|  |
| --- |
| UNISA | Home  **Università degli Studi di Salerno Corso di Ingegneria del Software** |

**Mockbuster  
Test Plan Document**

**Versione 1.0**

****

Data: 14/12/2024

**Coordinatore del progetto:**

|  |  |
| --- | --- |
| Nome | Matricola |
|  |  |
|  |  |

**Partecipanti:**

|  |  |
| --- | --- |
| Nome | Matricola |
| Roberto Ambrosino | 0512117886 |

|  |  |
| --- | --- |
| **Scritto da:** | Roberto Ambrosino |

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| Data | Versione | Descrizione | Autore |
| 13/12/2024 | 0.1 | Prima stesura del documento | Roberto Ambrosino |
| 14/12/2024 | 1.0 | Aggiunti Introduzione, Riferimenti ad altri documenti, Panoramica del sistema  Funzionalità da testare/non testare, Criteri di superamento/fallimento, Approccio, Sospensione e ripresa, Materiale per il testing, Casi di test, Pianificazione del testing | Roberto Ambrosino |

|  |  |
| --- | --- |
| **1 - Introduzione** | 4 |
| **2 – Riferimenti ad altri documenti** | 4 |
| **3 - Panoramica del sistema** | 4 |
| **4 - Funzionalità da testare/non testare** | 5 |
| **5 - Criteri di superamento/fallimento** | 5 |
| **6 - Approccio** | 5 |
| **7 - Sospensione e ripresa** | 6 |
| **8 - Materiale per il testing** | 6 |
| **9 - Casi di test** | 6 |
| **10 - Pianificazione del testing** | 12 |

**1 - Introduzione**

Il presente piano di test descrive gli obiettivi e l'estensione delle attività di testing per il sistema Mockbuster, una piattaforma digitale per il noleggio e l'acquisto di film online. L'obiettivo è fornire un quadro di riferimento per pianificare ed eseguire i test, garantendo che il sistema soddisfi i requisiti definiti

**2 – Riferimenti ad altri documenti**

Questo piano di test fa riferimento ai seguenti documenti:

* Problem Statement (PS\_Mockbuster): Descrive il dominio del problema e i requisiti iniziali del sistema
* Requirements Analysis Document (RAD\_Mockbuster): Definisce i requisiti funzionali (RF) e non funzionali (RNF), oltre ai casi d'uso principali
* System Design Document (SDD\_Mockbuster): Fornisce una descrizione dettagliata dell'architettura del sistema e dei suoi sottosistemi

Le specifiche dei test si basano sui requisiti definiti nel RAD e sul design descritto nell'SDD, con un sistema di nomenclatura che collega direttamente i requisiti alle relative attività di test

**3 - Panoramica del sistema**

Mockbuster è progettato per:

* Fornire un catalogo di film da noleggiare o acquistare
* Registrare profili utente
* Visualizzare ordini effettuati
* Consentire ai gestori di catalogo di aggiungere, modificare e rimuovere film
* Garantire la sicurezza e l'affidabilità del sistema

L'architettura si basa su un modello Client-Server con un database relazionale (MySQL) per la gestione dei dati

**4 - Funzionalità da testare**

Funzionalità da testare:

* Registrazione e autenticazione degli utenti (RF2, RF3)
* Visualizzazione e ricerca nel catalogo (RF1, RF5)
* Gestione del carrello e finalizzazione degli ordini (RF6, RF7, RF9)
* Gestione del catalogo da parte dei gestori (RF14, RF15)

**5 - Criteri di superamento/fallimento**

Criteri generali

* Superamento: Un test è considerato superato se il sistema si comporta come previsto rispetto ai requisiti specificati
* Fallimento: Un test è considerato fallito se si verifica uno o più dei seguenti casi:
  + Mancato rispetto dei requisiti funzionali
  + Errori non gestiti o crash del sistema
  + Prestazioni inferiori agli standard definiti nei requisiti non funzionali

**6 - Approccio**

L'approccio ai test comprende:

* Test unitari: Verifica dei singoli componenti software (ad esempio, servlet o moduli JSP) eseguiti mediante l’uso di jUnit e Mockito
* Test di integrazione: Validazione delle interazioni tra i componenti, con l'ausilio di diagrammi UML per identificare le dipendenze
* Test di sistema: Valutazione del sistema nella sua interezza rispetto ai requisiti
* Test di regressione: Garantire che modifiche o aggiornamenti non introducano nuovi errori

**7 - Sospensione e ripresa**

Criteri di sospensione

* Malfunzionamenti critici che impediscono la prosecuzione dei test
* Mancanza di risorse hardware o software necessarie

Criteri di ripresa

* Risoluzione del malfunzionamento critico
* Ripristino delle risorse necessarie

**8 - Materiale per il testing**

Hardware richiesto:

* WebServer di test
* Server di test con installazione DBMS MySQL
* Client di test con browser moderni (Chrome, Firefox, Safari)

Software richiesto:

* jUnit e Mockito
* Ambiente di sviluppo per il server applicativo (Glassfish)
* Browser web

**9 - Casi di test**

**Accesso degli Utenti**

**PARAMETRI**

* username: indirizzo email dell'utente
* password: stringa per autenticazione

**OGGETTI DELL'AMBIENTE**

* Database: contiene i dati degli utenti già registrati

|  |  |
| --- | --- |
| **Categorie** | **Scelte** |
| Categoria 1: Email Presente | * EP1: Email presente nel database * EP2: Email assente nel database |
| Categoria 2: Password | * PW1: Password corretta * PW2: Password errata |

|  |  |
| --- | --- |
| **TEST FRAME** | |
| **Combinazione scelte** | **Oracolo** |
| EP1, PW1 | Accesso effettuato con successo |
| EP1, PW2 | Messaggio di errore: "Email e/o Password errata" |
| EP2, PW1 | Messaggio di errore: " Email e/o Password errata" |

**TEST CASE**

* TC1: email: "pippo@example.com", password: "Pluto123" – in DB presente la coppia (email = "pippo@example.com", password = "Pluto123") – ORACOLO: Accesso effettuato con successo
* TC2: email: "pippo@example.com", password: "Topolino456" – in DB presente l'utente con email "pippo@example.com" ma password diversa – ORACOLO: Messaggio di errore " Email e/o Password errata"
* TC3: email: "nuovo@example.com", password: "Pluto123" – in DB assente l'utente con email "nuovo@example.com" – ORACOLO: Messaggio di errore " Email e/o Password errata"

**Registrazione degli Utenti**

**PARAMETRI**

* email: indirizzo email dell'utente
* password: stringa per autenticazione
* nome: nome dell'utente
* cognome: cognome dell'utente
* Indirizzo fatturazione: stringa che rappresenta l'indirizzo di fatturazione dell'utente

**OGGETTI DELL'AMBIENTE**

* Database: contiene i dati degli utenti già registrati

|  |  |
| --- | --- |
| **Categorie** | **Scelte** |
| Categoria 1: Formato Email | * EM1: Email corretta * EM2: Email non corretta |
| Categoria 2: Password | * PW1: Password valida (lunghezza ≥ 8) * PW2: Password NON valida |
| Categoria 3: Nome e Cognome | * NC1: Nome e cognome validi (non vuoti) * NC2: Nome e/o cognome non validi (vuoti) |
| Categoria 4: Indirizzo di Fatturazione | * IF1: Indirizzo valido (non vuoto) * IF2: Indirizzo non valido (vuoto) |

|  |  |
| --- | --- |
| **TEST FRAME** | |
| **Combinazione scelte** | **Oracolo** |
| EM1, PW1, NC1, IF1 | Messaggio di errore: "Email già registrata" |
| EM1, PW1, NC1, IF1 | Registrazione effettuata con successo |
| EM2, PW1, NC1, IF1 | Messaggio di errore: "Email non corretta" |
| EM1, PW2, NC1, IF1 | Messaggio di errore: "Password non valida" |
| EM1, PW1, NC2, IF1 | Messaggio di errore: "Nome non valido" o “Cognome non valido” |
| EM1, PW1, NC1, IF2 | Messaggio di errore: " Indirizzo non valido" |

**TEST CASE**

* TC4: email: "pippo@example.com", password: "Pluto123", nome: "Pippo", cognome: "Rossi", indirizzo fatturazione: " Via Roma 12 40100 Bologna" – in DB presente l'utente con email "pippo@example.com" – ORACOLO: Messaggio di errore "Email già registrata"
* TC5: email: "nuovo@example.com", password: "Pluto123", nome: "Mario", cognome: "Bianchi", indirizzo fatturazione: "Via Roma 12 40100 Bologna" – in DB assente l'utente con email "nuovo@example.com" – ORACOLO: Registrazione effettuata con successo
* TC6: email: "errata", password: "Pluto123", nome: "Luigi", cognome: "Verdi", indirizzo fatturazione: " Via Roma 12 40100 Bologna" – ORACOLO: Messaggio di errore "Email non valida"
* TC7: email: "nuovo@example.com", password: "aa", nome: "Giulia", cognome: "Ferrari", indirizzo fatturazione: " Via Roma 12 40100 Bologna" – ORACOLO: Messaggio di errore "Password non valida"
* TC8: email: "nuovo@example.com", password: "Pluto123", nome: "", cognome: "Verdi", indirizzo fatturazione: "Via Roma 12 40100 Bologna" – ORACOLO: Messaggio di errore "Nome non valido"
* TC9: email: "nuovo@example.com", password: "Pluto123", nome: "Anna", cognome: "Bianchi", indirizzo fatturazione: "" – ORACOLO: Messaggio di errore "Indirizzo non valido"

**Visualizzazione e Ricerca nel Catalogo**

**PARAMETRI**

* query: stringa di ricerca per titolo

**OGGETTI DELL'AMBIENTE**

* Database: contiene i dati del catalogo

|  |  |
| --- | --- |
| **Categorie** | **Scelte** |
| Categoria 1: Query | * Q1: Query valida (titolo esistente) * Q2: Query non valida (titolo inesistente) * Q3: Query vuota |

|  |  |
| --- | --- |
| **TEST FRAME** | |
| **Combinazione scelte** | **Oracolo** |
| Q1 | Visualizzazione dei film corrispondenti |
| Q2 | Messaggio di "Nessun risultato trovato" |
| Q3 | Visualizzazione dell’intero catalogo |

**TEST CASE**

* TC10: query: "Inception" – presente nel catalogo – ORACOLO: Visualizzazione del film "Inception"
* TC11: query: "FilmNonEsistente" – assente nel catalogo – ORACOLO: Messaggio "Nessun risultato trovato"
* TC12: query: "" (vuota) – ORACOLO: Visualizzazione dell'intero catalogo

**Gestione del Carrello e Finalizzazione degli Ordini**

**PARAMETRI**

* carrello: memorizza i film selezionati per l’acquisto o il noleggio
* film: film da aggiungere o rimuovere

**OGGETTI DELL'AMBIENTE**

* Carrello: memorizza i film selezionati per l’acquisto o il noleggio

|  |  |
| --- | --- |
| **Categorie** | **Scelte** |
| Categoria 1: Stato del Carrello | * C1: Carrello vuoto * C2: Carrello con articoli |
| Categoria 2: Azione | * A1: Aggiungere un articolo * A2: Rimuovere un articolo * A3: Procedere con l'ordine |

|  |  |
| --- | --- |
| **TEST FRAME** | |
| **Combinazione scelte** | **Oracolo** |
| C1, A1 | Film aggiunto con successo |
| C2, A2 | Film rimosso con successo |
| C1, A3 | Messaggio di errore: "Carrello vuoto" |
| C2, A3 | Ordine completato con successo |

**TEST CASE**

* TC13: carrello vuoto, azione: aggiungi film "Inception" – ORACOLO: Film aggiunto con successo
* TC14: carrello con "Inception", azione: rimuovi film "Inception" – ORACOLO: Film rimosso con successo
* TC15: carrello vuoto, azione: procedi all'ordine – ORACOLO: Messaggio di errore "Carrello vuoto"
* TC16: carrello con "Inception", azione: procedi all'ordine – ORACOLO: Ordine completato con successo

**Gestione del Catalogo da Parte dei Gestori**

**PARAMETRI**

* film: dettagli del film (titolo, descrizione, prezzo, numero licenze)

**OGGETTI DELL'AMBIENTE**

* Database: contiene i dati del catalogo

|  |  |
| --- | --- |
| **Categorie** | **Scelte** |
| Categoria 1: Azione | * A1: Aggiungere un film * A2: Modificare un film * A3: Rimuovere un film |
| Categoria 2: Validità Dati | * VD1: Dati validi * VD2: Dati non validi |

|  |  |
| --- | --- |
| **TEST FRAME** | |
| **Combinazione scelte** | **Oracolo** |
| A1, VD1 | Film aggiunto con successo |
| A2, VD2 | Messaggio di errore: "Dati non validi" |
| A3, VD1 | Film rimosso con successo |

**TEST CASE**

* TC17: azione: aggiungi film "Inception" con dati validi – ORACOLO: Film aggiunto con successo
* TC18: azione: modifica film "Inception" con dati invalidi – ORACOLO: Messaggio di errore "Dati non validi"
* TC19: azione: rimuovi film "Inception" – ORACOLO: Film rimosso con successo

**10 - Pianificazione del testing**

**Responsabilità**

Il team di testing è composto da tester dedicati e sviluppatori

**Formazione** **team**

I membri del team saranno formati sui requisiti e sull'architettura del sistema

**Rischi e contingenze**

Ritardi nello sviluppo: Eseguire test paralleli su componenti già disponibili