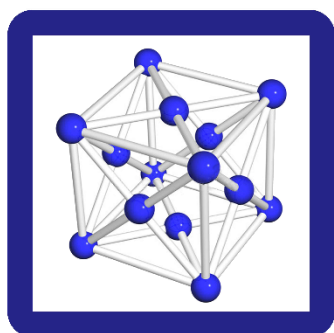




UNIVERSITÀ DEGLI STUDI DI SALERNO



UNIVERSITÀ DEGLI STUDI DI SALERNO  
**DIPARTIMENTO DI INFORMATICA**



earth minerals  
**EMC**  
EUROPEAN METALS CORPORATION

**CORSO DI INGEGNERIA DEL SOFTWARE PROF. A. DE LUCIA**  
**PROGETTO EMC**  
**TEST PLAN**  
**2020/2021**

PARTECIPANTI	MATRICOLA
ALESSANDRA POTESA'	<b>06188</b>
ROSARIO ANNUNZIATA	<b>05810</b>
GIOVANNI TAVOLO	<b>05912</b>

## Sommario

Relazioni Con Altri Documenti .....	4
Relazioni Con Il Documento Di Analisi DeiRequisiti (Rad).....	4
Relazioni Con Il System Design Document (Sdd).....	4
Relazioni Con L'object Design Document (Odd).....	4
Funzionalita' Da Testare E Da Non Testare .....	5
Gestione Utente .....	5
Gestione Categoria.....	5
Gestione Prodotto.....	5
Criteri Pass/Fail Testing.....	5
Approccio .....	5
Testing Di Unità.....	6
Testing Di Sistema .....	6
Sospensione E Ripresa.....	6
Criteri di sospensione.....	6
Criteri di ripresa.....	6
Materiale Per Il Testing .....	6
Test Case Registrazione Utente.....	6
Test Case Login Utente.....	10
Test Case Modifica Dati Utenti.....	11
Test Case Aggiunta Categoria.....	13
Test Case Modifica Categoria.....	14
Test Case Aggiunta Prodotto.....	15
Test Case Modifica Prodotto.....	17
Test Case Procedi Acquisto .....	19

## INTRODUZIONE

Lo scopo di questo documento è quello di gestire lo sviluppo e le attività di test riguardanti il sito web “European Metals Corporation”. Saranno identificati: gli elementi e le funzionalità da testare, le strategie di testing e gli strumenti utilizzati, il personale responsabile dei test, le risorse e le attività richieste per completare i test e i rischi associati al piano. Lo scopo del testing è quello di rilevare errori in maniera pianificata all'interno del codice realizzato. L'obiettivo del testing consiste nell'evitare che gli errori si presentino durante l'utilizzo del sistema dell'utente finale. I risultati prodotti dai test saranno utilizzati per comprendere dove intervenire per correggere gli errori o apportare modifiche per il migliorare il sistema. In questo documento verranno analizzate, in particolar modo, le seguenti attività:

- Gestione Utente
- Gestione Categoria
- Gestione Prodotto

### **Relazioni Con Altri Documenti**

Il Test Plan ha una stretta relazione con i documenti prodotti finora, dato che il sistema è stato pianificato nelle precedenti documentazioni. Per verificare il corretto funzionamento del sito web “European Metals Corporation” saranno usati i test cases individuati e documentati precedentemente nel processo di sviluppo del sistema. I test cases sono basati sulle funzionalità individuate nel documento di raccolta ed analisi dei requisiti.

### **Relazioni Con Il Documento Di Analisi Dei Requisiti (Rad)**

La relazione tra test plan e RAD (Requirement Analysis Document) riguarda in particolare i requisiti funzionali e non funzionali del sistema visto che i test saranno eseguiti su quelle funzionalità tenendo conto delle specifiche espresse nel documento precedente. In particolare, il RAD contiene lo scopo del sistema, l'ambito del sistema e gli obiettivi, evidenziando una panoramica di requisiti funzionali, requisiti non funzionali, scenari, casi d'uso, diagrammi e mock-up del sistema.

### **Relazioni Con Il System Design Document (Sdd)**

Tramite il System Design Document sarà possibile definire i sottosistemi e i servizi da porre a Testing.

## Relazioni Con L'object Design Document (Odd)

Tramite l'Object Design Document sarà possibile definire le classi e le componenti del sistema da porre a Testing.

## Funzionalità Da Testare E Da Non Testare

Di seguito saranno elencate le funzionalità introdotte nel sistema che saranno sottoposte a test, suddivise per ogni gestione del sistema.

### Feature da testare:

#### Gestione Utente

- Registrazione utente
- Login utente
- Modifica dati utenti

#### Gestione Categoria

- Aggiungi categoria
- Modifica categoria

#### Gestione Prodotto

- Aggiungi prodotto
- Modifica prodotto

### Feature da non testare:

Le funzionalità escluse dal testing riguardano i requisiti funzionali di bassa e media priorità o i requisiti per cui non è necessario creare dei casi di test perché non accettano input utente particolari.

#### Gestione Utente

- Visualizzazione pagine informazioni (chi Siamo, About, sedi)
- Visualizzazione ordine

#### Gestione Categoria

- Visualizzazione categoria

#### Gestione Prodotto

- Visualizzazione prodotto

## **Criteri Pass/Fail Testing**

Lo scopo del testing è quello di trovare delle failure durante l'esecuzione del sistema. Il testing ha successo se l'output osservato (finale) è diverso dall'output atteso: ciò significa che la fase di testing avrà successo se individua una failure. Nel caso verrà riscontrata una failure, bisognerà verificare a che tipo di fault è legata, se di tipo meccanico o algoritmico. Al termine dell'individuazione del fault si procederà alla sua correzione. Sarà infine iterata la fase di testing per verificare che la modifica non abbia impattato su altri componenti del sistema. La failure quindi è uno stato di condizione nel quale non si trova l'output desiderato.

## **Approccio**

L'approccio alla fase di testing si compone di tre fasi. Si inizia con il testing di unità che ha lo scopo di testare le componenti del sistema singolarmente, poi si passa al testing di integrazione in cui le componenti del sistema verranno combinate e testate come un unico gruppo, ed infine si effettua il testing di sistema che mira a verificare il funzionamento dell'intero sistema.

## **Testing Di Unità**

Per il testing di unità si utilizzerà la tecnica BLACK-BOX. Con il BLACK-BOX testing ci focalizzeremo sul comportamento dell'input/output delle singole componenti senza tener conto della loro struttura interna. A causa della mancanza di fattibilità di effettuare un test esaustivo per l'ingente quantitativo di dati di input, verrà utilizzata la strategia del Category Partition, che consente di decomporre lo spazio di input in categorie per poi partizionare le "categorie" in classi di equivalenza chiamate "scelte". Al termine saranno specificate le "combinazioni" delle scelte da testare creando delle istanze di casi di test specificando i valori dei dati effettivi per ciascuna scelta e determinare i risultati corrispondenti. Mediante il Category Partition otterremo, quindi, un test efficiente e privo di ridondanze.

## **Testing Di Sistema**

Il testing di sistema è stato realizzato tramite il tool Selenium.

## **Sospensione E Ripresa**

### **Criteri di sospensione**

La fase di testing sarà sospesa nel caso verrà rilevato un difetto che può limitare il processo di test. La sospensione del processo dovrà incidere il meno possibile sulle risorse disponibili. La

fase di testing può essere sospesa qualora si raggiungeranno gli obiettivi dichiarati, rispettando i tempi fissati.

### **Criteri di ripresa**

La fase di testing riprenderà quando il difetto verrà risolto con successo. I test verranno ripetuti per controllare se le modifiche non hanno generato nuovi errori.

### **Materiale Per Il Testing**

Gli strumenti necessari per svolgere le attività di testing sono:

- WebServer Apache Tomcat 9 in locale che gira sul sistema.
- Un DBMS MySQL che gestisce l'utilizzo del database.
- Selenium IDE per il test di integrazione.
- JUnit per il test di unità.

## Test Case Registrazione Utente

### TC\_RegistrazioneUtente

Parametro: e-mail Formato: ^\\w+([\\.-]?\\w+)*@\\w+([\\.-]?\\w+)*\\.\\w+\\.+\$	
CATEGORIE	SCELTE
Lunghezza e-mail - LE	<ol style="list-style-type: none"> <li>1. Lunghezza = 0 - campo vuoto [lunghezzaLEinvalid]</li> <li>2. Lunghezza &gt;= 1 - [property lunghezzaLEvalid]</li> </ol>
Formato e-mail - FE	<ol style="list-style-type: none"> <li>1. Non rispetta il formato [if lunghezzaLEvalid] [formatoFEinvalid]</li> <li>2. Rispetta il formato [if lunghezzaLEvalid] [property formatoFEvalid]</li> </ol>
Esiste email - EE	<ol style="list-style-type: none"> <li>1. Esiste nel DB [if lunghezzaLEvalid AND formatoFEvalid] [esisteEEinvalid]</li> <li>2. Non esiste nel DB [if lunghezzaLEvalid AND formatoFEvalid] [property EsisteEEvalid]</li> </ol>

Parametro: nome Formato: ^[a-zA-Z\u00C0-\u00ff]+\$	
CATEGORIE	SCELTE
Lunghezza nome - LN	<ol style="list-style-type: none"> <li>1. Lunghezza = 0 - campo vuoto [lunghezzaLNinvalid]</li> <li>2. Lunghezza &gt;=1 - [property lunghezzaLNvalid]</li> </ol>
Formato nome - FN	<ol style="list-style-type: none"> <li>1. Non rispetta il formato [if lunghezzaLNvalid] [formatoFNinvalid]</li> <li>2. Rispetta il formato [if lunghezzaLNvalid] [property formatoFNvalid]</li> </ol>



Parametro: username Formato: ^[0-9a-zA-Z]+\$	
CATEGORIE	SCELTE
Lunghezza username - LU	<ol style="list-style-type: none"> <li>1. Lunghezza = 0 - campo vuoto [lunghezzaLUinvalid]</li> <li>2. Lunghezza &gt;=6 - [property lunghezzaLCvalid]</li> </ol>
Formato username - FU	<ol style="list-style-type: none"> <li>1. Non rispetta il formato [if <b>lunghezzaLUvalid</b> <b>formatoFUinvalid</b>]</li> <li>2. Rispetta il formato [if <b>lunghezzaLUvalid</b>] [property <b>formatoFUvalid</b>]</li> </ol>
Esiste username - EU	<ol style="list-style-type: none"> <li>3. Esiste nel DB [if <b>lunghezzaLUvalid</b> AND <b>formatoFUvalid</b>] [<b>esisteEUinvalid</b>]</li> <li>4. Non esiste nel DB [if <b>lunghezzaEUvalid</b> AND <b>formatoFUvalid</b>] [<b>EsisteEUvalid</b>]</li> </ol>

Parametro: password Formato: .*[0-9].*	
CATEGORIE	SCELTE
Lunghezza password - LP	<ol style="list-style-type: none"> <li>1. Lunghezza = 0 - campo vuoto [lunghezzaLPinvalid]</li> <li>2. Lunghezza &gt;= 8 - [lunghezzaLPinvalid]</li> </ol>
Formato password - FP	<ol style="list-style-type: none"> <li>1. Non rispetta il formato [if <b>lunghezzaLPvalid</b>] [property <b>formatoFPinvalid</b>]</li> <li>2. Rispetta il formato [if <b>lunghezzaLPvalid</b>] [property <b>formatoFPvalid</b>]</li> </ol>

Corrispondenza password - CP	<ol style="list-style-type: none"> <li>1. Lunghezza ConfermaPassword = 0 - [<b>lunghezzaLCPinvalid</b>]</li> <li>2. ConfermaPassword non corrisponde a password [if <b>lunghezzaFPvalid</b> AND <b>formatoFPvalid</b>] [property <b>corrispondenzaCPinvalid</b>]</li> <li>3. ConfermaPassword corrisponde a password [if <b>lunghezzaFPvalid</b> AND <b>formatoFPvalid</b>] [property <b>corrispondenzaCPvalid</b>]</li> </ol>
------------------------------	--

CODICE	COMBINAZIONE	ESITO
TC_RegistrazioneUtente_1	LN1	ERROR
TC_RegistrazioneUtente_2	LN2.FN1	ERROR
TC_RegistrazioneUtente_3	LN2.FN2.LC1	ERROR
TC_RegistrazioneUtente_4	LN2.FN2.LC2.FC1	ERROR
TC_RegistrazioneUtente_5	LN2.FN2.LC2.FC2.LE1	ERROR
TC_RegistrazioneUtente_6	LN2.FN2.LC2.FC2.LE2.FE1	ERROR
TC_RegistrazioneUtente_7	LN2.FN2.LC2.FC2.LE2.FE2.EE1	ERROR
TC_RegistrazioneUtente_8	LN2.FN2.LC2.FC2.LE2.FE2.EE2.LU1	ERROR
TC_RegistrazioneUtente_9	LN2.FN2.LC2.FC2.LE2.FE2.EE2.LU2.FU1	ERROR
TC_RegistrazioneUtente_10	LN2.FN2.LC2.FC2.LE2.FE2.EE2.LU2.FU2.EU1	ERROR
TC_RegistrazioneUtente_11	LN2.FN2.LC2.FC2.LE2.FE2.EE2.LU2.FU2.EU2.LP1	ERROR
TC_RegistrazioneUtente_12	LN2.FN2.LC2.FC2.LE2.FE2.EE2.LU2.FU2.EU2.LP2.FP1	ERROR
TC_RegistrazioneUtente_13	LN2.FN2.LC2.FC2.LE2.FE2.EE2.LU2.FU2.EU2.LP2.FP2.CP1	ERROR
TC_RegistrazioneUtente_14	LN2.FN2.LC2.FC2.LE2.FE2.EE2.LU2.FU2.EU2.LP2.FP2.CP2	ERROR
TC_RegistrazioneUtente_15	LN2.FN2.LC2.FC2.LE2.FE2.EE2.LU2.FU2.EU2.LP2.FP2.CP3	CORRETTO

## Test Case Login Utente

### TC\_LoginUtente

Parametro: username Formato: ^[0-9a-zA-Z]+\$	
CATEGORIE	SCELTE
Lunghezza username - LU	3. Lunghezza = 0 - campo vuoto [lunghezzaLUinvalid] 4. Lunghezza >=6 - [property lunghezzaLCvalid]
Esiste username - EU	5. Esiste nel DB [if <b>lunghezzaLUvalid</b> AND <b>formatoFUvalid</b> [ <b>esisteEUinvalid</b> ] 6. Non esiste nel DB [if <b>lunghezzaEUvalid</b> AND <b>formatoFUvalid</b> ] [ <b>EsisteEUvalid</b> ]

Parametro: password Formato: .*[0-9].*	
CATEGORIE	SCELTE
Lunghezza password - LP	1. Lunghezza = 0 - campo vuoto [lunghezzaLPinvalid] 2. Lunghezza >= 8 - [lunghezzaLPinvalid]
Corrispondenza password e-mail - CPE	1. Non c'è corrispondenza tra password e e-mail nel database [if <b>lunghezzaLPvalid</b> ] [property <b>corrispondenzaCPEinvalid</b> ] 2. C'è corrispondenza tra password e e-mail nel database [if <b>lunghezzaLPvalid</b> ] [property <b>corrispondenzaCPEvalid</b> ]

CODICE	COMBINAZIONE	ESITO
TC_LoginUtente_1	LE1	ERROR
TC_LoginUtente_2	LE2.EE1	ERROR
TC_LoginUtente_3	LE2.EE2.LP1	ERROR
TC_LoginUtente_4	LE2.EE2.LP2.CPE1	ERROR
TC_LoginUtente_5	LE.EE2.LP2.CPE2	CORRETTO

## Test Case Modifica Dati Utenti

TC\_ModificaDatiUtenti

Parametro: username Formato: ^[0-9a-zA-Z]+\$	
CATEGORIE	SCELTE
Lunghezza username - LU	5. Lunghezza = 0 - campo vuoto [lunghezzaLUinvalid] 6. Lunghezza >=6 - [property lunghezzaLCvalid]
Formato username - FU	1. Non rispetta il formato [if <b>lunghezzaLUinvalid</b> ] [formatoFUinvalid] 2. Rispetta il formato [if <b>lunghezzaLUinvalid</b> ] [property <b>formatoFUinvalid</b> ]
Esiste username - EU	1. Esiste nel DB [if <b>lunghezzaLUinvalid</b> AND <b>formatoFUinvalid</b> ] [esisteEUinvalid] 2. Non esiste nel DB [if <b>lunghezzaEUinvalid</b> AND <b>formatoFUinvalid</b> ] [EsisteEUinvalid]

Parametro: nome Formato: ^[ a-zA-Z\u00C0-\u00ff]+\$	
CATEGORIE	SCELTE
Lunghezza nome - LN	<ol style="list-style-type: none"> <li>1. Lunghezza = 0 - campo vuoto [lunghezzaLNinvalid]</li> <li>2. Lunghezza &gt;=1 - [property lunghezzaLNvalid]</li> </ol>
Formato nome - FN	<ol style="list-style-type: none"> <li>1. Non rispetta il formato [if <b>lunghezzaLNvalid</b> <b>formatoFNinvalid</b>]</li> <li>2. Rispetta il formato [if <b>lunghezzaLNvalid</b>] [property <b>formatoFNvalid</b>]</li> </ol>

Parametro: e-mail Formato: ^\\w+([\\.-]?\\w+)*@\\w+([\\.-]?\\w+)*\\.\\w+\$	
CATEGORIE	SCELTE
Lunghezza e-mail - LE	<ol style="list-style-type: none"> <li>3. Lunghezza = 0 - campo vuoto [lunghezzaLEinvalid]</li> <li>4. Lunghezza &gt;= 1 - [property lunghezzaLEvalid]</li> </ol>
Formato e-mail - FE	<ol style="list-style-type: none"> <li>3. Non rispetta il formato [if <b>lunghezzaLEvalid</b> <b>formatoFEinvalid</b>]</li> <li>4. Rispetta il formato [if <b>lunghezzaLEvalid</b>] [property <b>formatoFEvalid</b>]</li> </ol>
Esiste email - EE	<ol style="list-style-type: none"> <li>3. Esiste nel DB [if <b>lunghezzaLEvalid</b> AND <b>formatoFEvalid</b>] [esisteEEinvalid]</li> <li>4. Non esiste nel DB [if <b>lunghezzaLEvalid</b> AND <b>formatoFEvalid</b>] [property <b>EsisteEEvalid</b>]</li> </ol>

CODICE	COMBINAZIONE	ESITO
TC_ModificaDatiPersonali_1	LU1	ERROR
TC_ModificaDatiPersonali_2	LU2.FU1	ERROR
TC_ModificaDatiPersonali_3	LU2.FU2.EU1	ERROR
TC_ModificaDatiPersonali_4	LU2.FU2.EU2.LN1	ERROR
TC_ModificaDatiPersonali_5	LU2.FU2.EU2.LN2.FN1	ERROR
TC_ModificaDatiPersonali_6	LU2.FU2.EU2.LN2.FN2.LE1	ERROR
TC_ModificaDatiPersonali_7	LU2.FU2.EU2.LN2.FN2.LE2.FE1	ERROR
TC_ModificaDatiPersonali_8	LU2.FU2.EU2.LN2.FN2.LE2.FE2.EE1	ERROR
TC_ModificaDatiPersonali_5	LU2.FU2.EU2.LN2.FN2.LE2.FE2.EE2	CORRETTO

## Test Case Aggiunta Categoria

TC\_AggiuntaCategoria

Parametro: nome	
CATEGORIE	SCELTE
Lunghezza nome - LN	<ol style="list-style-type: none"> <li>Lunghezza = 0 - campo vuoto [lunghezzaLNinvalid]</li> <li>Lunghezza &gt;=1 - [property lunghezzaLNvalid]</li> </ol>
Esiste nome - EN	<ol style="list-style-type: none"> <li>Esiste nel DB [if lunghezzaLNvalid] [esisteENinvalid]</li> <li>Non esiste nel DB [if lunghezzaLNvalid] [EsisteENvalid]</li> </ol>

Parametro: descrizione	
CATEGORIE	SCELTE
Lunghezza descrizione - LD	1. Lunghezza = 0 - campo vuoto [ <b>lunghezzaLDinvalid</b> ] 2. Lunghezza >=1 - [property <b>lunghezzaLDvalid</b> ]

CODICE	COMBINAZIONE	ESITO
TC_AggiuntaCategoria_1	LN1	ERROR
TC_AggiuntaCategoria_2	LN2.EN1	ERROR
TC_AggiuntaCategoria_3	LN2.EN2.LD1	ERROR
TC_AggiuntaCategoria_4	LN2.EN2.LD2	CORRETTO

## Test Case Modifica Categoria

TC\_ModificaCategoria

Parametro: nome	
CATEGORIE	SCELTE
Lunghezza nome - LN	1. Lunghezza = 0 - campo vuoto [ <b>lunghezzaLNinvalid</b> ] 2. Lunghezza >=1 - [property <b>lunghezzaLNvalid</b> ]

Parametro: nome e descrizione	
CATEGORIE	SCELTE
Esiste categoria - EC	1. Esiste nel DB [ <b>esisteECinvalid</b> ] 2. Non esiste nel DB [ <b>EsisteECvalid</b> ]

Parametro: descrizione	
CATEGORIE	SCELTE
Lunghezza descrizione - LD	3. Lunghezza = 0 - campo vuoto [ <b>lunghezzaLDinvalid</b> ] 4. Lunghezza >=1 - [ <b>property lunghezzaLDvalid</b> ]

CODICE	COMBINAZIONE	ESITO
TC_ModificaCategoria_1	LN1	ERROR
TC_ModificaCategoria_2	LN2.EC1	ERROR
TC_ModificaCategoria_3	LN2.EC2.LD1	ERROR
TC_ModificaCategoria_4	LN2.EC2.LD2	CORRETTO



## Test Case Aggiunta Prodotto

### TC\_AggiuntaProdotto

Parametro: categoria	
CATEGORIE	SCELTE
Selezione categoria - SC	<ol style="list-style-type: none"><li>1. Selezione = false [selezioneSCinvalid]</li><li>2. Selezione = true - [property <b>selezioneSCvalid</b>]</li></ol>

Parametro: nome	
CATEGORIE	SCELTE
Lunghezza nome - LN	<ol style="list-style-type: none"><li>1. Lunghezza = 0 - campo vuoto [lunghezzaLNinvalid]</li><li>2. Lunghezza &gt;=1 - [property <b>lunghezzaLNvalid</b>]</li></ol>
Esiste nome - EN	<ol style="list-style-type: none"><li>1. Esiste nel DB [if <b>lunghezzaLNvalid</b>] [esisteENinvalid]</li><li>2. Non esiste nel DB [if <b>lunghezzaLNvalid</b>] [EsisteENvalid]</li></ol>

Parametro: descrizione	
CATEGORIE	SCELTE
Lunghezza descrizione - LD	<ol style="list-style-type: none"><li>1. Lunghezza = 0 - campo vuoto [lunghezzaLDinvalid]</li><li>2. Lunghezza &gt;=1 - [property <b>lunghezzaLDvalid</b>]</li></ol>

Parametro: prezzo	
CATEGORIE	SCELTE
Inserimento prezzo - IP	1. Inserimento = false [ <b>inserimentoIPinvalid</b> ] 2. Inserimento = true - [property <b>inserimentoIPvalid</b> ]

Parametro: iva	
CATEGORIE	SCELTE
Lunghezza iva - LI	3. Lunghezza = 0 - campo vuoto [ <b>lunghezzaLIinvalid</b> ] 4. Lunghezza >=1 - [property <b>lunghezzaLIvalid</b> ]

CODICE	COMBINAZIONE	ESITO
TC_AggiuntaProdotto_1	SC1	ERROR
TC_AggiuntaProdotto_2	SC2.LN1	ERROR
TC_AggiuntaProdotto_3	SC2.LN2.EN1	ERROR
TC_AggiuntaProdotto_4	SC2.LN2.EN2.LD1	ERROR
TC_AggiuntaProdotto_5	SC2.LN2.EN2.LD2.IP1	ERROR
TC_AggiuntaProdotto_6	SC2.LN2.EN2.LD2.IP2.LI1	ERROR
TC_AggiuntaProdotto_7	SC2.LN2.EN2.LD2.IP2.LI2	CORRETTO

## Test Case Modifica Prodotto

### TC\_ModificaProdotto

Parametro: categoria	
CATEGORIE	SCELTE
Selezione categoria - SC	3. Selezione = false [selezioneSCinvalid] 4. Selezione = true - [property <b>selezioneSCvalid</b> ]

Parametro: nome	
CATEGORIE	SCELTE
Lunghezza nome - LN	3. Lunghezza = 0 - campo vuoto [lunghezzaLNinvalid] 4. Lunghezza >=1 - [property <b>lunghezzaLNvalid</b> ]

Parametro: categoria, nome, descrizione, prezzo, iva	
CATEGORIE	SCELTE
Esiste prodotto- EP	5. Esiste nel DB [ <b>esisteEPinvalid</b> ] 6. Non esiste nel DB [ <b>EsisteEPvalid</b> ]

Parametro: descrizione	
CATEGORIE	SCELTE
Lunghezza descrizione - LD	5. Lunghezza = 0 - campo vuoto [ <b>lunghezzaLDinvalid</b> ] 6. Lunghezza >=1 - [property <b>lunghezzaLDvalid</b> ]

Parametro: prezzo	
CATEGORIE	SCELTE
Inserimento prezzo - IP	3. Inserimento = false [ <b>inserimentoIPinvalid</b> ] 4. Inserimento = true - [property <b>inserimentoIPvalid</b> ]

Parametro: iva	
CATEGORIE	SCELTE
Lunghezza iva - LI	7. Lunghezza = 0 - campo vuoto [ <b>lunghezzaLlinvalid</b> ] 8. Lunghezza >=1 - [property <b>lunghezzaLlvalid</b> ]

CODICE	COMBINAZIONE	ESITO
TC_ModificaProdotto_1	SC1	ERROR
TC_ModificaProdotto_2	SC2.LN1	ERROR
TC_ModificaProdotto_3	SC2.LN2.EP1	ERROR
TC_ModificaProdotto_4	SC2.LN2.EP2.LD1	ERROR
TC_ModificaProdotto_5	SC2.LN2.EP2.LD2.IP1	ERROR

TC_ModificaProdotto_6	SC2.LN2.EP2.LD2.IP2.LI1	ERROR
TC_ModificaProdotto_7	SC2.LN2.EP2.LD2.IP2.LI2	CORRETTO

## Test Case Procedi Acquisto

### TC\_ProcediAcquisto

Parametro: indirizzo	
CATEGORIE	SCELTE
Lunghezza indirizzo - LI	1. Lunghezza = 0 - campo vuoto [ <b>lunghezzaLIinvalid</b> ] 2. Lunghezza >=1 - [property <b>lunghezzaLIvalid</b> ]

Parametro: utente	
CATEGORIE	SCELTE
Log utente - LU	1. Login = false [ <b>inserimentoLUinvalid</b> ] 2. Login = true - [property <b>inserimentoLUvalid</b> ]

CODICE	COMBINAZIONE	ESITO
TC_ProcediAcquisto_1	LI1	ERROR
TC_ProcediAcquisto_2	LI2.LU1	ERROR
TC_ProcediAcquisto_3	LI2.LU2	CORRETTO