

Winer Test Plan

Versione 1.0.1



Antica enoteca di Roma - Unsplash

Giovanni Prisco
051 21 06534

Indice

Indice.....	2
Revision History	3
1. Introduzione	4
1.1 Riferimenti.....	4
2. Funzionalità da testare	5
2.1 Criteri pass/fail testing.....	6
2.2 Approccio	6
2.3 Materiale per il testing	7
3. Test Cases.....	8
3.1 Autenticazione	8
3.2 Catalogo	10
3.3 Ordini	14
3.4 Carrello	16
3.5 Pagamenti.....	20
3.6 Note aggiuntive	21

Revision History

Data	Versione	Descrizione
01/02/2022	1.0.0	Creazione Test Plan Document
01/02/2022	1.0.0	Completamento Test Plan Document
08/02/2022	1.0.1	Fix al test case registrazione

1. Introduzione

Con il presente documento si intende fornire un'overview dal punto di vista manageriale del testing che bisogna effettuare sul sistema creato.

Andremo ad identificare gli elementi da testare e la metodologia di testing da utilizzare oltre agli strumenti necessari all'implementazione.

Lo scopo di questa fase di test sarà quello di identificare eventuali errori nella logica di business prima che il sistema possa essere consegnato al cliente e utilizzabile dai suoi utenti.

Si noti che la fase di testing è iterativa, durante la stesura dei test è ovviamente possibile fermarsi in caso di rilevamento di un problema, andare al codice per risolverlo e tornare alla scrittura dei test.

1.1 Riferimenti

In questo documento daremo per scontata la presa visione di tutti i documenti concernenti gli aspetti implementativi del sistema, a partire dal "*Problem Statement*" (PSD), passando per i documenti di requisiti e casi d'uso "*Requirements Analysis*" (RAD) e "*System Design*" (SDD) fino ad arrivare allo "*Object Design*" (ODD).

2. Funzionalità da testare

Andremo ad effettuare alcuni unit tests su componenti creati ad-hoc e alcuni test e2e per testare la completa integrazione di tutte le componenti del sistema.

Di seguito un elenco di funzionalità implementate nel sistema che devono essere testate.

Autenticazione

- Registrazione utente
- Login
- Controllo Token

Catalogo

- Visualizza Catalogo
- Aggiungi vino
- Modifica vino
- Elimina vino
- Mostra dettagli vino

Ordini

- Crea ordine

Carrello

- Aggiungi vino
- Modifica quantità
- Elimina vino
- Svuota carrello

Pagamenti

- Processa transazione

2.1 Criteri pass/fail testing

Il sistema implementato fornisce dati in risposta a degli input, lo scopo del testing è proprio quello di verificare che sulla base di determinati input, l'output fornito dal sistema sia giusto. Possiamo dire che il testing ha avuto esito positivo se l'output del test è diverso da quello atteso, in questa situazione possiamo dire di aver trovato una failure da correggere nel sistema, possiamo quindi rivedere il codice, correggerla e tornare a scrivere i test.

2.2 Approccio

Andremo a suddividere la fase di testing in 3 fasi:

- Test dei singoli componenti (Unit Tests)
- Test dei componenti nel loro insieme (Integration tests)
- Test di sistema

Per sviluppare i nostri unit tests, utilizzeremo la tecnica "BLACK-BOX".

Ci andremo quindi ad occupare solo ed esclusivamente dei dati di input ed output, senza curarci della struttura interna delle componenti.

Inoltre, data la mancanza di fattibilità nell'effettuare un test esaustivo che riesca a coprire per intero la quantità di dati di input, utilizzeremo la strategia "Category Partition", ovvero andremo a dividere i dati di input in categorie, ogni categoria sarà partizionata in "scelte". Al termine di questo processo andremo a combinare queste scelte in modo da poter creare un caso di test con la relativa istanza di input da testare. Utilizzando questa strategia otteniamo un insieme di test che utilizza diverse istanze di input per coprire la maggior parte dei casi in modo non ridondante.

Per quanto riguarda il testing di integrazione, si utilizzerà la tecnica “WHITE-BOX”. Con questa tecnica andremo a testare il funzionamento interno dei componenti, utilizzando dei dati che navigheranno nel percorso “sbagliato” del componente in modo da cercare eventuali errori. Chiamiamo questo percorso “cammino”, ovviamente non esiste un modo per testare ogni singolo cammino, ma possiamo trovare il giusto numero di cammini indipendenti tra loro che ci permetteranno di avere una copertura del codice massima.

2.3 Materiale per il testing

Saranno utilizzate le seguenti tecnologie per effettuare i test descritti sopra:

- Express Server
- MySQL
- Jest
- Client web per effettuare richieste alle risorse REST

3. Test Cases

3.1 Autenticazione

TC_Registrazione

Parametro: email

Categorie	Scelte
Struttura indirizzo email - SIE	<ol style="list-style-type: none"> 1. La stringa contiene un carattere "@" che separa due stringhe non vuote e almeno 2 parti di dominio dopo la "@" (...@firstpart.secondpart...) [property validSIEValue] 2. La stringa non è divisa in due parti da "@" o ha solo una parte di dominio [property invalidSIEValue]
Esistenza nel Database - ESDB	<ol style="list-style-type: none"> 1. L'indirizzo email è già presente nel database [if validSIEValue] [property invalidESDBValue] 2. L'indirizzo email non compare nel database [if validSIEValue] [property validESDBValue]

Parametro: password

Categorie	Scelte
Lunghezza password - LP	<ol style="list-style-type: none"> 1. La stringa ha lunghezza ≥ 8 [property validLP] 2. La stringa ha lunghezza < 8 [property invalidLP]
Struttura password - SP	<ol style="list-style-type: none"> 1. La stringa contiene almeno 1 lettera maiuscola, almeno 1 lettera minuscola, almeno 1 numero o 1 carattere speciale [if validLP] [property validSPValue] 2. La stringa non rispetta tutti i requisiti del punto (1.) [property invalidSPValue]

Codice	Combinazione	Esito
TC_RegistrazioneOK	SIE1.ESDB2.LP1.SP1	Registrazione avvenuta con successo
TC_RegistrazioneErrore1	SIE1.ESDB2.LP1.SP2	La password non rispetta i requisiti minimi
TC_RegistrazioneErrore2	SIE1.EDB2.LP2	La password non rispetta i requisiti minimi
TC_RegistrazioneErrore3	SIE1.EDB1	Email già esistente nel database

Codice	Combinazione	Esito
TC_RegistrazioneErrore4	SIE2	L'indirizzo email non è valido

TC_Login

Parametro: email-login

Categorie	Scelte
Esistenza nel Database - ESDB	<ol style="list-style-type: none"> 1. L'indirizzo email è presente nel database [property validESDBValue] 2. L'indirizzo email non compare nel database [property invalidESDBValue]

Parametro: password-login

Categorie	Scelte
Corrispondenza degli hash - CDH	<ol style="list-style-type: none"> 1. L'hash della password dell'utente trovato corrisponde con l'hash generato dal testo immesso dall'utente [property validCDHValue] 2. Gli hash non corrispondono [property invalidCDHValue]

Codice	Combinazione	Esito
TC_LoginOK	ESDB1.CDH1	Login effettuato con successo
TC_LoginErrore1	ESDB1.CDH2	Password errata
TC_LoginErrore2	ESDB2	Email errata

TC_ControlloToken

Parametro: email-login-1

Categorie	Scelte
Struttura JWT - SJWT	<ol style="list-style-type: none"> 1. Il token è una stringa divisa in 3 parti da un "." [property validSJWTValue] 2. Il token non è composto da 3 parti [property invalidSJWTValue]

Categorie	Scelte
Firma JWT - FJWT	<ol style="list-style-type: none"> 1. La terza parte del token contiene un HMAC valido che corrisponde ad header + payload firmati con un secret code [if validSJWT] [property validFJWTValue] 2. La firma del token non è valida [id validSJWT] [property invalidFJWT]

Codice	Combinazione	Esito
TC_ControlloTokenOK	SJWT1.FJWT1	Autorizzazione concessa
TC_ControlloTokenErrore1	SJWT1.FJWT2	Firma non valida
TC_ControlloTokenErrore2	SJWT2	Token non valido

3.2 Catalogo

TC_VisualizzaCatalogo

Parametro: nome-vino

Categorie	Scelte
Lunghezza nome vino - LNV	<ol style="list-style-type: none"> 1. Il nome del vino ha lunghezza ≥ 3 [property valid-LNVValue] 2. Il nome del vino ha lunghezza < 3 [property invalid-LNVValue]

Parametro: range-prezzo

Categorie	Scelte
Tipo range prezzo - RP	<ol style="list-style-type: none"> 1. Il range prezzo consiste di due numeri validi [property validRPValue] 2. Il range prezzo manca di validità in quanto a tipo [property invalidRPValue]
Valore range prezzo (min, max) - VRP	<ol style="list-style-type: none"> 1. $\text{max} \geq \text{min}$ [if validRPValue] [property validVRPValue] 2. $\text{max} < \text{min}$ [if validRPValue] [property invalidVRPValue]

Codice	Combinazione	Esito
TC_VisualizzaCatalogoOK	LVN1.RP1.VRP1	Visualizza Catalogo con successo
TC_VisualizzaCatalogoErrore1	LVN1.RP1.VRP2	Range prezzo non valido
TC_VisualizzaCatalogoErrore2	LVN1.RP2	Range prezzo non valido
TC_VisualizzaCatalogoErrore3	LVN2.*	Nome vino non valido

TC_AggiungiVino

Parametro: ID

Categorie	Scelte
Lunghezza nome vino - LV	<ol style="list-style-type: none"> Il nome del vino ha lunghezza ≥ 3 [property valid-LNVValue] Il nome del vino ha lunghezza < 3 [property invalid-LNVValue]
Tipo annata - TA	<ol style="list-style-type: none"> L'annata del vino è un numero [property validTAValue] L'annata del vino non è un numero [property invalid-TAValue]
Valore annata - VA	<ol style="list-style-type: none"> L'annata del vino è ≥ 0 [if validTAValue] [property validVAValue] L'annata del vino è < 0 [if validTAValue] [property invalidVAValue]
Esistenza Vino in Database - EVD	<ol style="list-style-type: none"> L'ID formato da vino-annata non esiste nel database [if validLNVValue and validTAValue and validVAValue] [property validEvdValue] Il vino esiste già nel database [if validLNVValue and validTAValue and validVAValue] [property invalidEvdValue]

Parametro: prezzo

Categorie	Scelte
Tipo prezzo - TP	<ol style="list-style-type: none"> Il prezzo del vino è un numero [property validTPValue] Il prezzo del vino non è un numero [property invalidTPValue]
Valore prezzo - VP	<ol style="list-style-type: none"> Il prezzo del vino è ≥ 0.01 [if validTPValue] [property validVPValue] L'annata del vino è < 0.01 [if validTPValue] [property invalidVPValue]

Parametro: disponibilità

Categorie	Scelte
Tipo disponibilità - TD	<ol style="list-style-type: none"> 1. Il numero di bottiglie del vino è un numero [property validTDValue] 2. Il numero di bottiglie del vino non è un numero [property invalidTDValue]
Valore disponibilità - VD	<ol style="list-style-type: none"> 1. Il numero di bottiglie del vino è ≥ 0 [if validTDValue] [property validVDValue] 2. L'anata del vino è < 0 [if validTDValue] [property invalidVDValue]

Parametro: ID Azienda

Categorie	Scelte
Tipo ID Azienda - TIDA	<ol style="list-style-type: none"> 1. L'ID Azienda del vino è un numero [property validTIDAValue] 2. L'ID Azienda del vino non è un numero [property invalidTIDAValue]
Esistenza Azienda in Database - EAD	<ol style="list-style-type: none"> 1. L'azienda del vino esiste nel database [if validTIDAValue] [property validEADValue] 2. L'azienda del vino non esiste nel database [if validTIDAValue] [property invalidEADValue]

Parametro: ID Famiglia

Categorie	Scelte
Tipo ID Famiglia - TIFA	<ol style="list-style-type: none"> 1. L'ID Famiglia del vino è un numero [property validTIFAValue] 2. L'ID Famiglia del vino non è un numero [property invalidTIFAValue]
Esistenza Famiglia in Database - EFD	<ol style="list-style-type: none"> 1. La famiglia del vino esiste nel database [if validTIDAValue] [property validEFDValue] 2. La famiglia del vino non esiste nel database [if validTIDAValue] [property invalidEFDValue]

Codice	Combinazione	Esito
TC_AggiungiVinoOK	LV1.TA1.VA1.EVD1.TP1.VP1.TD1.VD1.TIDA1.EAD1.TIFA1.EFD1	Vino aggiunto con successo
TC_AggiungiVinoErrore1	LV1.TA1.VA1.EVD1.TP1.VP1.TD1.VD1.TIDA1.EAD1.TIFA1.EFD2	La famiglia scelta non è valida
TC_AggiungiVinoErrore2	LV1.TA1.VA1.EVD1.TP1.VP1.TD1.VD1.TIDA1.EAD1.TIFA2	La famiglia scelta non è valida
TC_AggiungiVinoErrore3	LV1.TA1.VA1.EVD1.TP1.VP1.TD1.VD1.TIDA1.EAD2	L'azienda scelta non è valida

Codice	Combinazione	Esito
TC_AggiungiVinoErrore4	LV1.TA1.VA1.EVD1.TP1.VP1.TD1.VD1.TIDA2	L'azienda scelta non è valida
TC_AggiungiVinoErrore5	LV1.TA1.VA1.EVD1.TP1.VP1.TD1.VD2	Il numero di bottiglie disponibili non è valido
TC_AggiungiVinoErrore6	LV1.TA1.VA1.EVD1.TP1.VP1.TD2	Il numero di bottiglie disponibili non è valido
TC_AggiungiVinoErrore7	LV1.TA1.VA1.EVD1.TP1.VP2	Il prezzo scelto non è valido
TC_AggiungiVinoErrore8	LV1.TA1.VA1.EVD1.TP2	Il prezzo scelto non è valido
TC_AggiungiVinoErrore9	LV1.TA1.VA1.EVD2	La combinazione vino-annata è già presente nel database
TC_AggiungiVinoErrore10	LV1.TA1.VA2	L'annata non è valida
TC_AggiungiVinoErrore11	LV1.TA2	L'annata non è valida
TC_AggiungiVinoErrore12	LV2	Il nome del vino non è valido

TC_ModificaVino

Per il test case che riguarda la modifica del vino facciamo riferimento a **TC_AggiungiVino** con l'unica differenza che in questo caso (modifica) tutti i parametri sono opzionali, non sussiste quindi dipendenza nelle validazioni.

TC_EliminaVino

Parametro: ID Vino

Categorie	Scelte
Lunghezza nome vino - LV	<ol style="list-style-type: none"> 1. Il nome del vino ha lunghezza ≥ 3 [property valid-LNVValue] 2. Il nome del vino ha lunghezza < 3 [property invalid-LNVValue]
Tipo annata - TA	<ol style="list-style-type: none"> 1. L'annata del vino è un numero [property validTAValue] 2. L'annata del vino non è un numero [property invalid-TAValue]

Categorie	Scelte
Valore annata - VA	<ol style="list-style-type: none"> 1. L'annata del vino è ≥ 0 [if validTAValue] [property validVAValue] 2. L'annata del vino è < 0 [if validTAValue] [property invalidVAValue]
Esistenza Vino in Database - EVD	<ol style="list-style-type: none"> 1. L'ID formato da vino-annata esiste nel database [if validLNVValue and validTAValue and validVAValue] [property validEvdValue] 2. Il vino non esiste nel database [if validLNVValue and validTAValue and validVAValue] [property invalidEvdValue]

Codice	Combinazione	Esito
TC_EliminaVinoOK	LV1.TA1.VA1.EVD1	Vino aggiunto con successo
TC_EliminaVinoErrore1	LV1.TA1.VA1.EVD2	Vino non presente nel database
TC_EliminaVinoErrore2	LV1.TA1.VA2	Annata non valida
TC_EliminaVinoErrore3	LV1.TA2	Annata non valida
TC_EliminaVinoErrore4	LV2	Nome vino non valido

TC_MostraDettagliVino

Per testare l'operazione sui dettagli di un singolo vino utilizziamo le stesse caratteristiche del test **TC_EliminaVino** ovvero controlliamo che esista un vino che abbia per chiave primaria una combinazione valida di nome ed annata.

3.3 Ordini

TC_CreaOrdine

Parametro: ID Utente

Categorie	Scelte
Esistenza Utente nel Database - EUD	<ol style="list-style-type: none"> 1. Esiste un utente nel database che abbia l'id richiesto [property validEUDValue] 2. Non esiste un utente con l'id richiesto [property invalidEUDValue]

Parametro: Prezzo totale

Categorie	Scelte
Valore prezzo - VP	<ol style="list-style-type: none"> Il valore del prezzo è un numero ≥ 0.01 [property validVPValue] Il valore del prezzo è un numero < 0.01 [property invalidVPValue]

Parametro: ID vini + quantità

Categorie	Scelte
Valore ID Vini - VIDV	<ol style="list-style-type: none"> Il valore del parametro è una lista di id vini (nome-annata) [property validVIDVValue] La lista di ID vini contiene ID non validi [property invalidVIDVValue]
Esistenza Vini nel Database - EVD	<ol style="list-style-type: none"> Ogni ID della lista si riferisce ad un vino esistente nel db [if validVIDVValue] [property validEVDValue] Esiste un ID della lista che non punta ad un vino nel database [if validVIDVValue] [property invalidEVDValue]
Valore Quantità - VQ	<ol style="list-style-type: none"> La quantità specificata è un numero \leq wine.availability [if validEVDValue] [property validEVDValue] La quantità non è un numero valido o è $>$ wine.availability [if validEVDValue] [property invalidEVDValue]

Codice	Combinazione	Esito
TC_CreaOrdineOK	EUD1.VP1.VIDV1.EVD1.VQ1	Ordine creato con successo
TC_CreaOrdineErrore1	EUD1.VP1.VIDV1.EVD1.VQ2	La quantità specificata non è valida
TC_CreaOrdineErrore2	EUD1.VP1.VIDV1.EVD2	Vini selezionati non validi
TC_CreaOrdineErrore3	EUD1.VP1.VIDV2	Vini selezionati non validi
TC_CreaOrdineErrore4	EUD1.VP2	Prezzo non valido
TC_CreaOrdineErrore5	EUD2	L'utente non esiste nel database

3.4 Carrello

TC_AggiungiAlCarrello

Parametro: ID Vino Carrello

Categorie	Scelte
Lunghezza nome vino - LV	<ol style="list-style-type: none"> Il nome del vino ha lunghezza ≥ 3 [property valid-LNVValue] Il nome del vino ha lunghezza < 3 [property invalid-LNVValue]
Tipo annata - TA	<ol style="list-style-type: none"> L'annata del vino è un numero [property validTAValue] L'annata del vino non è un numero [property invalid-TAValue]
Valore annata - VA	<ol style="list-style-type: none"> L'annata del vino è ≥ 0 [if validTAValue] [property validVAValue] L'annata del vino è < 0 [if validTAValue] [property invalidVAValue]
Esistenza Vino in Database - EVD	<ol style="list-style-type: none"> L'ID formato da vino-annata esiste nel database [if validLNVValue and validTAValue and validVAValue] [property validEvdValue] Il vino non esiste nel database [if validLNVValue and validTAValue and validVAValue] [property invalidEvdValue]

Parametro: ID Utente Carrello

Categorie	Scelte
Esistenza Utente in Database - EUD	<ol style="list-style-type: none"> Esiste un utente nel database con l'id specificato [property validEUDValue] Non esiste un utente con l'id specificato [property invalidEUDValue]

Codice	Combinazione	Esito
TC_AggiungiAlCarrelloOK	LV1.TA1.VA1.EVD1.EUD1	Vino aggiunto al carrello
TC_AggiungiAlCarrelloErrore1	LV1.TA1.VA1.EVD1.EUD2	Utente non valido
TC_AggiungiAlCarrelloErrore2	LV1.TA1.VA1.EVD2	Il vino non esiste
TC_AggiungiAlCarrelloErrore3	LV1.TA1.VA2	Annata non valida

Codice	Combinazione	Esito
TC_AggiungiAlCarrelloErrore4	LV1.TA2	Annata non valida
TC_AggiungiAlCarrelloErrore5	LV2	Nome vino non valido

TC_ModificaQuantità

Parametro: ID Utente Carrello (Modifica)

Categorie	Scelte
Esistenza Utente in Database - EUD	<ol style="list-style-type: none"> Esiste un utente nel database con l'id specificato [property validEUDValue] Non esiste un utente con l'id specificato [property invalidEUDValue]

Parametro: ID Vino Carrello (Modifica)

Categorie	Scelte
Lunghezza nome vino - LV	<ol style="list-style-type: none"> Il nome del vino ha lunghezza ≥ 3 [property validLNVValue] Il nome del vino ha lunghezza < 3 [property invalidLNVValue]
Tipo annata - TA	<ol style="list-style-type: none"> L'annata del vino è un numero [property validTAValue] L'annata del vino non è un numero [property invalidTAValue]
Valore annata - VA	<ol style="list-style-type: none"> L'annata del vino è ≥ 0 [if validTAValue] [property validVAValue] L'annata del vino è < 0 [if validTAValue] [property invalidVAValue]
Esistenza Vino in Database - EVD	<ol style="list-style-type: none"> L'ID formato da vino-annata esiste nel database [if validLNVValue and validTAValue and validVAValue] [property validEvdValue] Il vino non esiste nel database [if validLNVValue and validTAValue and validVAValue] [property invalidEvdValue]

Parametro: Quantità

Categorie	Scelte
Valore Quantità - VQ	<ol style="list-style-type: none"> Il valore della quantità è > 0 e $< \text{wine.availability}$ [property validVQValue] Il valore della quantità è ≤ 0 o $> \text{wine.availability}$ [property invalidVQValue]

Codice	Combinazione	Esito
TC_ModificaQuantitàOK	EUD1.LV1.TA1.VA1.EVD1.VQ1	Quantità modificata correttamente
TC_ModificaQuantitàErrore1	EUD1.LV1.TA1.VA1.EVD1.VQ2	Valore quantità non valido
TC_ModificaQuantitàErrore2	EUD1.LV1.TA1.VA1.EVD2	Il vino non esiste
TC_ModificaQuantitàErrore3	EUD1.LV1.TA1.VA2	Annata non valida
TC_ModificaQuantitàErrore4	EUD1.LV1.TA2	Annata non valida
TC_ModificaQuantitàErrore5	EUD1.LV2	Nome vino non valido
TC_ModificaQuantitàErrore6	EUD2	Utente non valido

TC_EliminaVino

Parametro: ID Utente Carrello (Elimina)

Categorie	Scelte
Esistenza Utente in Database - EUD	<ol style="list-style-type: none"> Esiste un utente nel database con l'id specificato [property validEUDValue] Non esiste un utente con l'id specificato [property invalidEUDValue]

Parametro: ID Vino Carrello (Elimina)

Categorie	Scelte
Lunghezza nome vino - LV	<ol style="list-style-type: none"> Il nome del vino ha lunghezza ≥ 3 [property validLNVValue] Il nome del vino ha lunghezza < 3 [property invalidLNVValue]
Tipo annata - TA	<ol style="list-style-type: none"> L'annata del vino è un numero [property validTAValue] L'annata del vino non è un numero [property invalidTAValue]
Valore annata - VA	<ol style="list-style-type: none"> L'annata del vino è ≥ 0 [if validTAValue] [property validVAValue] L'annata del vino è < 0 [if validTAValue] [property invalidVAValue]

Categorie	Scelte
Esistenza Vino in Database - EVD	<ol style="list-style-type: none"> 1. L'ID formato da vino-annata esiste nel database [if validLNValue and validTAValue and validVAValue] [property validEvdValue] 2. Il vino non esiste nel database [if validLNValue and validTAValue and validVAValue] [property invalidEvdValue]

Codice	Combinazione	Esito
TC_EliminaVinoDalCarrelloOK	EUD1.LV1.TA1.VA1.EVD1	Vino eliminato dal carrello correttamente
TC_EliminaVinoDalCarrelloErrore1	EUD1.LV1.TA1.VA1.EVD2	Il vino non esiste
TC_EliminaVinoDalCarrelloErrore2	EUD1.LV1.TA1.VA2	Annata non valida
TC_EliminaVinoDalCarrelloErrore3	EUD1.LV1.TA2	Annata non valida
TC_EliminaVinoDalCarrelloErrore4	EUD1.LV2	Nome vino non valido
TC_EliminaVinoDalCarrelloErrore5	EUD2	Utente non valido

TC_SvuotaCarrello

Parametro: ID Utente Carrello (Svuota)

Categorie	Scelte
Esistenza Utente in Database - EUD	<ol style="list-style-type: none"> 1. Esiste un utente nel database con l'id specificato [property validEUDValue] 2. Non esiste un utente con l'id specificato [property invalidEUDValue]

Codice	Combinazione	Esito
TC_EliminaVinoDalCarrelloOK	EUD1	Carrello svuotato correttamente
TC_EliminaVinoDalCarrelloErrore1	EUD2	L'utente non esiste

3.5 Pagamenti

TC_ProcessaTransazione

Parametro: Carrello

Categorie	Scelte
Stato Carrello - SC	<ol style="list-style-type: none"> 1. Il carrello contiene dei vini [property validSCValue] 2. Il carrello non contiene vini [property invalidSCValue]

Parametro: Carta di Credito

Categorie	Scelte
Validità Carta di Credito - VCC	<ol style="list-style-type: none"> 1. La stringa inserita è correttamente validata secondo l'algoritmo di Luhn [if validDCValue] [property valid-VCCValue] 2. La stringa non supera la validazione tramite algoritmo di Luhn [if validDCValue] [property invalidVCCValue]
Saldo Carta di Credito - SCC	<ol style="list-style-type: none"> 1. La carta di credito consente di effettuare una transazione per il prezzo totale degli elementi nel carrello [if validVCCValue] 2. L'ente emittente della carta di credito rifiuta la transazione [if validVCCValue]

Parametro: CVC

Categorie	Scelte
Validità CVC - VCVC	<ol style="list-style-type: none"> 1. La stringa inserita ha lunghezza 3 [if validSCCValue] [property validVCVCValue] 2. La stringa non supera la validazione tramite algoritmo di Luhn [if validSCCValue] [property invalidVCVCValue]

Codice	Combinazione	Esito
TC_ProcessaTransazioneOK	SC1.VCC1.SCC1.VCVC1	Transazione effettuata correttamente
TC_ProcessaTransazioneErrore1	SC1.VCC1.SCC1.VCVC2	CVC non valido
TC_ProcessaTransazioneErrore2	SC1.VCC1.SCC2	Saldo non sufficiente
TC_ProcessaTransazioneErrore3	SC1.VCC2	Carta di Credito non valida
TC_ProcessaTransazioneErrore4	SC2	Carrello vuoto

3.6 Note aggiuntive

Per le operazioni che prevedono ID Utente in input, la validità dell'utente in questione viene assicurata dalla verifica del JWT, all'interno del suo payload è presente un claim che rappresenta proprio l'id dell'utente che sta utilizzando quel servizio.

Ogni qualvolta la verifica non va a buon fine, non si continua nella catena di handler della richiesta, bensì viene restituito un errore all'utente invitandolo ad effettuare nuovamente il login per ottenere un nuovo JWT.