Valida	SPD: stringa con ID presente nel database SNPD: stringa con ID NON presente nel database SV: stringa vuota
TEST FRAME	
SPD	ORACOLO: Viene restituita un'istanza di PersonaleBean che contiene i dati di un account personale presente nella tabella Personale del database che ha quella determinata email.
SNPD	ORACOLO: Viene restituito un'istanza di PersonaleBean con riferimento a null perché non esiste nessun account personale con quell'email nel database.
SV	ORACOLO: Viene generato un errore in quanto il metodo non può essere eseguito perché non abbiamo fornito l'email necessaria per la ricerca nel database di un account personale

metodo:	<b>stampaTuttoliPersonale()</b>
Parametri:	nessuno
Oggetti dell'ambiente	Database
<u>Risultati</u>	
TEST FRAME	

PDB	ORACOLO: vengono restituiti tutti gli account “Personale” all’interno della tabella Personale del data base.
-----	--

metodo:	<b>InserisciPersonale(PersonaleBean item)</b>
Parametri:	Personale item
Oggetti dell’ambiente	Database
Item Valido	<ul style="list-style-type: none"> <li>• PDB: il nuovo Personale che stiamo salvando nel database non è ancora presente nella tabella clienti.</li> <li>• PPDB: il nuovo Personale che stiamo cercando di salvare nel database esiste già.</li> <li>• PNULL: il parametro item è null.</li> </ul>
<b>TEST FRAME</b>	
UDB	ORACOLO: Viene inserita una nuova tupla nella tabella Cliente del database contenente i dati nel parametro item.
UPDB	ORACOLO: Viene generato un errore dovuto alla chiave primaria(email) è già presente nel data base.
UNULL	ORACOLO: Viene generato un errore in quanto il metodo non può essere eseguito perché non abbiamo fornito l’email del personale che stiamo cercando.

metodo:	<b>doUpdate(PersonaleBean item)</b>
Parametri:	PersonaleBean item
Oggetti dell’ambiente	Database
Item Valido	<ul style="list-style-type: none"> <li>• UNDB: il PersonaleBean di cui stiamo cercando di fare l’aggiornamento dei suoi dati NON è presente nel database.</li> <li>• UPDB: il PersonaleBean di cui stiamo cercando di fare l’aggiornamento dei suoi dati è presente nel database .</li> <li>• UNULL: il parametro item è null.</li> </ul>
<b>TEST FRAME</b>	
UNDB	ORACOLO: Ritorna false in quanto stiamo cercando di aggiornare i dati di un personale che NON esistente nel database e quindi il database rimane invariato.
UPDB	ORACOLO: Vengono aggiornati i dati del personale che ha quella determinata email dell’item specificata nell’oggetto item.
UNULL	ORACOLO: Viene generato un errore in quanto il metodo non può essere eseguito perché non abbiamo fornito l’email del personale che stiamo cercando.

metodo:	<b>doDelete(PersonaleBean item)</b>
Parametri:	PersonaleBean item
Oggetti dell’ambiente	Database
Item Valido	<ul style="list-style-type: none"> <li>• UNDB: il PersonaleBean di cui stiamo cercando di fare l’eliminazione NON è presente nel database.</li> <li>• UPDB: il PersonaleBean di cui stiamo cercando di fare l’eliminazione è</li> </ul>

	presente nel database . • UNULL: il parametro item è null.
<b>TEST FRAME</b>	
UNDB	ORACOLO: Ritorna false in quanto stiamo cercando di eliminare un personale che NON esistente nel database e quindi il database rimane invariato.
UPDB	ORACOLO: Viene eliminato il personale che ha quella determinata email dell'item specificata nell'oggetto item.
UNULL	ORACOLO: Viene generato un errore in quanto il metodo non può essere eseguito perché non abbiamo fornito l'email del personale che stiamo cercando.

## 2) Pacchetto Gestione Acquisti

### 2.1 GestioneOrdiniModelDS

metodo:	<b>ritornaTuttiGliOrdiniDaControllare()</b>
Parametri:	nessuno
Oggetti dell'ambiente	Database
Risultati	
<b>TEST FRAME</b>	
<u>ODB</u>	ORACOLO: vengono restituiti tutti gli ordini non ancora confermati all'interno della tabella ordini del database.

metodo:	<b>doRetrieveByKey(String code)</b>
Parametri:	String code
Oggetti dell'ambiente	Database
Striga Valida	SPD: stringa con ID presente nel database SNPD: stringa con ID NON presente nel database SV: stringa vuota
<b>TEST FRAME</b>	
SPD	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.
SNPD	ORACOLO: Viene restituito un'istanza di Ordine con riferimento a null perché non esiste nessun ordine con quel numeroOrdine nel database.
SV	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)

metodo:	<b>confermaOrdine(OrdineBean item)</b>
Parametri:	OrdineBean item
Oggetti dell'ambiente	Database
Item Valido	• ONDB: L'Ordine che stiamo cercando di confermare NON è presente nel database. • OPDB: L'Ordine cui stiamo cercando di confermare l'ordine è presente nel



	database . • ONULL: il parametro item è null.
TEST FRAME	
ONDB	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.
OPDB	ORACOLO: Viene confermato l'ordine che ha quel determinato numero ordine dell'item specificata nell'oggetto item e ritorna true.
ONULL	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.

metodo:	<b>doUpdate(OrdineBean item)</b>
Parametri:	OrdineBean item
Oggetti dell'ambiente	Database
Item Valido	<ul style="list-style-type: none"> <li>• ONDB: L'ordine di cui stiamo cercando di fare l'aggiornamento dei dati NON è presente nel database.</li> <li>• OPDB: L'ordine di cui stiamo cercando di fare l'aggiornamento dei dati è presente nel database .</li> <li>• ONULL: il parametro item è null.</li> </ul>
TEST FRAME	
ONDB	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.
OPDB	ORACOLO: Vengono aggiornati i dati dell'ordine che ha quel determinato numero ordine dell'item specificata nell'oggetto item e ritorna true.
ONULL	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.

metodo:	<b>doDelete(OrdineBean item)</b>
Parametri:	OrdineBean item
Oggetti dell'ambiente	Database
Item Valido	<ul style="list-style-type: none"> <li>• ONDB: L'OrdineBean di cui stiamo cercando di fare l'eliminazione NON è presente nel database.</li> <li>• OPDB: L'OrdineBean di cui stiamo cercando di fare l'eliminazione è presente nel database .</li> <li>• ONULL: il parametro item è null.</li> </ul>
TEST FRAME	
ONDB	ORACOLO: Ritorna false in quanto stiamo cercando di eliminare un ordine che NON esiste nel database e quindi il database rimane invariato.
OPDB	ORACOLO: Viene eliminato l'ordine i che ha quella determinata email dell'item specificata nell'oggetto item e ritorna true.
ONULL	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.

## 2.2 OrdineModelDS

3 metodo:	SalvaOrdine(GestoreOrdiniBean item)
Parametri:	Personale item
Oggetti dell'ambiente	Database
Item Valido	<ul style="list-style-type: none"> <li>• ONDB: il nuovo GestoreOrdiniBean che stiamo salvando nel database non è ancora presente nella tabella clienti.</li> <li>• OPDB: il nuovo GestoreOrdiniBean che stiamo cercando di salvare nel database esiste già.</li> <li>• ONULL: il parametro item è null.</li> </ul>
TEST FRAME	
ONDB	ORACOLO: Viene inserita una nuova tupla nella tabella GestoreOrdini del database contenente i dati nel parametro item.
ODB	ORACOLO: Viene generato un errore dovuto alla chiave primaria(numero ordine) è già presente nel data base.
ONULL	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.

## 3) Pacchetto Gestione Prodotti

### 3.1 ShopModelDS

metodo:	doRetrieveAllConsole()
Parametri:	nessuno
Oggetti dell'ambiente	Database
Riusultati	
TEST FRAME	
PDB	ORACOLO: vengono restituiti tutti i prodotti della categoria console presenti all'interno della tabella Prodotti del database.

metodo:	doRetrieveAllAccessori()
Parametri:	nessuno
Oggetti dell'ambiente	Database
Riusultati	
TEST FRAME	
PDB	ORACOLO: vengono restituiti tutti i prodotti della categoria accessori presenti all'interno della tabella Prodotti del data base.

metodo:	doRetrieveAllVideogiochi()
Parametri:	nessuno

Oggetti dell'ambiente	Database
Riusultati	
TEST FRAME	
PDB	ORACOLO: vengono restituiti tutti i prodotti della categoria videogiochi presenti all'interno della tabella Prodotti del data base.

metodo:	<b>doRetrieveByKey(String code)</b>
Parametri:	String code
Oggetti dell'ambiente	Database
Striga Valida	SPD: stringa con ID presente nel database SNPD: stringa con ID NON presente nel database SV: stringa vuota
TEST FRAME	
SPD	ORACOLO: Viene restituita un'istanza di ShopBean che contiene i dati di un prodotto presente nella tabella Prodotti del database che ha quella determinato numero prodotto.
SNPD	ORACOLO: Viene restituito un'istanza di ShopBean con riferimento a null perché non esiste nessun prodotto con quel codice prodotto nel database.
SV	ORACOLO: Viene generato un errore in quanto il metodo non può essere eseguito perché non abbiamo fornito il codice prodotto del prodotto che stiamo cercando

metodo:	<b>doRetrieveAllProduct()</b>
Parametri:	nessuno
Oggetti dell'ambiente	Database
Riusultati	
TEST FRAME	
PDB	ORACOLO: vengono restituiti tutti i prodotti all'interno della tabella vestiti del data base.

4 metodo:	<b>doSave(ShopBean item)</b>
Parametri:	ShopBean item
Oggetti dell'ambiente	Database
Item Valido	<ul style="list-style-type: none"> <li>• SNDB: il nuovo ShopBean che stiamo salvando nel database non è ancora presente nella tabella Prodotti.</li> <li>• SPDB: il nuovo ShopBean che stiamo cercando di salvare nel database esiste già.</li> <li>• SNULL: il parametro item è null.</li> </ul>
TEST FRAME	
SNDB	ORACOLO: Viene inserita una nuova tupla nella tabella vestiti del database contenente i dati nel parametro item.
SPDB	ORACOLO: Viene generato un errore dovuto alla chiave primaria(codice prodotto) è già presente nel data base.

SNULL	ORACOLO: Viene generato un errore in quanto il metodo non può essere eseguito perché non abbiamo fornito il codice vestito che stiamo cercando.
-------	---

metodo:	<b>doUpdate(ShopBean item)</b>
Parametri:	ShopBean item
Oggetti dell'ambiente	Database
Item Valido	<ul style="list-style-type: none"> <li>• ONDB: il ShopBean di cui stiamo cercando di fare l'aggiornamento dei suoi dati NON è presente nel database.</li> <li>• OPDB: il ShopBean di cui stiamo cercando di fare l'aggiornamento dei suoi dati è presente nel database .</li> <li>• ONULL: il parametro item è null.</li> </ul>
TEST FRAME	
ONDB	ORACOLO: l'aggiornamento non va a buon fine quanto stiamo cercando di aggiornare i dati di un vestito che NON esistente nel database e quindi il database rimane invariato.
OPDB	ORACOLO: Vengono aggiornati i dati del vstito che ha quel determinato codice vestito dell'item specificata nell'oggetto item e ritorna true.
ONULL	ORACOLO: Viene generato un errore in quanto il metodo non può essere eseguito perché non abbiamo fornito il codice vestito che stiamo cercando.

metodo:	<b><u>doDelete</u>(ShopBean item)</b>
Parametri:	ShopBean item
Oggetti dell'ambiente	Database
Item Valido	<ul style="list-style-type: none"> <li>• SNDB: il ShopBean di cui stiamo cercando di fare l'eliminazione NON è presente nel database.</li> <li>• SPDB: il ShopBean di cui stiamo cercando di fare l'eliminazione è presente nel database .</li> <li>• SNULL: il parametro item è null.</li> </ul>
TEST FRAME	
SNDB	ORACOLO: l'eliminazione non va a buon fine in quanto stiamo cercando di eliminare un vestito del che NON esistente nel database e quindi il database rimane invariato.
SPDB	ORACOLO: Viene eliminata la tupla nel data base che ha quel determinato codice vestito dell'item specificata nell'oggetto item.
SNULL	ORACOLO: Viene generato un errore in quanto il metodo non può essere eseguito perché non abbiamo fornito il codice vestito che stiamo cercando.

## 4) Pacchetto Gestione Carrello

### 4.1 GestioneCarrelloModelDS

5 metodo:	<b>doSave(SessionCarrelloBean item)</b>
Parametri:	SessionCarrelloBean item
Oggetti	Database

dell'ambiente	
Item Valido	<ul style="list-style-type: none"> <li>• SNDB: : L'utente che deve effettuare l'ordine e non è presente nel data base.</li> <li>• SPDB: L'utente che deve effettuare l'ordine è presente nel data base.</li> <li>• SNULL: il parametro item è null.</li> </ul>
TEST FRAME	
SNDB	ORACOLO: Non viene inserita una nuova tupla nella tabella carrello del database contenente i dati nel parametro item perché non è presente l'utente a cui associare tale aggiunta e restituisce false.
SPDB	ORACOLO: Viene inserita una nuova tupla nella tabella carrello del database contenente i dati nel parametro item e restituisce true.
SNULL	ORACOLO: Viene generato un errore in quanto il metodo non può essere eseguito perché non abbiamo fornito il codice vestito che stiamo cercando.

metodo:	<b>doDelete(SessionCarrelloBean item)</b>
Parametri:	SessionCarrelloBean item
Oggetti dell'ambiente	Database
Item Valido	<ul style="list-style-type: none"> <li>• SNDB: il SessionCarrelloBean di cui stiamo cercando di fare l'eliminazione NON è presente nel database.</li> <li>• SPDB: il SessionCarrelloBean di cui stiamo cercando di fare l'eliminazione è presente nel database .</li> <li>• SNULL: il parametro item è null.</li> </ul>
TEST FRAME	
SNDB	ORACOLO: Ritorna false in quanto stiamo cercando di eliminare un prodotto dal carrello che NON esistente nel database e quindi il database rimane invariato.
SPDB	ORACOLO: Viene eliminato il vestito dal carrello che ha quella determinata email e codice vestito dell'item specificata nell'oggetto item.
SNULL	ORACOLO: Viene generato un errore in quanto il metodo non può essere eseguito perché non abbiamo fornito il codice vestito e email che stiamo cercando.

metodo:	<b>trovaCarrello(String email)</b>
Parametri:	String email
Oggetti dell'ambiente	Database
Item Valido	<ul style="list-style-type: none"> <li>• SNDB: l'email di cui stiamo cercando di recuperare il carrello NON è presente nel database.</li> <li>• SPDB: l'email di cui stiamo cercando di recuperare il carrello presente nel database .</li> <li>• SNULL: il parametro item è null.</li> </ul>
TEST FRAME	
SNDB	ORACOLO: La collection che restituisce sarà vuota perché non è stata trovata nessuna email.
SPDB	ORACOLO: vengono restituiti tutti i prodotti all'interno della tabella carrello del data base. (per recuperare il carrello).
SNULL	ORACOLO: Viene generato un errore in quanto il metodo non può essere eseguito perché non abbiamo fornito l' email per cercare il nostro carrello.

metodo:	<b>deleteAll(String email)</b>
Parametri:	String email
Oggetti dell'ambiente	Database
Item Valido	<ul style="list-style-type: none"> <li>• SNDB: l'email di cui stiamo cercando di fare l'eliminazione NON è presente nel database.</li> <li>• SPDB: l'email di cui stiamo cercando di fare l'eliminazione è presente nel database .</li> <li>• SNULL: il parametro item è null.</li> </ul>
TEST FRAME	
SNDB	ORACOLO: Ritorna false in quanto stiamo cercando di eliminare un carrello che NON esistente nel database e quindi il database rimane invariato.
SPDB	ORACOLO: Viene eliminato il carello che ha quella determinata email dell'item specificata nell'oggetto item.
SNULL	ORACOLO: Viene generato un errore in quanto il metodo non può essere eseguito perché non abbiamo fornito l'email per eliminare il carrello.