

Università degli Studi di Salerno

Corso di Ingegneria Del Software

# BEerHAPPY

Test Plan  
Document

**Docente:**

Prof. Andrea De Lucia

**Studenti:**

Giovanni Forlenza

## Sommario

- 1 Introduzione**
- 2 Relazione con altri documenti**
- 3 Funzionalità da testare e da non testare**
- 4 Criteri Pass/Failed**
- 5 Approccio**
- 6 Sospensione e ripresa**
- 7 Materiale per il Testing**
- 8 Test Case**

## 1. Introduzione

Lo scopo di questo documento è di pianificare l'attività di test del sistema "Beer HAPPY" al fine di verificare se esistono differenze tra il comportamento atteso e il comportamento osservato. In questa attività andremo a rilevare gli eventuali errori prodotti all'interno del codice, per evitare che essi si presentino nel momento in cui il sistema verrà utilizzato dall'utente finale. Le attività di test sono state pianificate per le seguenti gestioni:

- Gestione autenticazione
- Gestione account
- Gestione indirizzi
- Gestione metodi di pagamento
- Gestione catalogo
- Gestione utenti

Quasi ognuna di queste gestioni prevede principalmente operazioni di inserimento e modifica, saranno proprio queste funzionalità ad essere testate nel corso della fase di testing del sistema.

## 2. Relazione con altri documenti

Il Test Plan si basa sui documenti di analisi dei requisiti e di progettazione del sistema per definire le strategie di testing, le attività e le modalità di verifica necessarie per garantire che il software funzioni come previsto.

### Relazione con il documento di analisi dei requisiti (RAD)

Il Test Plan è un documento che descrive l'approccio e la strategia di testing che verranno utilizzati per verificare che il sistema o l'applicazione software funzioni correttamente e soddisfi i requisiti funzionali e non funzionali specificati nel RAD (documento di analisi dei requisiti).

### Relazioni con il System Design Document (SDD)

SDD (System Design Document) è un documento che descrive l'architettura del sistema software e fornisce dettagli su come il sistema verrà implementato. Il Test Plan utilizza le informazioni contenute nel SDD per comprendere l'architettura del sistema e per definire gli aspetti da testare.

### Relazione con l'Object Design Document (ODD)

L'ODD (Object Design Document) è un documento che descrive la progettazione dettagliata degli oggetti di sistema, come classi e moduli. Il Test Plan utilizza le informazioni contenute nell'ODD per definire i test unitari che verranno eseguiti per verificare che gli oggetti funzionino correttamente.

### 3. Funzionalità da testare e da non testare

Di seguito saranno elencate le funzionalità da testare per ogni gestione:

- Gestione autenticazione
  - Login
  - Registrazione
  - Recupero password
- Gestione Account
  - Modifica dati personali
  - Modifica password
- Gestione Indirizzi
  - Aggiunge un indirizzo
- Gestione metodo di pagamento
  - Aggiunge una carta
- Gestione catalogo
  - Aggiunge un prodotto
- Gestione utenti
  - Aggiunge un utente back-office

### 4. Criteri Pass/Failed

Il Test Plan deve definire chiaramente i criteri per determinare se un test viene superato o fallito, in modo da consentire una valutazione obiettiva dei risultati del testing.

I criteri pass/fail saranno definiti in base agli obiettivi specifici di ogni test e ai requisiti del software. I dati di input del test saranno suddivisi in insiemi dalle caratteristiche comuni, per i quali sarà sufficiente testare un solo elemento rappresentativo.

Un test viene considerato superato quando produce un risultato atteso, mentre verrà considerato fallito se il risultato ottenuto è diverso da quello atteso.

### 5. Approccio

L'approccio usato per testare il sistema è diviso in tre fasi: **Il test di unità** che ha lo scopo di testare le componenti del sistema singolarmente, poi si passa al **test di integrazione** che combina le componenti del sistema e testate come un unico gruppo, e il **testing di sistema** che ha come obbiettivo quello di verificare il funzionamento dell'intero sistema.

Per il testing di unità si utilizzerà la tecnica **Category Partition**.

La tecnica si basa sull'idea che i possibili input del sistema possano essere suddivisi in gruppi di equivalenza o categorie, dove ogni categoria rappresenta un insieme di input che si comportano in modo simile o producono risultati simili.

La suddivisione delle categorie verrà fatta in modo che gli input all'interno di una stessa categoria generi risultati simili e che gli input in categorie diverse generino risultati diversi.

In questo modo, il numero di casi di test necessari per coprire tutti i possibili input del sistema viene ridotto.

Per il testing di integrazione si utilizzerà la tecnica di **Testing incrementale**

I moduli del sistema vengono integrati gradualmente, partendo dai moduli più semplici e aggiungendo gradualmente quelli più complessi. In questo modo, è possibile identificare e correggere eventuali problemi di integrazione man mano che si procede con l'integrazione dei moduli.

Per il testing di sistema si utilizzerà la tecnica di **Testing funzionale**.

Il test funzionale viene eseguito per verificare che il sistema funzioni correttamente rispetto alle specifiche funzionali. Questo tipo di test può essere eseguito utilizzando casi di test basati su scenari di utilizzo realistici.

## 6. Sospensione e ripresa

**Sospensione:** La fase di testing verrà sospesa se si verificheranno problemi tecnici o se il software subirà modifiche sostanziali.

**Ripresa:** La ripresa del testing avverrà dopo che le modifiche saranno apportate ed eventuali errori risolti. I test saranno ripetuti per controllare che le modifiche non abbiano generato nuovi errori.

La sospensione e la ripresa del testing possono influire sulla pianificazione generale del progetto e sulla durata delle attività di testing. Queste attività verranno gestite in modo efficiente per minimizzare gli impatti sul progetto

## 7. Materiale per il Testing

Gli strumenti necessari per svolgere le attività di testing sono:

- WebServer Apache Tomcat 10 in locale che gira sul sistema
- Un client-web
- Un DBMS MySQL che gestisce l'utilizzo del database
- Selenium IDE per il test incrementale
- JUnit per il test di unità

## 8. Test case

TC\_0: Gestione autenticazione

TC\_0.1: Login

Parametro: E-mail	
Formato: /^\\w+([\\.-]?\\w+)*@\\w+([\\.-]?\\w+)*\\.\\w{2,3}+\$/;	
Lunghezza [LE]	1. < 5 or > 40 [Error] 2. >= 5 or < 40 [Property ok]
Formato [FE]	1. Non rispetta il formato 2. Rispetta il formato [Property ok]

Parametro: Password	
Match [MP]	1. Match con password utente = false [Error] 2. Match con password utente = true [Property ok]

Codice	Combinazione	Esito
TC 0.1.1	LE1	Error
TC 0.1.2	LE2, FE1	Error
TC 0.1.3	LE2, FE2, MP1	Error
TC 0.1.4	LE2, FE2, MP2	Success

TC\_0.2: Registrazione

Parametro: Nome	
Formato: \\d/;	
Lunghezza [LN]	1. >= 30 or ==0 [Error] 2. < 30 or > 0 [Property ok]
Formato [FN]	1. Rispetta il formato 2. Non rispetta il formato [Property ok]

Parametro: Cognome Formato: \d/;	
Lunghezza [LC]	1. >= 30 or ==0 [Error] 2. < 30 or > 0 [Property ok]
Formato [FC]	1. Rispetta il formato 2. Non rispetta il formato [Property ok]

Parametro: E-mail Formato: /^\\w+([\\.-]?\\w+)*@\\w+([\\.-]?\\w+)*\\.\\w{2,3}+\$/;	
Lunghezza [LE]	1. < 5 or > 50 [Error] 2. >= 5 or < 40 [Property ok]
Formato [FE]	1. Non rispetta il formato 2. Rispetta il formato [Property ok]

Parametro: Password	
Lunghezza [LP]	1. >= 25 or < 8 [Error] 2. >= 8 or <= 25 [Property ok]

Parametro: Conferma Password	
Match [MP]	1. Match con password password != conferma password [Error] 2. Match con password password == conferma password [Property ok]

Codice	Combinazione	Esito
TC 0.2.1	LN1	Error
TC 0.2.2	LN2, FN1	Error
TC 0.2.3	LN2, FN2, LC1	Error
TC 0.2.4	LN2, FN2, LC2, FC1	Error
TC 0.2.5	LN2, FN2, LC2, FC2, LE1	Error
TC 0.2.6	LN2, FN2, LC2, FC2, LE2, FE1	Error
TC 0.2.7	LN2, FN2, LC2, FC2, LE2, FE2, LP1	Error
TC 0.2.8	LN2, FN2, LC2, FC2, LE2, FE2, LP2, MP1	Error
TC 0.2.9	LN2, FN2, LC2, FC2, LE2, FE2, LP2, MP2	Success

### TC\_0.3: Recupero password

Parametro: E-mail	
Match [ME]	1. Match con email utente = false [Error] 2. Match con email utente = true [Property ok]

Codice	Combinazione	Esito
TC 0.3.1	ME1	Error
TC 0.3.2	ME2	Success

### TC\_1: Gestione Account

#### TC\_1.1: Modifica dati personali

Parametro: Nome Formato: \d/;	
Lunghezza [LN]	1. >= 30 or ==0 [Error] 2. < 30 or > 0 [Property ok]
Formato [FN]	1. Rispetta il formato 2. Non rispetta il formato [Property ok]

Parametro: Cognome Formato: \d/;	
Lunghezza [LC]	1. >= 30 or ==0 [Error] 2. < 30 or > 0 [Property ok]
Formato [FC]	1. Rispetta il formato 2. Non rispetta il formato [Property ok]

Codice	Combinazione	Esito
TC 1.1.1	LN1	Error
TC 1.1.2	LN2, FN1	Error
TC 1.1.3	LN2, FN2, LC1	Error
TC 1.1.4	LN2, FN2, LC2, FC1	Error
TC 1.1.5	LN2, FN2, LC2, FC2	Success



## TC\_1.2: Modifica password

Parametro: Password Attuale	
Match [MPA]	<ol style="list-style-type: none"><li>1. Match con password utente = false [Error]</li><li>2. Match con password utente = true [Property ok]</li></ol>

Parametro: Nuova Password	
Lunghezza [LNP]	<ol style="list-style-type: none"><li>1. <math>\geq 25</math> or <math>&lt; 8</math> [Error]</li><li>2. <math>\geq 8</math> or <math>\leq 25</math> [Property ok]</li></ol>

Parametro: Conferma Password	
Match [MCP]	<ol style="list-style-type: none"><li>1. Match con password nuova password <math>\neq</math> conferma password [Error]</li><li>2. Match con password nuova password <math>=</math> conferma password [Property ok]</li></ol>

Codice	Combinazione	Esito
TC 1.2.1	MPA1	Error
TC 1.2.2	MPA2, LNP1	Error
TC 1.2.3	MPA2, LNP2, MCP1	Error
TC 1.2.4	MPA2, LNP2, MCP2	Success

## TC\_2: Gestione indirizzi

### TC\_2.1: Aggiunge un indirizzo

Parametro: Via Formato: \d/;	
Lunghezza [LV]	<ol style="list-style-type: none"><li>1. <math>&lt; 3</math> or <math>&gt; 30</math> [Error]</li><li>2. <math>\geq 3</math> or <math>\leq 30</math> [Property ok]</li></ol>
Formato [FV]	<ol style="list-style-type: none"><li>1. Rispetta il formato</li><li>2. Non rispetta il formato [Property ok]</li></ol>

Parametro: Città Formato: \d/;	
Lunghezza [LC]	<ol style="list-style-type: none"><li>1. <math>&lt; 3</math> or <math>&gt; 30</math> [Error]</li><li>2. <math>\geq 3</math> or <math>\leq 30</math> [Property ok]</li></ol>
Formato [FC]	<ol style="list-style-type: none"><li>1. Rispetta il formato</li><li>2. Non rispetta il formato [Property ok]</li></ol>

Parametro: CAP Formato: /\d{5}\$/	
Lunghezza [LCAP]	<ol style="list-style-type: none"><li>1. <math>&lt; 5</math> or <math>&gt; 5</math> [Error]</li><li>2. <math>= 5</math> [Property ok]</li></ol>
Formato [FCAP]	<ol style="list-style-type: none"><li>1. Non rispetta il formato [Error]</li><li>2. Rispetta il formato [Property ok]</li></ol>

Parametro: Civico Formato: /^[0-9]+\$	
Lunghezza [LCIV]	1. < 3 or > 3 [Error] 2. == 3 [Property ok]
Formato [FCIV]	1. Non rispetta il formato [Error] 2. Rispetta il formato [Property ok]

Parametro: Telefono Formato: /^[0-9]+\$	
Lunghezza [LT]	1. < 9 or > 10 [Error] 2. >= 9 or <= 10 [Property ok]
Formato [FT]	1. Non rispetta il formato [Error] 2. Rispetta il formato [Property ok]

Codice	Combinazione	Esito
TC 2.1.1	LV1	Error
TC 2.1.2	LV2, FV1	Error
TC 2.1.3	LV2, FV2, LC1	Error
TC 2.1.4	LV2, FC2, LC2, LCAP1	Error
TC 2.1.5	LV2, FC2, LC2, LCAP2, FCAP1	Error
TC 2.1.6	LV2, FC2, LC2, LCAP2, FCAP2, LCIV1	Error
TC 2.1.7	LV2, FC2, LC2, LCAP2, FCAP2, FCIV2, LT1	Error
TC 2.1.8	LV2, FC2, LC2, LCAP2, FCAP2, FCIV2, LT2	Success

### TC\_3: Gestione metodi di pagamento

#### RF\_3.1: Aggiunge una carta

Parametro: Numero Formato: /^[0-9]+\$	
Lunghezza [LNUM]	1. < 16 or > 16[Error] 2. == 16 [Property ok]
Formato [FNUM]	1. Non rispetta il formato [Error] 2. Rispetta il formato [Property ok]

Parametro: Nome Formato: \d/;	
Lunghezza [LN]	1. < 3 or > 30 [Error] 2. >= 3 or <= 30 [Property ok]
Formato [FN]	1. Rispetta il formato 2. Non rispetta il formato [Property ok]

Parametro: Data Formato: \d{4}\-\d{1,2}\-\d{1,2}\$	
Formato [FD]	3. Non rispetta il formato [Error] 4. Rispetta il formato [Property ok]

Parametro: CVC Formato: /^d{3}\$/	
Lunghezza [LC]	3. < 3 or > 3 [Error] 4. == 3 [Property ok]
Formato [FC]	5. Non rispetta il formato [Error] 6. Rispetta il formato [Property ok]

Codice	Combinazione	Esito
TC 3.1.1	LNUM1	Error
TC 3.1.2	LNUM2, FNUM1	Error
TC 3.1.3	LNUM2, FNUM2, LN1	Error
TC 3.1.4	LNUM2, FNUM2, LN2, FN1	Error
TC 3.1.5	LNUM2, FNUM2, LN2, FN2, FD1	Error
TC 3.1.6	LNUM2, FNUM2, LN2, FN2, FD2, LC1	Error
TC 3.1.7	LNUM2, FNUM2, LN2, FN2, FD2, LC2, FC1	Error
TC 3.1.8	LNUM2, FNUM2, LN2, FN2, FD2, LC2, FC2	Success

#### TC\_4: Gestione catalogo

##### TC\_4.1: Aggiunge un prodotto

Parametro: Nome Formato: /^[A-Za-z0-9]+\$/;	
Lunghezza [LN]	1. < 3 or > 30 [Error] 2. >= 3 or <= 30 [Property ok]
Formato [FN]	1. Non rispetta il formato [Error] 2. Rispetta il formato [Property ok]

Parametro: Birrificio Formato: /^[A-Za-z0-9]+\$/;	
Lunghezza [LB]	1. < 3 or > 30 [Error] 2. >= 3 or <= 30 [Property ok]
Formato [FB]	1. Non rispetta il formato [Error] 2. Rispetta il formato [Property ok]

Parametro: Formato Formato: /^[a-zA-Z0-9]+\$/;	
Lunghezza [LF]	1. < 3 or > 25 [Error] 2. >= 3 or <= 25 [Property ok]
Formato [FF]	1. Non rispetta il formato [Error] 2. Rispetta il formato [Property ok]

Parametro: Descrizione Formato: /^[a-zA-Z0-9]+\$/;	
Lunghezza [LD]	1. < 0 or > 1000 [Error] 2. >= 0 or <= 1000 [Property ok]
Formato [FD]	1. Non rispetta il formato [Error] 2. Rispetta il formato [Property ok]

Parametro: Quantità Formato: /^[0-9]+\$	
Lunghezza [LQ]	1. <= 0 [Error] 2. > 0 [Property ok]
Formato [FQ]	1. Non rispetta il formato [Error] 2. Rispetta il formato [Property ok]

Parametro: Prezzo Formato: /(\d{1,2,3,4}\.(\d{1,2}))	
Formato [FP]	1. Non rispetta il formato [Error] 2. Rispetta il formato [Property ok]

Parametro: URL Image Formato: /\w+([\.-]?\w+)*@\w+([\.-]?\w+)*(\.\w{2,3})+\$/	
Lunghezza [LUI]	1. <= 0 or > 250 [Error] 2. > 0 or <= 250 [Property ok]
Formato [FUI]	1. Non rispetta il formato [Error] 2. Rispetta il formato [Property ok]

Codice	Combinazione	Esito
TC 4.1.1	LN1	Error
TC 4.1.2	LN2, FN1	Error
TC 4.1.3	LN2, FN2, LB1	Error
TC 4.1.4	LN2, FN2, LB2, FB1	Error
TC 4.1.5	LN2, FN2, LB2, FB2, LF1	Error
TC 4.1.6	LN2, FN2, LB2, FB2, LF2, FF1	Error
TC 4.1.7	LN2, FN2, LB2, FB2, LF2, FF2, LD2,	Error
TC 4.1.9	LN2, FN2, LB2, FB2, LF2, FF2, LD2, FD1	Error
TC 4.1.10	LN2, FN2, LB2, FB2, LF2, FF2, LD2, FD2, LQ1	Error
TC 4.1.11	LN2, FN2, LB2, FB2, LF2, FF2, LD2, FD2, LQ2, FP1	Error
TC 4.1.12	LN2, FN2, LB2, FB2, LF2, FF2, LD2, FD2, LQ2, FP2, LUI1	Error
TC 4.1.13	LN2, FN2, LB2, FB2, LF2, FF2, LD2, FD2, LQ2, FP2, LUI2, FUI1	Error
TC 4.1.14	LN2, FN2, LB2, FB2, LF2, FF2, LD2, FD2, LQ2, FP2, LUI2	Success

## TC\_5: Gestione utenti

### TC\_5.1: Aggiunge un utente back-office

Parametro: E-mail Formato: /^\\w+([\\.-]?\\w+)*@\\w+(beerhappy\\w+)*\\.\\w{2,3}+\$/	
Lunghezza [LE]	1. > 40 [Error] 2. <= 40 [Property ok]
Formato [FE]	1. Non rispetta il formato [Error] 2. Rispetta il formato [Property ok]

Codice	Combinazione	Esito
TC 5.1.1	LE1	Error
TC 5.1.2	LE2, FE1	Error
TC 5.1.3	LE2, FE2	Success