

Università degli Studi di Salerno

Corso di Ingegneria del Software

Nome Progetto: Gametop
Titolo Documento: Test Plan



Test Plan (TP)

Manager:

Professori
Prof. De Lucia Andrea

Partecipanti:

Nome	Matricola
Vincenzo Tarantino	0512108025
Luigi Sodano	0512107590

Revision History:

Data	Versione	Descrizione	Autore
18/10/2023	1.0	Prima stesura del document problem statement	Team members
30/10/2023	2.0	Stesura del document RAD	Team members
15/11/2023	3.0	Revisione del document RAD	Team members
03/01/2024	3.1	Ultima revisione del document	Team members
20/11/2023	3.2	Stesura document System Design	Team members

23/01/2024	3.3	Revisione document System Design	Team members
03/12/2023	4	Stesura Object design document	Team members
23/01/2024	4.1	Ultima revisione del Object design document	Team members
16/12/2023	5	Stesura Test Plan e Test Plan DAO	Team members
23/01/2024	5.1	Ultima revisione del Test Plan e Test Plan DAO	Team members

Sommario

1.	<u>Introduzione</u>	<u>3</u>
2.	<u>Documenti Correlati.....</u>	<u>3</u>
	2.1 <u>Relazioni con il documento di analisi dei requisiti (RAD)</u>	<u>3</u>
	2.2 <u>Relazioni con il System Design Document (SDD)</u>	<u>4</u>
	2.3 <u>Relazioni con l'Object Design Document (ODD)</u>	<u>4</u>
3.	<u>Panoramica del Sistema</u>	<u>4</u>
4.	<u>Funzionalità da Testare</u>	<u>5</u>
5.	<u>Criteri Pass/Fail</u>	<u>5</u>
6.	<u>Approccio</u>	<u>6</u>
	6.1 <u>Test di unità</u>	<u>6</u>
	6.2 <u>Test di integrazione</u>	<u>6</u>
	6.3 <u>Test di sistema</u>	<u>6</u>
7.	<u>Sospensione e Ripresa</u>	<u>7</u>
	7.1 <u>Criteri di sospensione</u>	<u>7</u>
	7.2 <u>Criteri di ripresa</u>	<u>7</u>
8.	<u>Materiale per il Testing.....</u>	<u>7</u>
9.	<u>Test Cases</u>	<u>8</u>
	9.1 <u>Gestione Account</u>	<u>8</u>
	9.1.1 <u>Log-in.....</u>	<u>8</u>
10.	<u>Registrazione</u>	

1. Introduzione

Questo documento ha come obiettivo la realizzazione e gestione delle attività di testing riguardanti le varie componenti del sistema Gametop.

Lo scopo di questa attività è quello di verificare il corretto funzionamento del sistema sviluppato in diversi casi.

I casi sono stati individuati per mettere alla prova le varie funzionalità del software.

Effettuando tali test saremo in grado di rilevare eventuali errori, bug o incongruenze tra il comportamento desiderato e quello effettivo del sistema.

I risultati di questi test saranno usati per capire dove bisognerà intervenire, e quindi effettuare correzioni degli eventuali errori rilevati o apportare modifiche per il miglioramento dei vari sottosistemi.

2. Documenti Correlati

Il documento Test Plan è correlato a tutti gli altri documenti stilati nel corso della progettazione del sistema, i test del Test Plan saranno utili per verificare se quando descritto nei vari documenti è stato rispettato.

2.1 Relazioni con il documento di analisi dei requisiti (RAD)

La relazione tra Test Plan e RAD (Requirement Analysis Document) è data dal fatto che i test che verranno effettuati sono legati ai requisiti funzionali e non funzionali specificati nel RAD.

In particolare, il RAD contiene lo scopo del sistema, l'ambito del sistema e gli obiettivi, mostrando una panoramica di requisiti funzionali e non funzionali, scenari, casi d'uso, diagrammi del sistema.

2.2 Relazioni con il System Design Document (SDD)

L'SDD (System Design Document) è un documento in cui viene rappresentata l'architettura del sistema sviluppato in tre layer. Il test deve tenere conto di queste suddivisioni, in particolare, l'SDD contiene l'architettura del software corrente e di quello proposto, informazioni sulla gestione dei servizi dei sistemi esterni e dei servizi dei sottosistemi.

2.3 Relazioni con l'Object Design Document (ODD)

L'ODD (Object Design Document) contiene i package e le class interfaces del sistema che vanno prese in considerazione durante le operazioni di testing.

3. Panoramica del Sistema

Come riportato nel System Design Document, la struttura del nostro sistema è suddivisa secondo un'architettura strutturata su tre livelli (Three-Tier): Interface Layer, Application Logic Layer, Data Layer.

Il sistema è inoltre suddiviso in sottosistemi:

- Gestione account
- Gestione carrello
- Gestione acquisti
- Gestione prodotto

4. Funzionalità da Testare

Di seguito saranno elencate le funzionalità introdotte nel sistema che saranno sottoposte a test:

- Gestione account
 - ❖ Log-in
 - ❖ Registrazione
 - ❖ Modifica Profilo
 - ❖ Password Dimenticata
 - ❖ Aggiunta personale
 - ❖ Modifica personale
 - ❖ Rimozione personale
- Gestione carrello
 - ❖ Aggiunta al carrello
 - ❖ Rimozione dal carrello
 - ❖ Procedere all'ordine
- Gestione acquisti
 - ❖ Aggiunta spedizione
 - ❖ Aggiunta pagamento
 - ❖ Modifica ordine
 - ❖ Conferma ordine
- Gestione prodotti
 - ❖ Aggiunta prodotto

- ❖ Modifica prodotto
- ❖ Rimozione prodotto

5. Criteri Pass/Fail

I dati di input saranno raggruppati in insiemi dalle caratteristiche comuni, questo permetterà di effettuare il testing su un unico elemento rappresentativo per ogni insieme.

Pass criteria

Il test è considerato superato se l'output prodotto dal test è uguale a quello atteso.

Fail criteria

Il test si considera non superato se l'output ottenuto per testare la componente non corrisponde all'output atteso cioè l'oracolo, ciò significa che la fase di testing avrà successo se individuerà una failure.

In tal caso questa verrà analizzata e, se legata ad un fault, si procederà alla sua correzione.

Sarà infine iterata la fase di testing per verificare che la modifica non abbia impatto su altre componenti del sistema.

6. Approccio

L'approccio che utilizzeremo per testare il sistema è l'approccio Bottom-up ovvero la strategia in cui i test vengono eseguiti partendo dalle unità più basse del sistema, come singole funzioni o moduli, per poi gradualmente salire verso livelli superiori dell'applicazione

6.1 Test di unità

Per realizzare il testing di ogni componente verrà utilizzata la tecnica “**Black-Box**”testing.

In questo modo andremo ad esaminare le funzionalità dell'applicazione ed il comportamento input/output delle singole componenti senza tener conto della loro struttura interna.

6.2 Test di integrazione

Il test di integrazione ha come obiettivo quello di verificare la corretta interazione tra le varie unità del sistema. Per realizzare il test di unità e di integrazione useremo JUnit e Mockito.

6.3 Test di sistema

Una volta effettuati i test di unità si passa poi ad effettuare il test di sistema.

Lo scopo di questa fase è quello di testare tutte le funzionalità più importanti, in particolare quelle usate maggiormente e con maggior probabilità di fallimento.

Trattandosi di un sistema web-based verrà utilizzato, per effettuare il testing di sistema, il tool Selenium.

Selenium si occupa di simulare l'interazione con il sistema dal punto di vista dell'utente.

7 . Sospensione e Ripresa

a. Criteri di sospensione

La fase di testing del sistema verrà sospesa quando si otterranno i risultati attesi in accordo con i tempi di sviluppo previsti, tenendo sempre conto dei costi dell'attività di testing.

b. Criteri di ripresa

La fase di testing potrà riprendere in seguito a delle modifiche o a delle correzioni apportate al sistema, che generano errori o fallimenti.

I test case verranno quindi sottoposti nuovamente al sistema, assicurandosi così di aver risolto effettivamente il problema.

8. Materiale per il Testing

Il materiale richiesto per il testing sarà:

- Un calcolatore
- IDE Java EE
- MySQL Server
- JUnit
- Selenium

9. Test Cases

Usiamo la tecnica del Category Partition

1.1 Registrazione

Parametro :	Nome
Formato [LN]	1. Lunghezza>0 2. Lunghezza=0

Parametro :	Cognome
Formato [LC]	1. Lunghezza>0 2. Lunghezza=0

Parametro :	Email
Formato [LE]	1. Lunghezza>0 2. Lunghezza=0

Parametro :	Email
Esistenza nel database[EE]	1. Email già esistente nel database 2. Email non esistente nel database

Parametro :	Indirizzo
Formato [LI]	1. Lunghezza>0 2. Lunghezza=0

Parametro :	Password
Formato [LP]	1. Lunghezza>0 2. Lunghezza=0

Parametro :	Email
Formato [FE]	1. Rispetta il formato <code>^[a-zA-Z0-9.!#\$%&'*/+=?^_`{ }~-]+@[a-zA-Z0-9](?:[a-zA-Z0-9]{0,61}[a-zA-Z0-9])?(?:\.[a-zA-Z0-9](?:[a-zA-Z0-9]{0,61}[a-zA-Z0-9])?)*\$</code> 2. Non rispetta il formato

Parametro :	Password
-------------	----------

Formato [FP]	1. Rispetta il formato <code>^(?=.*[a-z])(?=.*[A-Z])(?=.*\d)(?=.*[@\$!%*?&])[A-Za-z\d@\$!%*?&]{8,}\$</code> 2. Non rispetta il formato
--------------	---

TestCase 1.1

Identificativo	Combinazione	Esito
TC 1.1.1	LN.2, LC.2, LE.2, LP.2, LI.2	Negativo(campi vuoti)
TC 1.1.2	LN.1, LC.1, LE.1, EE.2, LP.1, FE.2, FP.1	Negativo(formato email errato)
TC 1.1.3	LN.1, LC.1, LE.1, EE.1, LP.1, FE.1, FP.2	Negativo(formato password errata)
TC 1.1.4	LN.1, LC.1, LE.1, EE.2, LP.1, FE.1, FP.1	Negativo(email già esistente nel db)
TC 1.1.5	LN.1, LC.1, LE.1, LP.1, FE.1, FP.1, EE.2	Positivo

1.2 Login

Parametro :	Email
Formato [LE]	1. Lunghezza>0 2. Lunghezza=0

Parametro :	Password
Formato [LP]	1. Lunghezza>0 2. Lunghezza=0

Parametro :	Email
Esistenza email [EE]	1. Trovata nel DB 2. Non trovata nel DB

Parametro :	Password
Corrispondenza [C]	1. Corrisponde 2. Non corrisponde

TestCase 1.2

Identificativo	Combinazione	Esito
TC 1.2.1	LE.2, LP.2	Negativo(Campi vuoti)
TC 1.2.2	LE.1, LP.1, EE.2,C.1	Negativo(email non esistente nel database)
TC 1.2.3	LE.1, LP.1, EE.1, C.2	Negativo(password non corrisponde)
TC 1.2.4	LE.1, LP.1, EE.1,C1	Positivo(dati inseriti correttamente)

1.3 Modifica password

Parametro :	VecchiaPassword
Formato [LVP]	<ol style="list-style-type: none"> 1. Lunghezza>0 2. Lunghezza=0

Parametro :	NuovaPassword
Formato [LNP]	<ol style="list-style-type: none"> 1. Lunghezza>0 2. Lunghezza=0

Parametro :	ConfermaPassword
Formato [LCP]	<ol style="list-style-type: none"> 1. Lunghezza>0 2. Lunghezza=0

Parametro :	VecchiaPassword
Corrispondenza [CVP]	<ol style="list-style-type: none"> 1. Corrisponde 2. Non corrisponde

Parametro :	NuovaPassword
Formato [FNP]	<ol style="list-style-type: none"> 1. Rispetta il formato <code>^(?=.*[a-z])(?=.*[A-Z])(?=.*\d)(?=.*[@\$!%*?&])[A-Za-z\d@\$!%*?&]{8,}\$</code> 2. Non rispetta il formato

Parametro :	ConfermaPassword
Corrispondenza [CCP]	<ol style="list-style-type: none"> 1. Corrisponde 2. Non corrisponde

TestCase 1.3

Identificativo	Combinazione	Esito
TC 1.3.1	LVP.2, LNP.2, LCP.2	Negativo(non inserisce dati)
TC 1.3.2	LVP.1, LNP.1, LCP.1, CVP.2, FNP.1, CNP.1	Negativo(vecchia password errata)
TC 1.3.3	LVP.1, LNP.1, LCP.1, CVP.1, FNP.2, CNP.1/CNP.2	Negativo(formato nuova password errata)
TC 1.3.4	LVP.1, LNP.1, LCP.1, CVP.1, FNP.1, CNP.2	Negativo(conferma nuova password errata)

TC 1.3.5	LVP.1, LNP.1, LCP.1, CVP.1, FNP.1, CNP.1	Positivo(dati inseriti correttamente)
----------	--	---------------------------------------

1.4 Inserimento personale

Parametro:	Nome
Formato [LN]	1. Lunghezza>0 2. Lunghezza=0

Parametro :	Cognome
Formato [LC]	1. Lunghezza>0 2. Lunghezza=0

Parametro :	Email
Formato [LE]	1.Lunghezza>0 2.Lunghezza=0

Parametro :	Password
Formato [LP]	1. Lunghezza>0 2. Lunghezza=0

Parametro :	Email
Formato [FE]	1. Rispetta il formato <code>^[a-zA-Z0-9.!#\$%&'*/+=?^_`{ }~-]+@[a-zA-Z0-9](?:[a-zA-Z0-9]{0,61}[a-zA-Z0-9])?(?:\.[a-zA-Z0-9](?:[a-zA-Z0-9]{0,61}[a-zA-Z0-9])?)*\$</code> 2. Non rispetta il formato

Parametro :	Email
Esistenza nel database[EE]	1. Email già esistente nel database 2. Email non esistente nel database

Parametro :	Password
Formato [FP]	1. Rispetta il formato <code>^(?=.*[a-z])(?=.*[A-Z])(?=.*\d)(?=.*[@\$!%*?&])[A-Za-z\d@\$!%*?&]{8,}\$</code> 2. Non rispetta il formato

TestCase 1.4

Identificativo	Combinazione	Esito
TC 1.4.1	LN.2, LC.2, LE.2, LP.2	Negativo(Dati non inseriti)
TC 1.4.2	LN.1, LC.1, LE.1, LP.1, FE.2, EE.1, FP.1	Negativo(Formato email errato)

TC 1.4.3	LN.1, LC.1, LE.1, LP.1, FE.1, EE.1, FP.2	Negativo(Formato password errato)
TC 1.4.4	LN.1, LC.1, LE.1, LP.1, FE.1, EE.1, FP.1	Negativo(email già esistente)
TC 1.4.5	LN.1, LC.1, LE.1, LP.1, FE.1, FP.1, EE2	Positivo()

1.5 Modifica personale

Parametro:	Nome
Formato [LN]	1. Lunghezza>0 2. Lunghezza=0

Parametro :	Cognome
Formato [LC]	1. Lunghezza>0 2. Lunghezza=0

Parametro :	Password
Formato [LP]	1. Lunghezza>0 2. Lunghezza=0

Parametro :	Password
Formato [FP]	1. Rispetta il formato <code>^(?=.*[a-z])(?=.*[A-Z])(?=.*\d)(?=.*[@\$!%*?&])[A-Za-z\d@\$!%*?&]{8,}\$</code> 2. Non rispetta il formato

TestCase 1.5

Identificativo	Combinazione	Esito
TC 1.5.1	LN.2, LC.2, LP.2	Negativo(Il gestore ruoli non inserisce dati)
TC 1.5.2	LN.1, LC.1, LP.1, FP.2	Negativo(Formato password errato)
TC 1.5.3	LN.1, LC.1, LP.1, FP.1	Positivo

2.1 Inserimento nuovo prodotto

Parametro:	CodiceProdotto
Formato [LCO]	1. Lunghezza>0 2. Lunghezza=0

Parametro:	Quantità
Formato [LQ]	1. Lunghezza>0 2. Lunghezza=0

Parametro:	Titolo
Formato [LT]	1. Lunghezza>0 2. Lunghezza=0

Parametro:	Descrizione
Formato [LD]	1. Lunghezza>0 2. Lunghezza=0

Parametro:	Prezzo
Formato [LP]	1. Lunghezza>0 2. Lunghezza=0

Parametro:	Copertina
Formato [LF]	1. Lunghezza>0 2. Lunghezza=0

Parametro:	Codice
Formato [FCO]	<ol style="list-style-type: none"> 1. Rispetta il formato <code>^[(a-z)(A-Z)]{2}[0-9]{5}\$</code> (stringa che inizi con due lettere (minuscole o maiuscole) seguite da esattamente cinque cifre.) 2. Non rispetta il formato

Parametro:	Prezzo
Formato [FP]	<ol style="list-style-type: none"> 1. Prezzo>0 2. Prezzo=0

Parametro:	Codice
Formato [FCE]	<ol style="list-style-type: none"> 1. Esistenza nel database 2. Non esistenza nel database

TestCase 2.1

Identificativo	Combinazione	Esito
TC 2.1.1	LCO.2, LCA.2, LQ.2, LT.2, LD.2, LP.2, LF.2	Negativo(Il gestore prodotti non inserisce dati)
TC 2.1.2	LCO.1, LCA.1, LQ.1, LT.1, LD.1, LP.1, LF.1, FCO.2, FP.1, FCE.1	Negativo(Il gestore prodotti inserisce formato codice scorretto)
TC 2.1.3	LCO.1, LCA.1, LQ.1, LT.1, LD.1, LP.1, LF.1, FCO.1, FP.2, FCE.1	Negativo(Il gestore prodotti inserisce il prezzo uguale a 0)
TC 2.1.4	LCO.1, LCA.1, LQ.1, LT.1, LD.1, LP.1, LF.1, FCO.1, FP.1, FCE.1	Negativo(Il Gestore Prodotti inserisce un codice già esistente)
TC 2.1.5	LCO.1, LCA.1, LQ.1, LT.1, LD.1, LP.1, LF.1, FCO.1, FP.1, FCE.2	Positivo

2.2 Modifica prodotto

Parametro:	Quantità
Formato [LQ]	1. Lunghezza>0 2. Lunghezza=0

Parametro:	Titolo
Formato [LT]	1. Lunghezza>0 2. Lunghezza=0

Parametro:	Descrizione
Formato [LD]	1. Lunghezza>0 2. Lunghezza=0

Parametro:	Prezzo
Formato [LP]	1. Lunghezza>0 2. Lunghezza=0

Parametro:	Prezzo
Formato [FP]	1. Prezzo Maggiore di 0 2. Prezzo uguale a 0

TestCase 2.2

Identificativo	Combinazione	Esito
TC 2.2.1	LQ.2, LT.2, LD.2, LP.2	Negativo(il gestore prodotti non inserisce i dati)
TC 2.2.2	LQ.1, LT.1, LD.1, LP.1, FP.2	Negativo(il gestore prodotti mette un prezzo uguale a 0)
TC 2.2.3	LQ.1, LT.1, LD.1, LP.1, LF.1, FP.1	Positivo

3.1 Inserimento dati spedizione

Parametro:	Nome
Nome [LN]	1.Lunghezza>0 2. Lunghezza=0

Parametro:	Cognome
Cognome [LC]	1. Lunghezza>0 2. Lunghezza=0

Parametro:	Indirizzo
Indirizzo [LI]	1. Lunghezza>0 2. Lunghezza=0

Parametro:	CAP
CAP [LC]	1. Lunghezza>0 2. Lunghezza=0

Parametro:	Comune
Comune [LCO]	1. Lunghezza>0 2. Lunghezza=0

Parametro:	Provincia
Provincia [LP]	1. Lunghezza>0 2. Lunghezza=0

Parametro:	CAP
Formato [FCAP]	1. Rispetta il formato <code>^[0-9]{0,5}\$</code> (stringhe che contengono da zero a cinque cifre consecutive) 2. Non rispetta il formato

TestCase 3.1

Identificativo	Combinazione	Esito
TC 3.1.1	LN.2, LC.2, LI.2, LC.2, LCO.2, LP.2	Negativo(Il cliente non ha inserito tutti i dati)
TC 3.1.2	LN.1, LC.1, LI.1, LC.1, LCO.1, LP.1, FCAP.2	Negativo(il cliente non rispetta il formato del cap)
TC 3.1.3	LN.1, LC.1, LI.1, LC.1, LCO.1, LP.1, FCAP.1	Positivo

3.2 Inserimento dati pagamento

Parametro:	Numero Carta
Numero Carta [LNC]	1. Lunghezza>0 2. Lunghezza=0

Parametro:	Nome e Cognome
Cognome [LNEC]	1. Lunghezza>0 2. Lunghezza=0

Parametro:	Data scadenza
Data scadenza [LDS]	1. Lunghezza>0 2. Lunghezza=0

Parametro:	CVC
LCVC [LC]	1. Lunghezza>0 2. Lunghezza=0

Parametro:	Numero Carta
Formato [FNC]	1. Rispetta il formato $^{[0-9]\{19,19\}}\$$ (esattamente 19 cifre) 2. Non rispetta il formato

Parametro:	Data scadenza
Formato [FDS]	1. Rispetta il formato (due cifre mese – 4 cifre anno) 2. Non rispetta il formato

Parametro:	Numero sicuro (CVC)
Formato [FNS]	1. Rispetta il formato $^{[0-9]\{3\}}\$$ (esattamente 3 cifre) 2. Non rispetta il formato

Identificativo	Combinazione	Esito
TC 3.2.1	LNC.2, LNEC.2, LDS.2, LCVC.2	Negativo(III cliente non ha inserito i dati della carta)
TC 3.2.2	LNC.1, LNEC.1, LDS.1, LCVC.1, FNC.2, FDS.1, FNS.1	Negativo(II cliente inserisce il numero carta scorretto)

TC 3.2.3	LNC.1, LNEC.1, LDS.1, LCVC.1, FNC.2, FDS.2, FNS.1	Negativo(non rispetta il formato della data)
TC 3.2.4	LNC.1, LNEC.1, LDS.1, LCVC.1, FNC.1, FDS.1, FNS.2	Negativo(Il cliente non rispetta il formato del cvc)
TC 3.2.5	LNC.1, LNEC.1, LDS.1, LCVC.1, FNC.1, FDS.1, FNS.1	Positivo

4.1 Modifica ordine

Parametro:	Nome
Formato [LN]	1. Lunghezza>0 2. Lunghezza=0

Parametro:	Cognome
Formato [LC]	1. Lunghezza>0 2. Lunghezza=0

Parametro:	Indirizzo
Formato [LI]	1.Lunghezza>0 2.Lunghezza=0

Parametro:	CAP
Formato [LCAP]	1.Lunghezza>0 2.Lunghezza=0

Parametro:	Comune
Formato [LCO]	1. Lunghezza>0 2. Lunghezza=0

Parametro:	Provincia
Formato [LDS]	1.Lunghezza>0 2.Lunghezza=0

Parametro:	Prezzo
Formaro [LP]	1. Lunghezza>0 2. Lunghezza=0

Parametro:	CAP
Formato [FCAP]	3. Rispetta il formato <code>^[0-9]{0,3}\$</code> 4. Non rispetta il formato

Parametro:	Prezzo
Formato [FP]	1. Prezzo >0 2. Prezzo=0

TestCase 4.1

Identificativo	Combinazione	Esito
TC 4.1.1	LN.2, LC.2, LI.2, LCAP.2, LCO.2, LP.2, LF.2	Negativo(Il gestore ordini non inserisce dati)
TC 4.1.2	LN.1, LC.1, LI.1, LCAP.1, LCO.1, LP.1, LF.1, FCAP.2, FP.1	Negativo(Il gestore ordini inserisce il cap con un formato non valido)
TC 4.1.3	LN.1, LC.1, LI.1, LCAP.1, LCO.1, LP.1, LF.1, FCAP.1, FP.2	Negativo(Il gestore ordini inserisce il prezzo=0)
TC 4.1.4	LN.1, LC.1, LI.1, LCAP.1, LCO.1, LP.1, LF.1, FCAP.1, FP.1	Positivo