



TPD

TEST PLAN DOCUMENT

ModuLink

Riferimento	NC08_TPD_ver.1.0
Versione	1.0
Data	12/12/2025
Destinatario	Studenti di Ingegneria del Software 2025/26
Presentato da	Buzi Arjel, Carpentieri Daniele, Chikviladze Aleksandre, Cito Roberto.
Codice Gruppo	NC08
Approvato da	



Revision History

Data	Versione	Descrizione	Autori
12/12/2025	0.1	Prima stesura	Roberto, Arjel
15/12/2025	0.2	Inizio scrittura Test Cases	Roberto
16/12/2025	0.3	Test delle Funzionalità	Daniele
17/12/2025	0.4	Completamento del documento con ultima tabella	Daniele
20/12/2025	1.0	Revisione completa	Tutti



Laurea Triennale in informatica - Università di Salerno
Corso di *Ingegneria del Software* - Prof. C. Gravino

Team members

Nome	Ruolo nel progetto	Acronimo	Informazioni di contatto
Roberto Cito	Team Member	RC	r.cito@studenti.unisa.it
Daniele Carpentieri	Team Member	DC	d.carpentieri8@studenti.unisa.it
Aleksandre Chikviladze	Team Member	AC	a.chikviladze@studenti.unisa.it
Arjel Buzi	Team Member	AB	a.buzi@studenti.unisa.it



SOMMARIO

Revision History	2
Team members	3
Revision History	
1. Introduzione	5
1.1 Obiettivo del Sistema	5
1.2 Riferimenti	5
2. Funzionalità da Testare e Non.....	6
3. Criteri di Accettazione e Fallimento	7
3.1 Criteri di Accettazione.....	7
3.2 Criteri di Fallimento.....	8
4. Strumenti Hardware e Software Per il Testing	9
4.1 Strumenti Hardware Per il Testing	9
4.2 Strumenti Software Per il Testing	9
5. Test Cases.....	11
5.1 Specifica dei Test Cases	11



1. Introduzione

1.1 Obiettivo del Sistema

L'obiettivo è fornire alle aziende uno strumento centralizzato e modulare capace di semplificare la gestione delle attività interne e delle risorse digitali. Il sistema è progettato per supportare:

- la gestione dei moduli aziendali
- l'organizzazione strutturata degli utenti e dei ruoli
- la gestione operativa di task, calendari ed eventi
- l'amministrazione di risorse e prodotti tramite il modulo magazzino
- l'integrazione di moduli personalizzati per esigenze specifiche dell'azienda

1.2 Riferimenti

Di seguito sono riportati i riferimenti a risorse che possono agevolare la lettura del documento, o fornire maggiori informazioni.

Nome Riferimento	Riferimento
Libro	“Object-Oriented Software Engineering: Conquering Complex and Changing Systems”, terza edizione, Bernd Bruegge & Allen Dutoit, 2014.
SOW	Statement Of Work (SOW) Chiuso
RAD	Requirement Analysis Document Chiuso
SDD	System Desing Document Chiuso



ODD

[Object Design Document](#) Chiuso

2. Funzionalità da Testare e Non

Poiché il budget destinato alla fase di testing del sistema non consente di validare l'intera piattaforma in tutte le sue funzionalità, è stato necessario selezionare solo un sottoinsieme mirato di componenti

In particolare, sono state individuate quattro funzionalità chiave, in linea con quanto stabilito nel documento SOW: ogni membro del gruppo deve occuparsi del testing di una funzionalità specifica, garantendo così una copertura di test significativa ma sostenibile.

Per ciascuna funzionalità vengono riportati:

- il sottosistema da cui è estratta
- una descrizione sintetica del sottosistema
- un nome rappresentativo della funzionalità
- una descrizione chiara della specifica da verificare

GTM

Il Gestore del Task Manager fornisce i servizi necessari per creare, modificare, assegnare e visualizzare task all'interno dell'istanza aziendale. È uno dei moduli più utilizzati dagli utenti operativi.

Nome	Descrizione
Creazione task	L'utente deve poter creare un task fornendo titolo, descrizione, priorità, eventuali assegnatari e scadenza.

GDM

Il Gestore del Magazzino permette di creare, aggiornare e mantenere l'inventario aziendale. Gestisce prodotti, quantità, operazioni di acquisto/vendita e la loro tracciabilità.

Nome	Descrizione
Aggiunta prodotto	L'utente deve poter creare un nuovo prodotto inserendo nome, descrizione, quantità, prezzo e categoria.



GDU	
Il Gestore degli Utenti si occupa della gestione completa degli account aziendali. Gestisce registrazione, login, cambio password e tutte le operazioni legate ai profili utente.	
Nome	Descrizione
Registrazione utente	Il responsabile deve poter registrare un nuovo utente e assegnargli uno o più ruoli all'interno dell'istanza aziendale.

GCA	
Il Gestore del Calendario consente di creare, gestire e visualizzare eventi e appuntamenti aziendali. È integrato con il Task Manager e con il sistema di notifiche interne.	
Nome	Descrizione
Creazione evento	L'utente deve poter creare un nuovo evento specificando titolo, descrizione, data, ora e partecipanti.

3. Criteri di Accettazione e Fallimento

3.1 Criteri di Accettazione

La fase di testing sarà considerata conclusa e accettata (validata) esclusivamente al raggiungimento della **copertura totale (100%)** dei Test Cases definiti nella sezione "Funzionalità da Testare". Ogni test deve essere eseguito e il suo esito debitamente documentato.

Nota sull'efficacia del testing (Fault Detection): In linea con i principi di *Fault Detection*, si adotta la seguente distinzione semantica per valutare il valore informativo dei test eseguiti:

- **Test Efficace (Successo):** Il test case intercetta una discrepanza tra il risultato atteso e quello attuale, portando alla luce un difetto (*failure*). Questo esito è considerato il vero "successo" del testing, in quanto rivela un bug latente e permette di migliorare la robustezza del sistema.
- **Test Inefficace (Conforme):** Il risultato attuale coincide con quello atteso. Sebbene confermi la correttezza della funzionalità nello scenario specifico, questo esito è considerato a basso valore aggiunto in ottica di *detection*, poiché non ha evidenziato nuove criticità.



3.2 Criteri di Fallimento

La fase di testing sarà considerata non valida, e quindi fallimentare ai fini del rilascio, qualora si verifichi una delle seguenti condizioni:

- **Copertura incompleta:** Non è stato possibile eseguire la totalità dei Test Cases pianificati (es. per blocchi tecnici o mancanza di tempo).
- **Mancata documentazione:** I risultati dei test non sono stati tracciati o documentati secondo gli standard definiti, impedendo la verificabilità delle operazioni svolte.



4. Strumenti Hardware e Software Per il Testing

4.1 Strumenti Hardware Per il Testing

Per garantire che la piattaforma web funzioni correttamente su diversi dispositivi e che l'interfaccia rispetti i requisiti di usabilità, l'ambiente di test comprende le seguenti categorie di hardware:

1. **Postazioni Desktop e Laptop:** necessarie per verificare il layout grafico su schermi di dimensioni standard e la compatibilità con i principali sistemi operativi.
 - **Ambiente Windows:** PC dotati delle versioni più recenti di Windows per simulare l'esperienza dell'utente medio.
 - **Ambiente macOS:** dispositivi Apple utilizzati per verificare la resa su schermi ad alta risoluzione (Retina) e il comportamento delle periferiche specifiche (trackpad).
2. **Dispositivi Mobile:** utilizzati per testare il Responsive Design e la corretta ricezione dei comandi touch.
 - **Dispositivi iOS:** iPhone e iPad aggiornati per coprire l'ecosistema Apple mobile.
 - **Dispositivi Android:** una selezione di smartphone con diverse risoluzioni dello schermo, fondamentale per gestire la varietà di dispositivi Android presenti sul mercato.

4.2 Strumenti Software Per il Testing

La strategia di testing per il progetto, sviluppato in Java con framework Spring, si basa sull'automazione e sull'integrazione continua (CI). Di seguito sono elencati gli strumenti software adottati.

1. **Testing Backend (Java/Spring):** per la verifica del codice sorgente e delle funzionalità lato server vengono utilizzati i seguenti strumenti standard:
 - **JUnit 5:** è il framework principale utilizzato per la scrittura dei test unitari (Unit Testing). Permette di verificare il corretto funzionamento delle singole classi e metodi tramite asserzioni e annotazioni (es. `@Test`).
 - **Spring Boot Test:** modulo utilizzato per i test di integrazione. Consente di avviare l'intero contesto dell'applicazione (o parti di esso, come il livello dati) per controllare che i vari componenti di Spring interagiscano correttamente tra loro.
 - **Mockito:** libreria di *mocking* usata insieme a JUnit. Serve a simulare il comportamento di dipendenze esterne (come il database), permettendo di testare una classe in isolamento senza che il test dipenda dal funzionamento di altri moduli.



2. **Continuous Integration (CI):** per evitare di eseguire manualmente i test ogni volta, il processo è stato automatizzato tramite pipeline.
 - **GitHub Actions:** strumento integrato in GitHub che permette di creare workflow automatici. È stato configurato un file YAML che si attiva ad ogni *push* sul repository.
 - **Automazione:** la pipeline si occupa di preparare l'ambiente Java, scaricare le librerie necessarie e lanciare automaticamente tutti i test. Se un test fallisce, il sistema impedisce di unire il codice (merge), garantendo che nel ramo principale arrivi solo codice funzionante.
3. **Frontend e Debugging:** per la verifica dell'interfaccia utente lato client.
 - **Browser DevTools:** strumenti di sviluppo integrati nei browser (Chrome, Firefox, Safari) utilizzati per ispezionare il codice HTML/CSS generato e per il debugging visivo delle pagine.



5. Test Cases

Per la definizione dei casi di test relativi alle funzionalità selezionate, è stato adottato il metodo del **Category Partition**. Questo approccio strutturale permette di analizzare le specifiche funzionali scomponendole in:

- **Parametri:** le unità di informazione che il sistema deve elaborare.
- **Categorie:** le caratteristiche o proprietà rilevanti di ciascun parametro.
- **Scelte:** i possibili valori (validi o errati) che ogni categoria può assumere.

Combinando sistematicamente queste scelte, abbiamo generato i **Test Frames**, ovvero degli scenari logici che rappresentano la base per l'esecuzione dei test

5.1 Specifica dei Test Cases

Per ciascuna funzionalità analizzata (Registrazione Utente, Creazione Task, Creazione Evento, Aggiunta Prodotto) vengono presentate due tabelle distinte:

1. **Tabella di Definizione:** in cui vengono identificati i parametri, le categorie e le singole scelte, distinguendo tra input validi e condizioni di errore.
2. **Tabella dei Test Frames:** che contiene le combinazioni delle scelte individuate, definendo per ognuna l'esito atteso (successo o fallimento).

Queste tabelle costituiscono la documentazione preliminare necessaria per la successiva formalizzazione dei test nel documento di specifica (TCSD).

ID		
VRF_GDU.1		
Nome	Non vuota [NBN]	1. NBN = false [ERR] 2. NBN = true [PROPERTY_NB_OK]
Nome	Lunghezza [LNEN]	1. LNEN < 2 [ERR] 2. LNEN > 50 [ERR] 3. LNEN >=2 && LNE<=50 [PROPERTY_LNE_OK]
Cognome	Non vuota [NBC]	1. NBC = false [ERR] 2. NBC = true [PROPERTY_NB_OK]



Cognome	Lunghezza [LNEC]	<ol style="list-style-type: none"> 1. LNEC < 2 [ERR] 2. LNEC > 50 [ERR] 3. LNEC >=2 && LNE<=50 [PROPERTY_LNE_OK]
E-Mail	Non vuota [NBE]	<ol style="list-style-type: none"> 1. NBE = false [ERR] 2. NBE = true [PROPERTY_NB_OK]
E-Mail	Lunghezza [LNEE]	<ol style="list-style-type: none"> 1. LNEE < 5 [ERR] 2. LNEE > 50 [ERR] 3. LNEE >=5 && LNE<=50 [PROPERTY_LNE_OK]
E-Mail	Esistenza [ESIE]	<ol style="list-style-type: none"> 1. ESIE = true [ERR] 2. ESIE = false [PROPERTY_ESI_OK]
E-Mail	Formato [FORE]	<ol style="list-style-type: none"> 1. FORE = true [PROPERTY_FOR_OK] 2. FORE = false [ERR]
Numero di telefono	Formato [FORT]	<ol style="list-style-type: none"> 1. FORT = true [PROPRERTY_FOR_OK] 2. FORT = false [ERR]
Password	Non vuota [NBP]	<ol style="list-style-type: none"> 1. NBP = false [ERR] 2. NBP = true [PROPERTY_NB_OK]
Password	Lunghezza [LNEP]	<ol style="list-style-type: none"> 1. LNEP < 2 [ERR] 2. LNEP > 50 [ERR] 3. LNEP >=2 && LNE<=50 [PROPERTY_LNE_OK]
Conferma Password	Non vuota [NBCP]	<ol style="list-style-type: none"> 1. NBCP = false [ERR] 2. NBCP = true [PROPERTY_NB_OK]



Conferma Password	Lunghezza [LNECP]	1. LNECP < 2 [ERR] 2. LNECP > 50 [ERR] 3. LNECP >=2 && LNE<=50 [PROPERTY_LNE_OK]
Password & Conferma Password	Uguaglianza [UGU]	1. UGU = true [PROPERTY_UGU_OK] 2. UGU = false [ERR]
Immagine	Caricata [CAR]	1. CAR = true [PROPERTY_CAR_OK] 2. CAR = false [ERR]
Immagine	Non una foto [NUF]	1. NUF = true [ERR] 2. NUF = false [PROPERTY_NUF_OK]
Immagine	Dimensione [DIM]	1. DIM >12 [ERR] 2. DIM <=12 [PROPERTY_DIM_OK]

ID		
TF_GDU.1		
TC1_GDU.1	NBN1	[ERR]: Campo nome vuoto
TC2_GDU.1	NBN2, LNEN1	[ERR]: Lunghezza del nome < 2 caratteri
TC3_GDU.1	NBN2, LNEN2	[ERR]: Lunghezza del nome > 50 caratteri
TC4_GDU.1	NBN2, LNEN3, NBC1	[ERR]: Campo cognome vuoto



Laurea Triennale in informatica - Università di Salerno
Corso di *Ingegneria del Software* - Prof. C. Gravino

TC5_GDU.1	NBN2, LNEN3, NBC2, LNEC1	[ERR]: Lunghezza del cognome < 2 caratteri
TC6_GDU.1	NBN2, LNEN3, NBC2, LNEC2	[ERR]: Lunghezza del cognome > 50 caratteri
TC7_GDU.1	NBN2, LNEN3, NBC2, LNEC3, NBE1	[ERR]: Campo E-Mail vuoto
TC8_GDU.1	NBN2, LNEN3, NBC2, LNEC3, NBE2, LNEE1	[ERR]: Lunghezza E-Mail < 2 caratteri
TC9_GDU.1	NBN2, LNEN3, NBC2, LNEC3, NBE2, LNEE2	[ERR]: Lunghezza E-Mail > 50 caratteri
TC10_GDU.1	NBN2, LNEN3, NBC2, LNEC3, NBE2, LNEE3, ESIE1	[ERR]: E-Mail già esistente nel sistema
TC11_GDU.1	NBN2, LNEN3, NBC2, LNEC3, NBE2, LNEE3, ESIE2, FORE2	[ERR]: Formato E-Mail non valido
TC12_GDU.1	NBN2, LNEN3, NBC2, LNEC3, NBE2, LNEE3, ESIE2, FORE1, FORT2	[ERR]: Formato Telefono non valido
TC13_GDU.1	NBN2, LNEN3, NBC2, LNEC3, NBE2, LNEE3, ESIE2, FORE1, FORT1, NBP1	[ERR]: Campo Password vuoto
TC14_GDU.1	NBN2, LNEN3, NBC2, LNEC3, NBE2, LNEE3, ESIE2, FORE1, FORT1, NBP2, LNEP1	[ERR]: Lunghezza Password < 2 caratteri
TC15_GDU.1	NBN2, LNEN3, NBC2, LNEC3, NBE2, LNEE3, ESIE2, FORE1, FORT1, NBP2, LNEP2	[ERR]: Lunghezza Password > 50 caratteri
TC16_GDU.1	NBN2, LNEN3, NBC2, LNEC3, NBE2, LNEE3, ESIE2, FORE1, FORT1, NBP2, LNEP3, NBCP1	[ERR]: Campo Conferma Password vuoto
TC17_GDU.1	NBN2, LNEN3, NBC2, LNEC3, NBE2, LNEE3, ESIE2, FORE1, FORT1, NBP2, LNEP3, NBCP2, LNECP1	[ERR]: Lunghezza Conf. Password < 2 caratteri
TC18_GDU.1	NBN2, LNEN3, NBC2, LNEC3, NBE2, LNEE3, ESIE2, FORE1, FORT1, NBP2, LNEP3, NBCP2, LNECP2	[ERR]: Lunghezza Conf. Password > 50 caratteri
TC19_GDU.1	NBN2, LNEN3, NBC2, LNEC3, NBE2, LNEE3, ESIE2, FORE1, FORT1, NBP2, LNEP3, NBCP2, LNECP3, UGU2	[ERR]: Password e Conferma Password non coincidono



TC20_GDU.1	NBN2, LNE3, NBC2, LNEC3, NBE2, LNEE3, ESIE2, FORE1, FORT1, NBP2, LNEP3, NBCP2, LNECP3, UGU1, CAR1, NUF1	[ERR]: Il file caricato non è un'immagine
TC21_GDU.1	NBN2, LNE3, NBC2, LNEC3, NBE2, LNEE3, ESIE2, FORE1, FORT1, NBP2, LNEP3, NBCP2, LNECP3, UGU1, CAR1, NUF2, DIM1	[ERR]: Dimensione immagine > 12 MB
TC22_GDU.1	NBN2, LNE3, NBC2, LNEC3, NBE2, LNEE3, ESIE2, FORE1, FORT1, NBP2, LNEP3, NBCP2, LNECP3, UGU1, CAR1, NUF2, DIM2	[OK]: Registrazione effettuata con successo

ID		
VRF_GTM.1		
Titolo	Non vuoto [NBT]	1. NBT = false [ERR] 2. NBT = true [PROPERTY_NB_OK]
Titolo	Lunghezza [LNT]	1. LNT < 2 [ERR] 2. LNT > 150 [ERR] 3. $2 \leq LNT \leq 150$ [PROPERTY_LN_OK]
Priorità	Valore valido [VALP]	1. VALP < 0 [ERR] 2. VALP > 5 [ERR] 3. $0 \leq VALP < 5$ [PROPERTY_VAL_OK]
Scadenza	Presente [PRS]	1. PRS = false [PROPERTY_PRS_NULL_OK] 2. PRS = true [PROPERTY_PRS_OK]
Scadenza	Data valida [DAT]	1. DAT < Data_Creazione [ERR] 3. DAT ≥ Data_Creazione [PROPERTY_DAT_OK]
Utente assegnato	Esistenza [ESUA]	1. ESUA = false [ERR] 2. ESUA = true [PROPERTY_ESI_OK]



ID		
TF_ GTM.1		
TC1_ GTM.1	NBT1	[ERR]: Titolo del task vuoto
TC2_ GTM.1	NBT2, LNT1	[ERR]: Lunghezza titolo < 2 caratteri
TC3_ GTM.1	NBT2, LNT2	[ERR]: Lunghezza titolo > 150 caratteri
TC4_ GTM.1	NBT2, LNT3, VALP1	[ERR]: Valore priorità < 0
TC5_ GTM.1	NBT2, LNT3, VALP2	[ERR]: Valore priorità > 3
TC6_ GTM.1	NBT2, LNT3, VALP3, PRS2, DAT1	[ERR]: Data scadenza antecedente alla data di creazione
TC7_ GTM.1	NBT2, LNT3, VALP3, PRS2, DAT2, ESUA1	[ERR]: Utente assegnato non esistente
TC8_ GTM.1	NBT2, LNT3, VALP3, PRS2, DAT2, ESUA2, NBC2	[OK]: Task creato con successo (<i>Con Scadenza</i>)
TC9_ GTM.1	NBT2, LNT3, VALP3, PRS1, ESUA2, NBC2	[OK]: Task creato con successo (<i>Senza Scadenza</i>)



ID		
VRF_GDE.1		
Nome	Non vuoto [NBE]	1. NBE = false [ERR] 2. NBE = true [PROPERTY_NB_OK]
Nome	Lunghezza [LNE]	1. LNE < 2 [ERR] 2. LNE > 200 [ERR] 3. $2 \leq LNE \leq 200$ [PROPERTY_LN_OK]
Luogo	Presente [PRL]	1. PRL = false [PROPERTY_NULL_OK] 2. PRL = true [PROPERTY_PR_OK]
Luogo	Lunghezza [LNEL]	1. LNEL > 200 [ERR] 2. LNEL \leq 200 [PROPERTY_LN_OK]
Data/Ora Inizio	Presente [PRI]	1. PRI = false [ERR] 2. PRI = true [PROPERTY_PR_OK]
Data/Ora Inizio	Valida [DATI]	1. DATI < Data_limite [ERR] 2. DATI \geq Data_limite [PROPERTY_DAT_OK]
Data Fine	Presente [PRF]	1. PRF = false [PROPERTY_NULL_OK] 2. PRF = true [PROPERTY_PR_OK]



Data Fine	Coerenza temporale [COF]	1. COF < Data_Ora_Inizio [ERR] 2. COF ≥ Data_Ora_Inizio [PROPERTY_COF_OK]
------------------	-----------------------------	---

ID		
TF_GDE.1		
TC1_GDE.1	NBE1	[ERR]: Nome dell'evento vuoto
TC2_GDE.1	NBE2, LNE1	[ERR]: Lunghezza nome < 2 caratteri
TC3_GDE.1	NBE2, LNE2	[ERR]: Lunghezza nome > 200 caratteri
TC4_GDE.1	NBE2, LNE3, PRL2, LNEL1	[ERR]: Lunghezza luogo > 300 caratteri
TC5_GDE.1	NBE2, LNE3, PRL2, LNEL2, PRI1	[ERR]: Data/Ora Inizio non selezionata
TC6_GDE.1	NBE2, LNE3, PRL2, LNEL2, PRI2, DATI1	[ERR]: Data Inizio antecedente alla data limite
TC7_GDE.1	NBE2, LNE3, PRL2, LNEL2, PRI2, DATI2, PRF2, COF1	[ERR]: Data Fine antecedente alla Data Inizio
TC8_GDE.1	NBE2, LNE3, PRL2, LNEL2, PRI2, DATI2, PRF2, COF2	[OK]: Successo (Evento Completo: con <i>Luogo e Data Fine</i>)
TC9_GDE.1	NBE2, LNE3, PRL1, PRI2, DATI2, PRF1	[OK]: Successo (Evento Minimo: <i>No Luogo, No Data Fine</i>)
TC10_GDE.1	NBE2, LNE3, PRL2, LNEL2, PRI2, DATI2, PRF1	[OK]: Successo (Misto: <i>Con Luogo, ma senza Data Fine</i>)
TC11_GDE.1	NBE2, LNE3, PRL1, PRI2, DATI2, PRF2, COF2	[OK]: Successo (Misto: <i>Senza Luogo, ma con Data Fine</i>)



ID		
VRF_GDM.1		
Nome	Non vuota [NBN]	1. NBN = false [ERR] 2. NBN = true [PROPERTY_NB_OK]
Nome	Lunghezza [LNEN]	1. LNEN < 2 [ERR] 2. LNEN > 50 [ERR] 3. LNEN >= 2 && LNE <= 50 [PROPERTY_LNE_OK]
Quantità	Valore [VALQ]	1. VAL < 0 [ERR] 2. VAL = NULL [ERR] 3. VAL >= 0 [PROPERTY_VAL_OK]
Quantità	Formato [FORQ]	1. FORQ = false [ERR] 2. FORQ = true [PROPERTY_FORQ_OK]
Prezzo	Valore [VALP]	1. VALP < 0 [ERR] 2. VALP >= 0 [PROPERTY_VALP_OK]
Prezzo	Formato [FORP]	1. FORP = false [ERR] 2. FORP = true [PROPERTY_FORP_OK]
Descrizione	Lunghezza [LNED]	1. LNED > 255 [ERR] 2. LNED <= 255 [PROPERTY_LNED_OK]
Categoria	Lunghezza [LNEC]	1. LNEC > 255 [ERR] 2. LNEC <= 255 [PROPERTY_LNEC_OK]



ID		
TF_GDM.1		
TC1_GDM.1	NBN1	[ERR]: Campo nome vuoto
TC2_GDM.1	NBN2, LNEN1	[ERR]: Lunghezza nome < 2 caratteri
TC3_GDM.1	NBN2, LNEN2	[ERR]: Lunghezza nome > 50 caratteri
TC4_GDM.1	NBN2, LNEN3, VALQ1	[ERR]: Quantità negativa (< 0)
TC5_GDM.1	NBN2, LNEN3, VALQ2	[ERR]: Quantità nulla (campo vuoto)
TC6_GDM.1	NBN2, LNEN3, VALQ3, FORQ1	[ERR]: Formato Quantità non valido
TC7_GDM.1	NBN2, LNEN3, VALQ3, FORQ2, VALP1	[ERR]: Prezzo negativo (< 0)
TC8_GDM.1	NBN2, LNEN3, VALQ3, FORQ2, VALP2, FORP1	[ERR]: Formato Prezzo non valido
TC9_GDM.1	NBN2, LNEN3, VALQ3, FORQ2, VALP2, FORP2, LNED1	[ERR]: Lunghezza descrizione > 255 caratteri
TC10_GDM.1	NBN2, LNEN3, VALQ3, FORQ2, VALP2, FORP2, LNED2, LNEC1	[ERR]: Lunghezza categoria > 255 caratteri
TC11_GDM.1	NBN2, LNEN3, VALQ3, FORQ2, VALP2, FORP2, LNED2, LNEC2	[OK]: Prodotto aggiunto con successo