Immagine che contiene testo, logo, emblema, simbolo

Descrizione generata automaticamente

Corso di Laurea in Informatica

Progetto Ingegneria del Software

TEST PLAN

VERSIONE 1.0

Immagine che contiene cerchio, Elementi grafici, simbolo, Policromia

Descrizione generata automaticamente

**GameChanger**

# Partecipanti

Isidoro D’Arco – 0512113779

Ruben Petina - 0512114733

Sommario

[Partecipanti 1](#_Toc184222124)

[1. Introduzione 3](#_Toc184222125)

[2. Relazione con altri documenti 3](#_Toc184222126)

[3. Funzionalità 4](#_Toc184222127)

[4. Criteri di successo/insuccesso 4](#_Toc184222128)

[5. Approccio 4](#_Toc184222129)

[5.1 Testing di unità 4](#_Toc184222130)

[5.2 Testing di integrazione 5](#_Toc184222131)

[5.3 Testing di integrazione 5](#_Toc184222132)

[6. Sospensione e ripresa 5](#_Toc184222133)

[6.1 Criteri di sospensione 5](#_Toc184222134)

[6.2 Criteri di ripresa 5](#_Toc184222135)

[7. Materiali per il test 5](#_Toc184222136)

[8. Casi di test 6](#_Toc184222137)

[8.1 Gestione account 6](#_Toc184222138)

[8.2 Gestione account 9](#_Toc184222139)

## Introduzione

Lo scopo del documento di test plan è quello di pianificare e gestire le attività di test per il sistema GameChanger.  
Verranno indicate le componenti e le funzionalità da testare, le strategie di testing adottate e gli strumenti utilizzati.

Lo scopo del testing è quello di rilevare errori in maniera pianificata all’interno del codice realizzato, affinché essi non si presentino durante l’utilizzo da parte dell’utente. I risultati prodotti dai test saranno utilizzati per comprendere dove intervenire per correggere gli errori o apportare modifiche per migliorare il sistema.

## Relazione con altri documenti

Il test plan ha una forte correlazione con i documenti prodotti finora. Per verificare il corretto funzionamento del sistema verranno utilizzati dei test case individuati durante il processo di sviluppo del sistema, e basati sulle funzionalità individuate nel documento di raccolta e analisi dei requisiti (RAD).

**RAD**: La relazione con il Requirement Analysis Document riguarda in particolare i requisiti funzionali e non funzionali del sistema, i test verranno progettati tenendo in considerazione le specifiche espresse in tale documento.

**SDD**: il System Design Document rappresenta l’architettura del sistema, e nel progettare i test si terrà conto dell’architettura del software e della suddivisione in sottosistemi.

**ODD:** i test si baseranno sulle interfacce definite nell’Object Design Document.

## Funzionalità

Abbiamo deciso di non testare i metodi get e set e i metodi con priorità bassa.

Di seguito vengono elencate le funzionalità nel sistema che saranno sottoposte a testing:

**Gestione account**

Registrazione

Login

**Operazioni sui prodotti**

Aggiunta di articoli al catalogo

Acquistare articoli

## Criteri di successo/insuccesso

I dati di input dei test case vengono raggruppati in sottoinsiemi, dalle caratteristiche comuni in modo da avere dei dati che siano rappresentativi per una intera porzione del dominio.

Il testing ha successo se l’output osservato differisce dall’output atteso: 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 impattato su altri componenti del sistema. La failure quindi è qualsiasi variazione del comportamento del sistema rispetto al comportamento atteso specificato in analisi dei requisiti.

## Approccio

L’approccio alla fase di testing si compone in tre fasi, la prima avrà lo scopo di testare le componenti del sistema singolarmente, poi verranno testate le componenti raggruppate integrandole insieme in gruppi, infine si testerà il funzionamento dell’intero sistema.

### 5.1 Testing di unità

Per i test di unità verrà utilizzata una strategia **black box**. Così facendo andremo ad esaminare le funzionalità dell’applicazione ed il comportamento input/output delle singole componenti senza tener conto della loro struttura interna. Essendo per la quasi totalità dei domini impossibile generare tutti i possibili input, verrà partizionato l’input utilizzando la tecnica del Category Partition. I risultati del testing verranno analizzati e usati per correggere gli errori che causano il fallimento del sistema.

### Testing di integrazione

Dopo aver sottoposto ogni componente al testing di unità, ed aver corretto gli eventuali errori scaturiti dal test, essi verranno integrati in sottosistemi più grandi per sottoporli ad un test di integrazione. Il testing verrà effettuato seguendo la strategia “Bottom-up” che prevede l’integrazione dal basso verso l’alto.

### Testing di integrazione

Prima di essere pronto all’uso, il sistema affronterà l’ultima fase di testing, quello di sistema, per dimostrare che siano soddisfatti tutti i requisiti richiesti. Lo scopo di questa fase è testare le funzionalità più importanti, usate maggiormente e con maggior probabilità di fallimento. Per determinare se il sistema rispecchia tutti i requisiti funzionali e globali sarà effettuato il test tramite **Eclipse IDE.**

## Sospensione e ripresa

### 6.1 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.

Questo processo verrà quindi portato avanti quanto più possibile nel tempo senza però rischiare di ritardare la consegna finale del progetto.

### Criteri di ripresa

La fase di testing potrà riprendere in seguito a modifiche o correzioni che generano errori o fallimenti, i test case verranno, quindi, sottoposti nuovamente al sistema assicurandosi così di aver risolto effettivamente il problema.

## Materiali per il test

Gli strumenti utilizzati sono:

* Web server Apache Tomcat
* MySQL Server
* Eclipse IDE

## Casi di test

### 8.1 Gestione account

**TC\_REG – Registrazione**

|  |  |
| --- | --- |
| Parametro: e-mail | |
| **CATEGORIE** | **SCELTE** |
| Formato e-mail [**fe**] | 1. Non rispetta il formato [property **formatoEmailNonValido**] 2. Rispetta il formato [property **formatoEmailValido**] |
| Esistenza nel database [**ee**] | 1. Esiste nel database [property **emailEsiste**] 2. Non esiste nel database [*if formatoEmailValido*] [property **emailNonEsiste**] |

|  |  |
| --- | --- |
| Parametro: password | |
| **CATEGORIE** | **SCELTE** |
| Lunghezza password [**lp**] | 1. Lunghezza < 8 [property **passwordCorta**] 2. Lunghezza >= 8 [property **passwordLunga**] |
| Formato password [**fp**] | 1. Non rispetta il formato [*if passwordLunga*] [property **formatoPassNonValido**] 2. Rispetta il formato [*if passwordLunga*] [property **formatoPassValido**] |

|  |  |
| --- | --- |
| Parametro: password2 | |
| **CATEGORIE** | **SCELTE** |
| Coincide con password [**ep**] | 1. Non coincide con *password* [*if formatoPassValido*] [property **nonCoincide**] 2. Coincide [*if formatoPassValido*] [property **coincide**] |

|  |  |
| --- | --- |
| Parametro: username | |
| **CATEGORIE** | **SCELTE** |
| Lunghezza username [**lu**] | 1. Lunghezza > 22 [property **usernameNonValido**] 2. Lunghezza < 22 [property **usernameValido**] |
| Formato username[**fu**] | 1. Non rispetta il formato [*if usernameValido*] [property **formatoUsernameNonValido**]  2. Rispetta il formato [*if usernameValido*] [property **formatoUsernameValido**] |

|  |  |
| --- | --- |
| Parametro: nome | |
| **CATEGORIE** | **SCELTE** |
| Lunghezza nome [**ln**] | 1. Lunghezza > 30 [property **nomeNonValido**] 2. Lunghezza < 30 [property **nomeValido**] |
| Formato nome[**fn**] | 1. Non rispetta il formato [*if nomeValido*] [property **formatoNomeNonValido**]  2. Rispetta il formato [*if nomeValido*] [property **formatoNomeValido**] |

|  |  |
| --- | --- |
| Parametro: cognome | |
| **CATEGORIE** | **SCELTE** |
| Lunghezza cognome [**lc**] | 1. Lunghezza > 30 [property **cognomeNonValido**] 2. Lunghezza < 30 [property **cognomeValido**] |
| Formato cognome[**fc**] | 1. Non rispetta il formato [*if cognomeValido*] [property **formatoCognomeNonValido**]  2. Rispetta il formato [*if cognomeValido*] [property **formatoCognomeValido**] |

|  |  |
| --- | --- |
| Parametro: data | |
| **CATEGORIE** | **SCELTE** |
| Formato data[**fd**] | 1. Non rispetta il formato [*if dataValido*] [property **formatoDataNonValido**] 2. Rispetta il formato [*if dataValido*] [property **formatoDataValido**] |

|  |  |
| --- | --- |
| Parametro: checkbox | |
| **CATEGORIE** | **SCELTE** |
| Conferma checkbox [**cc**] | 1. Checkbox non spuntata [property **checkboxNonValida**] 2. Checkbox spuntata [property **checkboxValida**] |

|  |  |  |
| --- | --- | --- |
| Codice | Combinazione | Esito |
| TC\_REG\_1.1 | fe1 | errore |
| TC\_REG\_1.2 | fe2, ee1 | errore |
| TC\_REG\_1.3 | fe2, ee2, lp1 | errore |
| TC\_REG\_1.4 | fe2, ee2, lp2, fp1 | errore |
| TC\_REG\_1.5 | fe2, ee2, lp2, fp2, ep1 | errore |
| TC\_REG\_1.6 | fe2, ee2, lp2, fp2, ep2, lu1 | errore |
| TC\_REG\_1.7 | fe2, ee2, lp2, fp2, ep2, lu2,fu1 | errore |
| TC\_REG\_1.8 | fe2, ee2, lp2, fp2, ep2, lu2,fu2, ln1 | errore |
| TC\_REG\_1.9 | fe2, ee2, lp2, fp2, ep2, lu2,fu2, ln2, fn1 | errore |
| TC\_REG\_1.10 | fe2, ee2, lp2, fp2, ep2, lu2,fu2, ln2, fn2, lc1 | errore |
| TC\_REG\_1.11 | fe2, ee2, lp2, fp2, ep2, lu2,fu2, ln2, fn2, lc2, fc1 | errore |
| TC\_REG\_1.12 | fe2, ee2, lp2, fp2, ep2, lu2,fu2, ln2, fn2, lc2, fc2, fd1 | errore |
| TC\_REG\_1.13 | fe2, ee2, lp2, fp2, ep2, lu2,fu2, ln2, fn2, lc2, fc2, fd2, cc1 | errore |
| TC\_REG\_1.14 | fe2, ee2, lp2, fp2, ep2, lu2,fu2, ln2, fn2, lc2, fc2, fd2, cc2 | successo |

**TC\_LOG – Login**

|  |  |
| --- | --- |
| Parametro: email | |
| **CATEGORIE** | **SCELTE** |
| Presenza nel database [**ee**] | 1. L'email non esiste nel database [property **emailNonEsiste**] 2. L'email esiste nel database [property **emailEsiste**] |

|  |  |
| --- | --- |
| Parametro: password | |
| **CATEGORIE** | **SCELTE** |
| Corrispondenza password [**cp**] | 1. la password non coincide [property **passwordNonCorretta**] 2. la password coincide [property **passwordCorretta**] |

|  |  |  |
| --- | --- | --- |
| Codice | Combinazione | Esito |
| TC\_LOG\_2.1 | ee1 | errore |
| TC\_LOG\_2.2 | ee2, cp1 | errore |
| TC\_LOG\_2.3 | ee2, cp2 | successo |

### 8.2 Gestione account

**TC\_AGG – Aggiunta nuovo prodotto**

|  |  |
| --- | --- |
| Parametro: nome | |
| **CATEGORIE** | **SCELTE** |
| Formato nome [**fn**] | 1. Non rispetta il formato del nome [property **nomeNonValido**] 2. Rispetta il formato del nome [property **nomeValido**] |

|  |  |
| --- | --- |
| Parametro: prezzo | |
| **CATEGORIE** | **SCELTE** |
| Formato prezzo [**fp**] | 1. Non rispetta il formato del prezzo [property **prezzoNonValido**] 2. Rispetta il formato del prezzo [property **prezzoValido**] |

|  |  |
| --- | --- |
| Parametro: marca | |
| **CATEGORIE** | **SCELTE** |
| Formato marca [**fm**] | 1. Non rispetta il formato della marca [property **marcaNonValida**] 2. Rispetta il formato della marca [property **marcaValida**] |

|  |  |
| --- | --- |
| Parametro: data | |
| **CATEGORIE** | **SCELTE** |
| Formato data[**fd**] | 1. Non rispetta il formato [property **formatoDataNonValido**] 2. Rispetta il formato [property **formatoDataValido**] |

|  |  |
| --- | --- |
| Parametro: descrizione | |
| **CATEGORIE** | **SCELTE** |
| Formato della descrizione[**fdd**] | 1. Non rispetta il formato [property **formatoDellaDescrizioneNonValido**] 2. Rispetta il formato [property **formatoDellaDescrizioneValido**] |

|  |  |
| --- | --- |
| Parametro: codice | |
| **CATEGORIE** | **SCELTE** |
| Formato codice[**fc**] | 1. Non rispetta il formato [property **formatoCodiceNonValido**]  2. Rispetta il formato [property **formatoCodiceValido**] |
| Esistenza nel database [**ee**] | 1. Esiste nel database [property **codiceEsiste**]  2. Non esiste nel database [*if formatoCodiceValido*] [property codice**NonEsiste**] |

|  |  |  |
| --- | --- | --- |
| Codice | Combinazione | Esito |
| TCC\_AGG\_1.1 | fn1 | Errore |
| TCC\_AGG\_1.2 | Fn2, fp1 | Errore |
| TCC\_AGG\_1.3 | Fn2, fp2, fm1 | Errore |
| TCC\_AGG\_1.4 | Fn2, fp2, fm2, fd1 | Errore |
| TCC\_AGG\_1.5 | Fn2, fp2, fm2, fd2, fdd1 | Errore |
| TCC\_AGG\_1.6 | Fn2, fp2, fm2, fd2, fdd2, fc1 | Errore |
| TCC\_AGG\_1.7 | Fn2, fp2, fm2, fd2, fdd2, fc2, ee1 | errore |
| TCC\_AGG\_1.8 | Fn2, fp2, fm2, fd2, fdd2, fc2, ee2 | successo |

**TC\_CHECK – Checkout ordine prodotto**

|  |  |
| --- | --- |
| Parametro: indirizzo | |
| **CATEGORIE** | **SCELTE** |
| Formato indirizzo [**fi**] | 1. Non rispetta il formato dell’indirizzo [property **indirizzoNonValido**] 2. Rispetta il formato dell’indirizzo [property **indirizzoValido**] |

|  |  |
| --- | --- |
| Parametro: città | |
| **CATEGORIE** | **SCELTE** |
| Formato città [**fc**] | 1. Non rispetta il formato della città [property **cittaNonValida**] 2. Rispetta il formato della città [property **cittaValida**] |

|  |  |
| --- | --- |
| Parametro: codice postale | |
| **CATEGORIE** | **SCELTE** |
| Formato codice postale [**fcp**] | 1. Non rispetta il formato del codice postale [property **codicePostaleNonValido**] 2. Rispetta il formato del codice postale [property **codicePostaleValido**] |

|  |  |
| --- | --- |
| Parametro: email | |
| **CATEGORIE** | **SCELTE** |
| Presenza nel database [**ee**] | 1. L'email non esiste nel database [property **emailNonEsiste**] 2. L'email esiste nel database [property **emailEsiste**] |

|  |  |
| --- | --- |
| Parametro: emailPaypal | |
| **CATEGORIE** | **SCELTE** |
| Formato email Paypal [**fep**] | 1. Non rispetta il formato [property **formatoEmailPaypalNonValido**]  2. Rispetta il formato [property **formatoEmailPaypalValido**] |
| Esistenza nel database B [**eeb**] | 1. Esiste nel database [property **emailPaypalEsiste**]  2. Non esiste nel database [*if formatoEmailPaypalValido*] [property **emailPaypalNonEsiste**] |

|  |  |
| --- | --- |
| Parametro: nome | |
| **CATEGORIE** | **SCELTE** |
| Lunghezza nome [**ln**] | 1. Lunghezza > 30 [property **nomeNonValido**] 2. Lunghezza < 30 [property **nomeValido**] |
| Formato nome[**fn**] | 1. Non rispetta il formato [*if nomeValido*] [property **formatoNomeNonValido**]  2. Rispetta il formato [*if nomeValido*] [property **formatoNomeValido**] |

|  |  |  |
| --- | --- | --- |
| Codice | Combinazione | Esito |
| TC\_CHECK\_1.1 | Fi1 | Errore |
| TC\_CHECK\_1.2 | Fi2, fc1 | Errore |
| TC\_CHECK\_1.3 | Fi2, fc2, fcp1 | Errore |
| TC\_CHECK\_1.4 | Fi2, fc2, fcp2, ee1 | Errore |
| TC\_CHECK\_1.5 | Fi2, fc2, fcp2, ee2, fep1 | Errore |
| TC\_CHECK\_1.6 | Fi2, fc2, fcp2, ee2, fep2, eeb1 | Errore |
| TC\_CHECK\_1.7 | Fi2, fc2, fcp2, ee2, fep2, eeb2, ln1 | Errore |
| TC\_CHECK\_1.8 | Fi2, fc2, fcp2, ee2, fep2, eeb2, ln2, fn1 | Errore |
| TC\_CHECK\_1.9 | Fi2, fc2, fcp2, ee2, fep2, eeb2, ln2, fn2 | Successo |