

Università degli Studi di Salerno
Corso di Ingegneria del Software

**Future Forge Gear
Test Plan
Versione 1.0**



Data: 22/12/2025

Progetto: Future Forge Gear	Versione: 1.0
Documento: Test Plan	Data: 19/12/2025

Coordinatore del progetto:

Nome	Matricola

Partecipanti:

Nome	Matricola
Francesco Mascolo	0512117352
Francesco Lamanna	0512117166
Luigi Mazza	0512118612

Scritto da:	Francesco Mascolo
--------------------	-------------------

Revision History

Data	Versione	Descrizione	Autore
19/12/2025	0.1	Creazione del Documento Test Plan	Full Team
21/12/2025	0.2	Definizione Testing Schedule	Full Team
22/12/2025	1.0	Revisione documento	Full Team

Progetto: Future Forge Gear	Versione: 1.0
Documento: Test Plan	Data: 22/12/2025

Indice

1. Introduzione	4
1.1.Riferimenti	4
2. Relazione con altri documenti.....	4
3. Panoramica Del Sistema	5
4. Caratteristiche Da Testare	6
4.1 caratteristiche da testare:	6
5. Pass/Fail Criteria	6
6. Approccio	7
7. Sospensione e Ripresa.....	8
8. Testing Materials.....	8
9. Test Case	8
10. Testing Schedule	8

Progetto: Future Forge Gear	Versione: 1.0
Documento: Test Plan	Data: 22/12/2025

1. Introduzione

Il seguente documento Test Plan definisce le strategie, i metodi e le risorse necessarie per eseguire il testing del sistema Future Forge Gear.

L'obiettivo principale è garantire che il software soddisfi i requisiti funzionali e non funzionali definiti nel documento Requirements Analysis Document (RAD) – Future Forge Gear.

Il piano di test ha lo scopo di individuare eventuali errori, malfunzionamenti o bug del sistema attraverso l'esecuzione di test unitari, test di integrazione e test di sistema, al fine di fornire un prodotto finale stabile, affidabile e conforme alle aspettative del cliente.

Il sistema Future Forge Gear, progettato come piattaforma e-commerce per la vendita di prodotti informatici, comprende funzionalità quali la gestione del catalogo prodotti, del carrello, degli ordini, degli utenti e delle richieste di approvvigionamento.

Il sistema è suddiviso in diversi sottosistemi, come descritto nel System Design Document, ciascuno dei quali richiede specifiche attività di testing per garantire la correttezza, l'affidabilità e l'efficienza delle funzionalità offerte. I test saranno eseguiti secondo approcci bottom-up e black-box per verificare sia i componenti atomici sia l'interazione fra di esse.

1.1. Riferimenti

Documenti:

RAD_FutureForgeGear (Requirements Analysis Document): Sono descritte le funzionalità individuate in fase di analisi.

SDD_FutureForgeGear (System Design Document): Contiene una descrizione completa dell'architettura del sistema, inclusi i sottosistemi individuati ed i servizi forniti da ciascun sottosistema.

ODD_FutureForgeGear (Object Design Document): Include dettagli sulle classi, i metodi e le interfacce.

2. Relazione con altri documenti

RAD_FutureForgeGear (Requirements Analysis Document):

In questo documento sono descritte le funzionalità individuate in fase di analisi, fornendo una descrizione dettagliata dei requisiti funzionali e non funzionali del sistema. Il piano di test fa riferimento a tali requisiti per verificare che tutte le funzionalità richieste siano correttamente implementate e rispettate.

SDD_FutureForgeGear (System Design Document):

Il documento descrive in modo completo l'architettura del sistema, basata su un modello three-tier, includendo i sottosistemi individuati e i servizi forniti da ciascun sottosistema. I test definiti nel presente piano sono progettati per verificare il corretto funzionamento dei singoli sottosistemi e la corretta interazione tra di essi, in accordo con quanto specificato nel SDD.

ODD_FutureForgeGear (Object Design Document):

Questo documento include i dettagli relativi alle classi, alle interfacce e ai metodi del sistema. Il Test

Progetto: Future Forge Gear	Versione: 1.0
Documento: Test Plan	Data: 22/12/2025

Plan utilizza le specifiche contenute nell'ODD per la definizione dei test unitari, finalizzati alla verifica delle singole classi, e dei test di integrazione, volti a controllare la corretta cooperazione tra i componenti dei sottosistemi.

3. Panoramica Del Sistema

Il sistema Future Forge Gear è stato progettato per gestire le operazioni relative alla vendita online di prodotti informatici, includendo la gestione del catalogo prodotti, del carrello, degli ordini, degli utenti e delle richieste di approvvigionamento verso i fornitori.

L'architettura proposta del sistema, come delineato nel System Design Document (SDD), si basa su un modello three-tier, che separa nettamente il livello di presentazione, il livello di logica di business e il livello di persistenza dei dati. Questa suddivisione consente di migliorare la manutenibilità, la modularità e la scalabilità del sistema, rendendolo inoltre più semplice da testare e da aggiornare.

Il Presentation Layer del sistema Future Forge Gear è dedicato all'interazione con l'utente finale. Attraverso un'interfaccia grafica responsive, realizzata mediante tecnologie web quali JSP, HTML5, CSS e JavaScript, gli utenti possono accedere alle principali funzionalità della piattaforma, tra cui la navigazione del catalogo prodotti, l'autenticazione, la gestione del carrello, la procedura di checkout e la consultazione dello storico degli ordini.

Gli utenti con ruoli specifici (come magazziniere, fornitore o gestore degli ordini) possono inoltre accedere a funzionalità avanzate quali la gestione dei prodotti, l'invio e la gestione delle richieste di approvvigionamento e l'aggiornamento dello stato degli ordini. Le pagine JSP sono strutturate per comunicare con il livello applicativo e per gestire in modo dinamico i dati visualizzati.

L'Application Layer (Denominato logica di Business) rappresenta il cuore della logica di business del sistema. Questo livello funge da intermediario tra il Presentation Layer e il Data Layer, occupandosi di elaborare le richieste provenienti dall'interfaccia utente e di coordinare le operazioni applicative. In questo strato sono implementati i servizi applicativi descritti nel SDD (ad esempio AccountService, ProdottoService, OrdineService, RichiestaService), che incapsulano la logica di business e garantiscono la corretta esecuzione delle funzionalità del sistema.

Il Data Layer, infine, si occupa della persistenza dei dati.

Questo livello utilizza un database relazionale MySQL per memorizzare in modo strutturato le informazioni relative alle principali entità del sistema, come utenti, prodotti, ordini e richieste. La gestione delle operazioni di lettura e scrittura dei dati è affidata a classi DAO (ad esempio UtenteDAO, ProdottoDAO, OrdineDAO, RichiestaDAO), che interagiscono direttamente con il database, garantendo l'integrità e la consistenza dei dati.

Il flusso di comunicazione tra i livelli segue il modello three-tier: il Presentation Layer invia le

Progetto: Future Forge Gear	Versione: 1.0
Documento: Test Plan	Data: 22/12/2025

richieste all'Application Layer, che le elabora e interagisce con il Data Layer per recuperare o salvare i dati richiesti, assicurando una chiara separazione delle responsabilità tra i diversi componenti del sistema.

4. Caratteristiche Da Testare

4.1 caratteristiche da testare:

Funzionalità dell'Utente:

- Validazione di Email nella registrazione
- Validazione di password nella registrazione
- Aggiunta di prodotti al carrello
- Login tramite mail e password

Funzionalità del magazziniere:

- Richiesta di un prodotto
- Rimozione di un prodotto dal catalogo
- Modifica del nome di un prodotto nel magazzino
- Modifica del prezzo di un prodotto nel magazzino

Funzionalità del fornitore:

- Creazione di un prodotto

Funzionalità del gestore degli ordini:

- Modifica stato dell'ordine

5. Pass/Fail Criteria

L'obiettivo dell'attività di testing è garantire che il sistema Future Forge Gear soddisfi pienamente i requisiti funzionali (RF) e i requisiti non funzionali (RNF) definiti nella documentazione di progetto. Ogni test è finalizzato a verificare che le funzionalità implementate operino correttamente e producano risultati coerenti rispetto agli input forniti. Il processo di testing si basa sul confronto tra il risultato atteso e il risultato effettivo (Oracolo) dell'operazione eseguita. Un test è considerato superato (PASS) se il comportamento osservato del sistema coincide con quello atteso. Al contrario, il test è considerato fallito (FAIL) nel caso in cui si riscontrino discrepanze tra il risultato ottenuto e quello previsto. L'individuazione degli errori avviene mediante il confronto con un oracolo di test, che rappresenta il comportamento corretto del sistema in base ai requisiti specificati. Qualora il sistema non soddisfi i criteri sopra elencati, sarà necessario procedere alla

Progetto: Future Forge Gear	Versione: 1.0
Documento: Test Plan	Data: 22/12/2025

correzione del codice, alla riesecuzione dei test e alla successiva verifica, fino a garantire il pieno rispetto di tutti i requisiti definiti.

6. Approccio

L'approccio al testing per il sistema Future Forge Gear è articolato in tre fasi principali, finalizzate a garantire una verifica progressiva del sistema, partendo dalle singole componenti fino alla valutazione del comportamento complessivo della piattaforma.

Testing Unitario

La prima fase è rappresentata dal testing unitario, focalizzato sulla verifica delle singole classi e dei metodi che compongono il sistema. In questa fase viene adottata una metodologia di testing black-box, in cui l'attenzione è rivolta esclusivamente agli input e agli output delle componenti, senza considerare la loro struttura interna. Per gestire la varietà dei possibili dati di ingresso, viene applicata la tecnica di category partition, che consente di suddividere gli input in categorie rappresentative di classi di equivalenza. Questo approccio permette di individuare tempestivamente eventuali anomalie nelle singole componenti, che vengono successivamente corrette e ritestate, garantendo la conformità ai requisiti definiti nel RAD.

Testing di Integrazione

La seconda fase consiste nel testing di integrazione, mirato a verificare le interazioni tra le diverse componenti del sistema.

L'obiettivo principale è assicurare che i moduli cooperino correttamente tra loro, evitando errori dovuti all'integrazione delle varie parti applicative.

In particolare, vengono considerate le interazioni tra le componenti responsabili della gestione degli utenti, dei prodotti, del carrello, degli ordini e delle richieste di approvvigionamento.

Per l'esecuzione dei test di integrazione viene adottato un approccio bottom-up, partendo dalle classi DAO del Data Layer e integrandole progressivamente con i servizi applicativi del Business Layer che ne fanno uso.

Testing di Sistema

La fase di testing di sistema è finalizzata alla verifica del comportamento complessivo della piattaforma Future Forge Gear, valutando il soddisfacimento dei principali requisiti funzionali definiti nel RAD. I test di sistema sono orientati alla validazione dei principali flussi di utilizzo dell'applicazione dal punto di vista dell'utente, come la gestione del catalogo, del carrello e degli ordini, considerando il sistema come un'unica entità integrata.

Questa fase consente di individuare eventuali anomalie residue che possono emergere solo a seguito dell'integrazione completa dei moduli, fornendo una verifica finale del corretto funzionamento complessivo del sistema.

Progetto: Future Forge Gear	Versione: 1.0
Documento: Test Plan	Data: 22/12/2025

7. Sospensione e Ripresa

La fase di testing del sistema Future Forge Gear può essere sospesa qualora si verifichino condizioni che impediscano il corretto proseguimento delle attività di verifica.

Le principali cause di sospensione includono:

Rilevazione di bug critici o bloccanti:

malfunzionamenti che compromettono le funzionalità principali del sistema, come errori applicativi gravi o anomalie nella gestione dei dati, tali da impedire l'esecuzione dei test pianificati.

Modifiche al sistema non ancora stabilizzate:

aggiornamenti o cambiamenti al codice che non siano stati ancora adeguatamente integrati o verificati, rendendo non attendibili i risultati dei test.

Ripresa delle attività di test (Recovery)

La ripresa delle attività di testing è subordinata alla risoluzione delle cause che hanno determinato la sospensione. In particolare, prima di procedere con nuove esecuzioni di test, è necessario:
Correggere i bug bloccanti individuati, seguiti da una riesecuzione mirata dei test interessati;
Eseguire test di regressione sui componenti coinvolti, al fine di verificare che le correzioni apportate non abbiano introdotto nuove anomalie.

8. Testing Materials

Gli strumenti utilizzati per il testing sono:

- **Testing d'unità:** JUnit e Mockito
- **Test d'integrazione:** SeleniumIDE

9. Test Case

I singoli test case relativi a questa sezione sono disponibili nel documento **Test Case Specification (TCS)** – disponibile su GitHub al seguente indirizzo:

https://github.com/francesco-mascolo-19/FutureForgeGear/blob/main/Deliverables/TCS_FutureForgeGear.pdf.

10. Testing Schedule

Il programma dei test per il sistema Future Forge Gear è stato definito per garantire un processo di verifica strutturato e coerente con le tempistiche di sviluppo.

Le attività di testing sono organizzate in tre fasi principali, ciascuna dedicata a un diverso livello di

Progetto: Future Forge Gear	Versione: 1.0
Documento: Test Plan	Data: 22/12/2025

verifica del sistema.

Testing Unitario

Durata: 1 settimana

Verifica delle singole classi e dei metodi implementati, eseguita in parallelo allo sviluppo del codice per individuare tempestivamente eventuali anomalie.

Testing di Integrazione

Durata: 1 settimana

Verifica delle interazioni tra i moduli del sistema, con particolare attenzione alle dipendenze e alla corretta cooperazione tra le componenti applicative.

Testing di Sistema

Durata: fase finale del progetto

Validazione del comportamento complessivo della piattaforma Future Forge Gear, attraverso la verifica dei principali flussi funzionali dal punto di vista dell'utente.

Al termine di ciascuna fase è prevista una revisione dei risultati, con eventuale correzione delle anomalie prima del passaggio alla fase successiva e una riesecuzione mirata dei test nella fase finale.