Test Plan

Document

GoldenOnlineStore

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Data** | **Versione** | **Descrizione** | **Autori** |
| 27/11/2023 | 0.1 | Prima stesura | Antonio Spatuzzi  Vittorio Giordano |
| 6/12/2023 | 0.2 | Aggiunta Test Case | Vittorio Giordano |
| 10/12/2023 | 0.3 | Aggiunta Test Case | Antonio Spatuzzi |
| 20/01/2024 | 0.4 | Revisione Documento | Tutti i membri |

[1. Introduction 4](#_Toc157785668)

[2. Relations with other Documents 4](#_Toc157785669)

[3. System overview 4](#_Toc157785670)

[4. Features to be tested/Not to be tested 4](#_Toc157785671)

[4.1 System functionality 4](#_Toc157785672)

[4.2 Non-Functional Features 5](#_Toc157785673)

[5. Pass/Fail criteria 5](#_Toc157785674)

[6. Approach 6](#_Toc157785675)

[7. Testing materials 7](#_Toc157785676)

[7.1 Hardware 7](#_Toc157785677)

[7.1.1 Server di Test 7](#_Toc157785678)

[7.1.2 Rete di Test 7](#_Toc157785679)

[7.1.3 Risorse di Storage 7](#_Toc157785680)

[7.2 Software 7](#_Toc157785681)

[7.2.1 Ambiente di Sviluppo Integrato (IDE): 7](#_Toc157785682)

[7.2.2 Navigatori Web: 8](#_Toc157785683)

[7.2.3 Strumenti di Automazione del Testing 8](#_Toc157785684)

[7.2.4 Sistemi di Gestione del Database 8](#_Toc157785685)

[7.2.5 Strumenti di Controllo di Versione: 8](#_Toc157785686)

[8 Test Cases 8](#_Toc157785687)

[9. Testing Schedule 8](#_Toc157785688)

[9.1 Attività e Compiti 8](#_Toc157785689)

[9.2 Pianificazione Temporale 9](#_Toc157785690)

[9.3 Revisione e Adattamento 9](#_Toc157785691)

# Introduction

Il presente Test Plan ha lo scopo di definire la strategia di testing e le attività correlate per il progetto GOS.

Questo documento fornirà una guida chiara per il team di testing sulle attività di testing pianificate.

# Relations with other Documents

* RAD (Requirements Analysis Document)
* SDD (System Design Document)
* SOW (Statement of Work)
* TCS (Test Case Specification)
* TSI (Test Summary and Incident)

# System overview

Il sistema GOS si presenta come uno strumento per una fumetteria di media/piccola taglia. che punta a migliorare le proprie vendite in loco ed online. GoldenOnlineStore intende anche semplificare l’acquisto di prodotti da parte di possibili utenti interessati.

# Features to be tested/Not to be tested

## System functionality

Verranno sottoposte a test la funzionalità di “*Acquisto di un prodotto*”, “*Cancellazione di un ordine*”, “*Cancellazione di un prodotto*”, “*Inserimento di un prodotto*”, “*Modifica di un prodotto*” ,“*Registrazione*” ,“*Modifica Profilo*”, “*Cancellazione Profilo*”.

1. **Acquisto di un prodotto**: Questa funzionalità verrà testata per garantire che il processo di acquisto funzioni correttamente e senza problemi per gli utenti. I test saranno eseguiti per verificare che gli utenti possano aggiungere prodotti al carrello, procedere al pagamento in modo corretto e completare l'acquisto con successo.

2. **Cancellazione di un ordine**: Questa funzionalità verrà testata per garantire che gli utenti possano cancellare gli ordini in modo semplice ed efficiente, rispettando le politiche di cancellazione dell'azienda. I test saranno eseguiti per verificare che gli ordini possano essere cancellati prima della spedizione.

3. **Cancellazione di un prodotto**: Questa funzionalità verrà testata per verificare che gli amministratori possano rimuovere correttamente i prodotti dal catalogo dell'e-commerce.

4. **Inserimento di un prodotto**: Questa funzionalità verrà testata per garantire che gli amministratori possano aggiungere nuovi prodotti al catalogo dell'e-commerce in modo corretto e completo. I test saranno eseguiti per verificare che tutte le informazioni necessarie vengano inserite correttamente.

5. **Modifica di un prodotto**: Questa funzionalità verrà testata per garantire che gli amministratori possano modificare correttamente le informazioni esistenti sui prodotti nel catalogo dell'e-commerce.

6. **Registrazione**: Questa funzionalità verrà testata per garantire che gli utenti possano registrarsi correttamente sul sito web dell'e-commerce.

7. **Modifica Profilo**: Questa funzionalità verrà testata per garantire che gli utenti possano modificare le informazioni del loro profilo utente (ad esempio, email o password) in modo corretto.

8. **Cancellazione Profilo**: Questa funzionalità verrà testata per garantire che gli utenti possano cancellare il proprio profilo utente dal sistema in modo corretto e completo.

## Non-Functional Features

**Usabilità**:Il sistema sarà sottoposto a test al fine di garantire che sia intuitivo e di facile comprensione per l'utente finale.

**Compatibilità**: Il sistema verrà testato su differenti browser per studiarne il comportamento atteso;

**Performance**: Il sistema sarà soggetto a test al fine di garantire tempi di risposta accettabili.

# Pass/Fail criteria

**Testing di Unità:**

* Verrà verificato che le unità comunichino correttamente con il database.
* Un test di unità viene considerato fallito se uno o più test case associati non soddisfano i criteri di successo definiti per quel test case.
  + Questi criteri di successo possono includere aspetti come :
    - il corretto funzionamento di una specifica funzionalità
    - il rispetto di determinate condizioni di input o output
    - il comportamento conforme alle specifiche
  + Se uno qualsiasi di questi criteri non è soddisfatto durante l'esecuzione del test, il test di unità viene considerato fallito.

**Testing di Integrazione:**

* Verrà verificato se le unità, precedentemente testate individualmente, interagiscono in modo idoneo tra di loro.
* Un test di integrazione viene considerato fallito se il sistema integrato non soddisfa i requisiti di integrazione specificati.
  + Questi requisiti di integrazione possono includere aspetti come :
    - la comunicazione corretta tra le componenti
    - lo scambio di dati accurato
    - il comportamento corretto nei flussi di lavoro integrati
  + Se uno qualsiasi di questi criteri non è soddisfatto durante l'esecuzione del test, il testing di integrazione viene considerato fallito.

**Testing di Sistema**:

* Verranno verificati diversi aspetti del sistema software al fine di garantire che soddisfi i requisiti e le aspettative degli utenti, come il Testing Funzionale o il Testing di Accettazione.
* **Testing Funzionale**:
  + Verrà verificato che ogni funzionalità inizialmente definita nei documenti sia stata effettivamente implementata e che il sistema agisca esattamente come previsto.
* **Testing di Accettazione:**
  + Verrà verificato che il sistema soddisfi pienamente le esigenze e le aspettative del cliente.

# Approach

**Testing di unità:**

* Nuovo Acquisto;
* Cancellazione Ordine;
* Aggiunta Prodotto;
* Cancellazione Prodotto;
* Modifica Prodotto;
* Registrazione Utente;
* Cancellazione Profilo;
* Modifica Profilo;

In questo testing utilizzeremo JUnit come framework di testing e Mockito per la simulazione delle dipendenze esterne. In questo testing verrà testata l’interazione delle singole unità con il database;

L'approccio al testing delle unità sarà orientato alla tecnica White-Box per garantire un approccio strutturale al testing.

* AcquistoProdottoServlet;
* AggiuntaProdottoServlet;
* AnnullaOrdineServlet;
* CancellaProfiloServlet;
* ModificaDatiServlet;
* ModificaProdottoServlet;
* RegistrazioneServlet;
* RimuoviProdottoServlet;

In questo testing utilizzeremo JUnit come framework di testing e Mockito per la simulazione delle dipendenze esterne. Verranno testate le Servlet per garantire il corretto funzionamento di quest’ultime.

**Testing di Integrazione:**

* Acquisto e Visualizza Ordine;
* Acquisto e Cancellazione Ordine;
* Aggiunta e Cancellazione Prodotto;
* Aggiunta e Modifica Prodotto;
* Registrazione e Cancellazione Profilo;
* Registrazione e Modifica Profilo;

In questo testing utilizzeremo JUnit come framework di testing e Mockito per la simulazione delle dipendenze esterne. In questo testing verrà testata l’interazione tra i vari sottosistemi.

L'approccio al testing di Integrazione sarà orientato alla tecnica White-Box per garantire un approccio strutturale al testing.

**Testing di Performance:**

Verranno esaminati e valutati i tempi di risposta del sistema. Questi test mirano a garantire che il sistema sia in grado di mantenere prestazioni accettabili.

Il test di Perfomance verrà eseguito simulando le interazioni dell'utente con Selenium.

Selenium simula le azioni dell'utente come clic sui pulsanti, inserimento di dati e navigazione tra le pagine. Durante l'esecuzione, vengono rilevati eventuali difetti o comportamenti anomali, che vengono poi documentati e risolti. Questo processo assicura che il sistema sia pronto per la distribuzione e soddisfi i requisiti funzionali e non funzionali.

**Pilot Testing:**

Alpha Test: Il sistema sarà sottoposto a test da parte di personale interno all'ambiente di sviluppo.

* L'obiettivo principale di questo test è quello di identificare eventuali difetti, problemi di funzionalità o discrepanze rispetto ai requisiti specificati prima di procedere con fasi successive di testing e distribuzione.

Beta Test: Il sistema sarà sottoposto a test da parte di personale esterno all'ambiente di sviluppo

* L'obiettivo principale di questo test è quello di valutare l'accessibilità e la facilità di navigazione e garantire che il sistema sia intuitivo e facile da usare per gli utenti finali.

# Testing materials

## Hardware

Per garantire un ambiente di testing efficiente e rappresentativo sarà necessario disporre di hardware affidabile.

Le risorse hardware richieste includono:

### 7.1.1 Server di Test

* Un server dedicato per ospitare l’ambiente di test del sito web, simile alle specifiche del server di produzione.

### 7.1.2 Rete di Test

* Una rete locale dedicata per simulare varie condizioni di connettività e testare la risposta del sito web in diverse situazioni di rete.

### 7.1.3 Risorse di Storage

* Spazio di archiviazione sufficiente per conservare dati di test, registrazioni e risultati dei test.

## Software

L’elenco di software necessario è progettato per supportare il testing garantendo una copertura completa delle funzionalità del sito web.

### 7.2.1 Ambiente di Sviluppo Integrato (IDE):

* Utilizzo di IntelliJ IDEA Ultimate per lo sviluppo e il debugging del codice.

### 7.2.2 Navigatori Web:

* Le versioni più recenti dei principali browser come Google Chrome, Mozilla Firefox, Microsoft Edge e Safari per testare la compatibilità del sito.

### 7.2.3 Strumenti di Automazione del Testing

* Framework di automazione del testing come Selenium per test automatizzati di funzionalità critiche del sito

### 7.2.4 Sistemi di Gestione del Database

* Un sistema di gestione di database DBMS compatibile con il database del sito web MySQL.

### 7.2.5 Strumenti di Controllo di Versione:

* Strumenti di controllo di versione come Git per gestire il codice sorgente e collaborare efficacemente con il team di sviluppo.

# Test Cases

Per sviluppare i casi di test, adotteremo il metodo del Category Partitioning. Questo approccio prevede l'identificazione di parametri per ciascuna funzionalità da testare. Ogni parametro verrà suddiviso in categorie, ciascuna delle quali con diverse opzioni.

Vedere il documento di Test Case Specification.

# Testing Schedule

Il processo dei test sarà suddiviso in diverse fasi, ognuna delle quali si concentrerà su specifiche aree del prodotto per garantire una copertura completa. Le fasi di test includeranno:

* Test di Unità: Questa fase si concentrerà sull’analisi delle singole unità del software.
* Test di Integrazione: In questa fase testeremo l’integrazione delle diverse unità per garantire che lavorino insieme correttamente.
* Test di Sistema: Questa fase si concentrerà sul test del sistema nel suo complesso, garantendo che tutte le parti interagiscano senza problemi.

## 9.1 Attività e Compiti

Verranno eseguite le seguenti attività durante ciascuna fase del processo di test:

* Identificazione dei casi di test
* Esecuzione dei test
* Registrazione e monitoraggio dei difetti

## 9.2 Pianificazione Temporale

La pianificazione temporale per le attività di test sarà la seguente:

* Test di Unità: [03/01/2024] - [07/01/2024]
* Test di Integrazione: [08/01/2024] - [12/01/2024]
* Test di Sistema: [13/01/2024] - [17/01/2024]

## 9.3 Revisione e Adattamento

Durante il corso del progetto, la pianificazione sarà soggetta a revisioni periodiche per adattarsi a eventuali cambiamenti nelle specifiche o nelle priorità del progetto. Le revisioni saranno pianificate in modo da minimizzare l'impatto sulle attività di test.