

# Test Plan

## TC\_TechShop

### Sommario

<b>1.</b>	<b>Introduction</b>	2
<b>2.</b>	<b>Relationship to other documents</b>	2
<b>3.</b>	<b>System overview</b>	2
<b>4.</b>	<b>Features to be tested/not to be tested</b>	2
<b>5.</b>	<b>Approach</b>	3
<b>6.</b>	<b>Suspension and Resumption</b>	3
<b>7.</b>	<b>Testing materials (Hardware/Software)</b>	4
<b>8.</b>	<b>Test Cases</b>	4
<b>9.</b>	<b>Pianificazione del testing</b>	6

## 1. Introduction

Questo documento ha lo scopo di descrivere l'ambito, gli approcci e le risorse utilizzate durante tutte le attività di testing del progetto software TechShop.

L'obiettivo di questo documento è di presentare le varie attività di testing da svolgere su "TechShop", con gli specifici vincoli e processi pianificati. Essendo quasi impossibile sviluppare un'applicazione software priva di errori, è indispensabile dedicare tempo a trovarli e correggerli attraverso un adeguato processo di testing.

## 2. Relationship to other documents

*Il test plan si basa sulla conoscenza e le informazioni acquisite attraverso tutti i documenti, quali:*

[PS TechShop.docx](#)

[RAD TechShop.docx](#)

[SDD TechShop.docx](#)

[ODD Tech Shop.docx](#)

<http://java.sun.com>

*Illustra il modo in cui tutti i test sono correlati ai requisiti funzionali e non funzionali, nonché alla progettazione del sistema indicata nei rispettivi documenti.*

## 3. System overview

*System overview, incentrata sugli aspetti strutturali dei test, fornisce una panoramica del sistema in termini di componenti che vengono testati durante il test unitario. La granularità dei componenti e le loro dipendenze sono definite in questa sezione.*

Lo scopo offerto dalla piattaforma era Quello di offrire la possibilità agli utenti di creare il sito web del proprio negozio, avendo la possibilità anche di interfacciarsi attraverso la stessa piattaforma con gli i propri clienti. Tenendo considerazione dei trade off come visto già in precedenza nel documento di Object Design, il team ha implementato le funzionalità inerenti all'utente identificato come Venditore.

Lo scopo principale del venditore era quello di poter creare un proprio negozio di e-commerce sulla piattaforma TechShop, con aggiunta di categorie, prodotti e la possibilità di modificarli.

## 4. Features to be tested/not to be tested

*Questa sezione si incentra sugli aspetti funzionali dei test, identifica tutte le funzionalità e le combinazioni di funzionalità da testare. Inoltre tutte quelle funzionalità che non devono essere testate e i motivi per cui non sono state testate.*

*Dopo aver avviato il sistema sulla nostra macchina abbiamo simulato quello che il venditore dovrebbe fare, e abbiamo individuato le funzionalità da testare e non da testare.*

#### **Funzionalità da testare:**

- Login Venditore
- Registrazione venditore
- Creazione del negozio
- Inserisci categoria
- Inserisci prodotto
- Modifica categoria
- Cancellazione categoria
- Modifica prodotto
- Cancellazione prodotto

*Caricamento dell'immagine del negozio, delle relative categorie e prodotti, questo è stato escluso dal testing per motivi di tempo ma per il semplice motivo che al caricamento dell'immagine nel database veniva solo memorizzato il path. La composizione del nome della foto inserito dipendesse dal nome del negozio, e un enumerazione per determinare le categorie e prodotti. Il problema era che il testing funzionasse con il path della posizione del server di Apache installato nel proprio pc quindi per non avere problemi in fase di installazione si è evitato questo di testare queste funzionalità.*

*Visualizza ordini perché il sottosistema cliente non è stato implementato quindi non è possibile trovare tutti i possibili errori, era inutile caricare nel database dati solo per visualizzarli nel testing.*

## **5. Approach**

Il testing verrà svolto utilizzando un approccio Bottom-Up.

Partiremo testando le componenti a più basso livello, in questo caso le classi riferite ai DAO dei vari sottosistemi.

Successivamente si procederà con l'integrazione verso i controller con i quali i vari model interagiscono, effettuando testing blackbox (utilizzando il category partition) e whitebox (definendo i test case effettuando una revisione del codice utilizzando la tecnica del branch-coverage).

Infine si procederà con il testing di sistema, attraverso l'uso di Selenium, per testare l'interazione delle componenti al fine di testare una funzionalità, inclusa la parte front-end del sistema.

## **6. Suspension and Resumption**

Il test è stato sospeso e ripreso più volte per consentire ai programmatori (che sono, ovviamente, esclusi dalla fase di test) di correggere le anomalie individuate al momento nella fase di testing.

## 7. Testing materials (Hardware/Software)

Il testing è stato effettuato utilizzando JUnit, Mockito e Selenium.

## 8. Test Cases

I casi di test sono definiti nel documento TestCaseSpecification.

Alcuni esempi sono riportati di seguito

### ***Test Registrazione \_venditore***

<b>Parametro: Nome</b>	
<b>Formato: [A-Z/a-z]</b>	
<b>Lunghezza [LN]:</b>	1. <1 or > 25[error] 2. >=1and <=25[property LN_OK]
<b>Formato [FN]:</b>	1. Non match [if LN_OK] error 2. Match [if LN_OK][property FN_OK]
<b>Parametro: Cognome</b>	
<b>Formato: [A-Z/a-z]</b>	
<b>Lunghezza [LN]:</b>	1. <1 or > 25[error] 2. >=1 and <=25[property LN_OK]
<b>Formato [FN]:</b>	1. Non match [if LN_OK] error 2. Match [if LN_OK][property FN_OK]
<b>Parametro : Sesso</b>	
<b>Formato:[Uomo-Donna-Altro]</b>	
<b>Selezionato[SD]:</b>	1.non selezionato[error] 2.selezionato [property SD_OK]
<b>Parametro: E-mail</b>	
<b>Formato:[A-Z/a-z][Numeri ][@]</b>	
<b>Parametri[PR]:</b>	1. Non inserito @ or dominio (.)[error] 2. Parametri rispettati[PR_OK]
<b>Lunghezza[LN]:</b>	1 .Lunghezza variabile [property LN_OK]
<b>Parametro: Telefono</b>	
<b>Formato: 10 Numeri[ ]</b>	
<b>Lunghezza[LN]:</b>	1.<1 or>10[error] 2.Match[LN_OK=10]
<b>Parametro: Via</b>	
<b>Formato:[A-Z/a-z]</b>	

<b>Lunghezza[LN]:</b>	1.Lungezza variabile[property LN_OK]
<b>Parametro: Città</b>	
<b>Formato:[A-Z/a-z]</b>	
<b>Lunghezza[LN]:</b>	1.Lungezza variabile[property LN_OK]
<b>Parametro: CAP</b>	
<b>Formato:5 Numeri[]</b>	
<b>Lunghezza[LN]:</b>	1.<1 or >5 [error] 2.Match[LN_OK=5]
<b>Parametro: Username</b>	
<b>Formato:[A-Z/a-z]</b>	
<b>Lunghezza[LN]:</b>	1.< 3 or >16[error] 2.>=3 and<=16 [property LN_OK]
<b>Parametro: Password</b>	
<b>Formato:[A-Z/a-z][0-9][]</b>	
<b>Lunghezza[LN]:</b>	1.Lunghezza variabile[property LN_OK]

### ***Tabella Registrazione\_cliente***

<b>Parametro: Nome</b>	
<b>Formato: [A-Z/a-z]</b>	
<b>Lunghezza [LN]:</b>	3. <1 or > 25[error] 4. >=1and <=25[property LN_OK]
<b>Formato [FN]:</b>	3. Non match [if LN_OK] error 4. Match [if LN_OK][property FN_OK]
<b>Parametro: Cognome</b>	
<b>Formato: [A-Z/a-z]</b>	
<b>Lunghezza [LN]:</b>	3. <1 or > 25[error] 4. >=1 and <=25[property LN_OK]
<b>Formato [FN]:</b>	3. Non match [if LN_OK] error 4. Match [if LN_OK][property FN_OK]
<b>Parametro : Sesso</b>	
<b>Formato:[Uomo-Donna-Altro]</b>	
<b>Selezionato[SD]:</b>	1.non selezionato[error] 2.selezionato [property SD_OK]
<b>Parametro: E-mail</b>	
<b>Formato:[A-Z/a-z][Numeri ][@]</b>	
<b>Parametri[PR]:</b>	3. Non inserito @ or dominio (.)[error] 4. Parametri rispettati[PR_OK]
<b>Lunghezza[LN]:</b>	1 .Lunghezza variabile [property LN_OK]
<b>Parametro: Telefono</b>	
<b>Formato: 10 Numeri[]</b>	
<b>Lunghezza[LN]:</b>	1.<1 or>10[error]

	2.Match[LN_OK=10]
<b>Parametro: Via</b>	
<b>Formato:[A-Z/a-z]</b>	
<b>Lunghezza[LN]:</b>	1.Lunghezza variabile[property LN_OK]
<b>Parametro: Città</b>	
<b>Formato:[A-Z/a-z]</b>	
<b>Lunghezza[LN]:</b>	1.Lunghezza variabile[property LN_OK]
<b>Parametro: CAP</b>	
<b>Formato:5 Numeri[]</b>	
<b>Lunghezza[LN]:</b>	1.<1 or >5 [error] 2.Match[LN_OK=5]
<b>Parametro: Username</b>	
<b>Formato:[A-Z/a-z]</b>	
<b>Lunghezza[LN]:</b>	1.< 3 or >16[error] 2.>=3 and<=16 [property LN_OK]
<b>Parametro: Password</b>	
<b>Formato:[A-Z/a-z][0-9][]</b>	
<b>Lunghezza[LN]:</b>	1.Lunghezza variabile[property LN_OK]

## 9. Pianificazione del testing

Il testing di unità si concentrare sulle classi DAO e le classi che le chiamano.

Il testing di integrazione è effettuato secondo il criterio di integrazione Bottum-up.

Per il testing delle varie funzionalità si è scelta la tecnica Black-Box "Category Partition".