



UNIVERSITÀ DEGLI STUDI DI SALERNO
DIPARTIMENTO DI INFORMATICA
INGEGNERIA DEL SOFTWARE 2017-2018

Test Plan 1.0

Easybook

Data: 8/01/2018

Sommario

1.PARTECIPANTI:	2
2.REVISION HISTORY:	2
3.INTRODUZIONE:	2
4.RELAZIONE CON ALTRI DOCUMENTI:	3
5.PANORAMICA DEL SISTEMA:	3
6.FUNZIONALITÀ DA TESTARE/DA NON TESTARE:	4
7.CRITERI DI SUCCESSO E DI INSUCCESSO:	5
8.APPROCCIO:	5
9.SOSPENSIONE E RIPRISTINO:	6
10.MATERIALE PER IL TESTING:	6
11.SPECIFICHE DEI TEST CASE:	7
12.TEST CASE SPECIFICATION:	13
13.TESTING SCHEDULE:	21

1.PARTECIPANTI:

Nome	Matricola
Andrea Di Lucia	0512103222

2.REVISION HISTORY:

Data	Versione	Descrizione	Autore
8/01/2018	1.0	Completamento documento	Andrea Di Lucia

3.INTRODUZIONE:

Il Testing rappresenta una delle più importanti tecniche per verificare la qualità del software, in quanto consente di analizzare, valutare e promuovere il miglioramento della correttezza dell'implementazione con riferimento alle caratteristiche definite in particolare dal modello dei requisiti software. Lo scopo di tale attività è quello di provare il sistema e rilevare problemi; si intende dunque massimizzare il numero dei test effettuati, in modo tale da poter correggere un numero cospicuo di errori. E' importante definire i casi d'uso su cui verranno testate le funzionalità del sistema. Tale attività va in contrasto con quelle svolte in precedenza: analisi, design, implementazione sono attività costruttive mentre il testing tenta di "rompere il sistema".

Questo è il test plan document dell'applicazione web "EasyBook". L'obiettivo del test è quello di provare il corretto funzionamento dei metodi delle classi di controllo e manager del sistema.

Si utilizzerà la modalità di test "black box" in quanto, data la semplicità dell'applicazione, si

è interessati unicamente al comportamento delle singole classi.

Si procederà in prima fase con il test di unità, successivamente verrà eseguito anche un test di integrazione per verificare la corretta interdipendenza delle varie classi.

4.RELAZIONE CON ALTRI DOCUMENTI:

Per la redazione di questo documento si è dovuto far riferimento ai documenti RAD, SDD e ODD; questo per comprendere al meglio il funzionamento delle singole classi e i vincoli di inserimento dei valori (come ad esempio espressi nel RAD).

Analizzando il documento RAD e in particolare i requisiti funzionali e non funzionali si sono potuti evidenziare i casi di test su tutte le operazioni riguardanti il carrello, il catalogo, il prestito, l'utente.

Per quanto riguarda i documenti SDD e ODD, sono serviti ad individuare le relazioni tra le classi e i metodi che le classi espongono.

2.1 Relazioni con il RAD

I test dovranno tenere conto delle specifiche espresse nel RAD. Naturalmente, si darà una maggiore rilevanza ai "casi limite", cioè tutti quei casi che sono al limite del nostro dominio di dati d'ingresso. Si darà rilevanza anche ai requisiti non funzionali ed ai vari modelli prodotti in fase di analisi dei requisiti.

2.2 Relazioni con il SSD

Il testing dovrà garantire la coerenza tra il software e gli obiettivi di design definiti in fase di System Design, specificati nel SDD. Si analizzeranno, quindi, i conflitti e le inconsistenze presenti tra le componenti del sistema testate e tali obiettivi. Si analizzerà, inoltre, la struttura del sistema al fine di scoprire le differenze presenti tra essa e quella prevista durante la fase di System Design.

2.3 Relazioni con l'ODD

La fase di testing dovrà considerare il contenuto del documento di Object Design, in quanto quest'ultimo rappresenta la base per la realizzazione dell'implementazione, fondamentale per il testing. Pertanto, sarà necessario effettuare il testing delle unità per individuare le differenze tra ciò che è stato stabilito in fase di Object Design ed il sistema effettivo.

5.PANORAMICA DEL SISTEMA:

EasyBook è un sistema software capace di gestire il sito di una biblioteca caratterizzato da un'architettura Client/Server. Sono presenti diversi sottosistemi, tra i quali quello che gestisce le operazioni dell'utente, quello per la gestione dei libri, quello per la gestione dei prestiti e quello per la gestione del carrello. Ognuno di questi sottosistemi, dopo essere stato diviso in componenti, sarà testato sia a livello di unità che a livelli più alti e comprensivi di più funzionalità.

Un test generale del sistema, comprensivo di tutti i sottosistemi e relativo ad ogni

interazione tra di essi sarà svolto solo durante il System Testing.

Le caratteristiche da testare per il controllo del corretto funzionamento di ciascuna funzionalità saranno:

- robustezza: capacità del sistema di reagire fornendo input non valido per il dominio applicativo;
- usabilità: analisi di ogni forma di interazione corrisposta da messaggi di aiuto (in caso di errore) o di notifica (in caso di operazioni eseguite con successo) ;
- sicurezza: verrà testato che un utente col solo interfacciamento grafico non possa intaccare dati non inerenti alla specifica dell'operazione o accedere a dati non consentiti;
- correttezza: verrà testato che le operazioni vengano eseguite correttamente così come dalla specifica dei requisiti sono stati designati.

6.FUNZIONALITÀ DA TESTARE/DA NON TESTARE:

Si è deciso di testare soltanto alcune delle funzionalità che il sistema EasyBook offre e che sono state implementate. Tali funzionalità sono quelle definite nelle fasi di Raccolta e di Analisi dei Requisiti e le funzionalità che garantiscono un controllo su queste ultime. Si è stabilito, dunque, di evitare il testing di gran parte delle funzionalità non richieste dal cliente o che servono come semplice supporto a quelle testate

Funzionalità da testare:

Le funzionalità da testare sono divise nelle seguenti categorie:

- Gestione utenti;
- Gestione prestiti;
- Gestione carrelli;
- Gestione libri.

In particolare, per lo Unit Testing si testeranno tutti i metodi che rientrano in queste categorie e che, al tempo stesso, rappresentano funzionalità richieste dal cliente o funzionalità aggiuntive, non visibili all'utente, ma importanti, se non necessarie, per offrire le prime.

Inerentemente l'Integration Testing, si svolgeranno test che aggregano in vari gruppi tutte le componenti e, quindi, le funzionalità già testate singolarmente. L'integrazione sarà svolta tenendo conto della categoria a cui le funzionalità appartengono e basandosi sulle interazioni tra tali funzionalità.

Per il System Testing saranno svolti test sull'intero sistema, senza divisioni tra funzionalità diverse. Si testerà, infatti, l'insieme completo di funzionalità.

Funzionalità da non testare:

Le funzionalità che non saranno testate sono divise in più categorie:

- funzionalità appartenenti a componenti prettamente grafiche che non offrono funzionalità di spicco per il corretto funzionamento del sistema ;
- funzionalità appartenenti a componenti molto semplici, il cui testing è implicito nella loro stessa struttura.

7. CRITERI DI SUCCESSO E DI INSUCCESSO:

Gli esiti dei test si baseranno su quanto detto nei documenti precedenti, in particolare nel RAD per quanto riguarda i vincoli di inserimento e nell'ODD per quanto riguarda l'output atteso dai metodi delle classi testate.

5.1 Criteri di successo

I criteri di successo del sistema software previsti per la fase di testing sono i seguenti:

- uguaglianza tra comportamento atteso e previsto dall'oracolo e comportamento effettivo della componente o dell'insieme di componenti;
- mancanza di errori dovuti a ragioni tecniche;
- mancanza di eccezioni generate a causa di una cattiva scrittura del codice.

5.2 Criteri di insuccesso

I criteri di insuccesso del sistema software previsti per la fase di testing sono i seguenti:

- differenza, anche minima, tra comportamento atteso e previsto dall'utente e comportamento effettivo della componente o dell'insieme di componenti;
- presenza di errori dovuti a ragioni tecniche;
- presenza di eccezioni generate a causa di una cattiva scrittura del codice.

8. APPROCCIO:

Le tipologie di testing che si svolgeranno per testare il software EasyBook saranno tre:

- *Unit Testing: un testing che coinvolge una singola unità del software e svolto indipendentemente dalle altre unità;*
- *Integration Testing: un testing che coinvolge più unità che interagiscono tra loro e che sono integrate per essere testate insieme;*
- *System Testing: un testing che coinvolge l'intero sistema e che quindi testa il software nella sua interezza, analizzando le interazioni tra tutte le unità, i gruppi di unità e le funzionalità .*

6.1 Unit testing

Sarà effettuato un testing per ogni unità del software individuata. Per ogni componente verranno costruiti uno o più test case, di tipo "Black Box". I test case "Black Box" saranno svolti analizzando unicamente il comportamento di I/O delle unità in varie situazioni, senza esaminare il codice.

6.2 Integration testing

Dopo aver testato le singole unità del software, ognuna di queste ultime sarà integrata insieme ad altre unità che interagiscono direttamente con essa. Si effettuerà, quindi, un testing che riguarderà gruppi di unità. Tale testing sarà incrementale e sarà di tipo "Top-down".

Anche nella fase di integration testing verrà utilizzato un approccio “Black box”.

6.3 System testing

Dopo aver effettuato il testing d'integrazione, si svolgerà il testing dell'intero sistema software con l'uso di molteplici test case. Si analizzeranno, in particolare, le differenze tra il comportamento effettivo software e quello previsto, dettato dal cliente e descritto nel Documento di Analisi dei Requisiti. Sarà svolto, quindi, un testing funzionale, che riguarderà l'insieme delle funzionalità del software implementato. Anche in tal caso si costruiranno test case di tipologia “Black Box”.

9.SOSPENSIONE E RIPRISTINO:

Nel testing, potrebbero esserci casi in cui debba essere sospeso e/o ripristinato.

7.1 Sospensione

Durante la fase di Unit Testing, il testing sarà sospeso se si sono riscontrati errori o fallimenti riguardanti una unità del software e la sospensione riguarderà soltanto il testing di tale unità.

Se il testing di un'unità viene sospeso, si passerà alla correzione dell'unità.

Cambiamenti ad un'unità del software non hanno influenza sullo Unit Testing delle altre.

Durante le altre due fasi di testing, ovvero Integration Testing e System Testing, il testing sarà sospeso quando saranno riscontrati errori o fallimenti per una o più funzionalità testate. In tale caso, si procederà alla correzione del codice.

7.2 Ripristino

Il ripristino del testing di una componente o un gruppo di componenti del software sarà svolto quando gli errori riscontrati per tale componente o per tale gruppo di componenti saranno corretti. Inoltre, ad ogni ripristino di testing, saranno effettuati nuovi test su ciò che si è corretto.

10.MATERIALE PER IL TESTING:

Le risorse software necessarie per il testing sono le seguenti:

- Eclipse, come ambiente di sviluppo ed esecuzione di applicazioni Java;
- Microsoft Office Word, come applicazione per la scrittura di test case e plan.

La risorse hardware sarà il computer del team.

11.SPECIFICHE DEI TEST CASE:

11.1 TC_1 Login

Parametro: numero tessera Formato: UNICODE {1-6}	
Categorie	Scelte
Lunghezza lnt	1. Lunghezza >= 1 && lunghezza <= 6 [ok] 2. Lunghezza <=0 lunghezza >6 [errore]
Esiste ent	1. Esiste nel database [if lunghezza ok] [ok] 2. Non esiste nel database [errore]

Parametro: password Formato: UNICODE {1-10}	
Categorie	Scelte
Lunghezza lp	1. Lunghezza >= 1 && lunghezza <= 10 [ok] 2. Lunghezza <=0 lunghezza >10 [errore]
Esiste ep	1. Esiste nel database ed è associato al numero di tessera [if lunghezza ok] [ok] 2. Non è associato al numero di tessera [errore]

Codice	Combinazione
TF_1.1	Lnt1,ent1,lp1,ep2
TF_1.2	Lnt1,ent1,lp1,ep1

11.2 TC_2 Registrazione

Parametro: numero tessera Formato: UNICODE {1-6}	
Categorie	Scelte
Lunghezza Int	1. Lunghezza >= 1 && lunghezza <= 6 [ok] 2. Lunghezza <=0 lunghezza >6 [errore]
Esistenza ent	1. Non esiste nel database [ok] 2. Esiste già nel database [errore]

Parametro: password Formato: UNICODE {1-10}	
Categorie	Scelte
Lunghezza lp	1. Lunghezza >= 1 && lunghezza <= 10 [ok] 2. Lunghezza <=0 lunghezza >10 [errore]

Parametro: nome Formato: [a-z A-Z] {1-20}	
Categorie	Scelte
Lunghezza ln	1. Lunghezza >= 1 && lunghezza <= 20 [ok] 2. Lunghezza <=0 lunghezza >20 [errore]
Formato fn	1. Rispetta il formato [if lunghezza ok] [ok] 2. Non rispetta il formato [errore]

Parametro: cognome Formato: [a-z A-Z] {1-20}	
Categorie	Scelte
Lunghezza lc	1. Lunghezza >= 1 && lunghezza <= 20 [ok] 2. Lunghezza <=0 lunghezza >20 [errore]
Formato fc	1. Rispetta il formato [if lunghezza ok] [ok] 2. Non rispetta il formato [errore]

Parametro: data di nascita	
Formato: [a-z A-Z] {1-10}	
Categorie	Scelte
Lunghezza ldn	1. Lunghezza >= 1 && lunghezza <= 10 [ok] 2. Lunghezza <=0 lunghezza >10 [errore]
Formato fdn	1. Rispetta il formato [if lunghezza ok] [ok] 2. Non rispetta il formato [errore]

Parametro: indirizzo	
Formato: [a-z A-Z] {1-60}	
Categorie	Scelte
Lunghezza li	1. Lunghezza >= 1 && lunghezza <= 10 [ok] 2 2. Lunghezza <=0 lunghezza >10 [errore]
Formato fi	1 Rispetta il formato [if lunghezza ok] [ok] 2 Non rispetta il formato [errore]

Codice	Combinazione
TF_2.1	Lnt2
TF_2.2	Lnt1,ent1,lp2
TF_2.3	Lnt1,ent1,lp1,ln2
TF_2.4	Lnt1,ent1,lp1,ln1,fn1,lc2
TF_2.5	Lnt1,ent1,lp1,ln1,fn1,lc1,fc1,ldn2
TF_2.6	Lnt1,ent1,lp1,ln1,fn1,lc1,fc1,ldn1,fdn1,li2
TF_2.7	Lnt1,ent1,lp1,ln1,fn1,lc1,fc1,ldn1,fdn1,li1,fi1

11.3 TC_3 Cancellazione di un libro dal catalogo

Codice	Combinazione
TF_3.1	

11.4 TC_4 Chiusura prestito

Codice	Combinazione
TF_4.1	

11.5 TC_5 Cancellazione utente

Codice	Combinazione
TF_5.1	

11.6 TC_6 Aggiunta di un libro al carrello

Codice	Combinazione
TF_6.1	

11.7 TC_7 Rimozione libro dal carrello

Codice	Combinazione
TF_7.1	

11.8 TC_8 Aggiunta libro al catalogo

Parametro: ISBN	
Formato: UNICODE {1-13}	
Categorie	Scelte
Lunghezza lis	1 Lunghezza >= 1 && lunghezza <= 13 [ok] 2 Lunghezza <=0 lunghezza >13 [errore]
Esistenza eis	1 Non esiste nel database [ok] 2 Esiste già nel database [errore]

Parametro: titolo	
Formato: [a-z A-Z] {1-50}	
Categorie	Scelte
Lunghezza lt	1 Lunghezza >= 1 && lunghezza <= 50 [ok] 2 Lunghezza <=0 lunghezza >50 [errore]
Formato ft	1 Rispetta il formato [if lunghezza ok] [ok] 2 Non rispetta il formato [errore]

Parametro: lingua	
Formato: [a-z A-Z] {1-20}	
Categorie	Scelte
Lunghezza ll	1 Lunghezza >= 1 && lunghezza <= 20 [ok] 2 Lunghezza <=0 lunghezza >20 [errore]
Formato fl	1 Rispetta il formato [if lunghezza ok] [ok] 2 Non rispetta il formato [errore]

Parametro: anno pubblicazione	
Formato: [0-9]	
Categorie	Scelte
Lunghezza lap	1 Dato inserito [ok] 2 Dato non inserito [errore]
Formato fap	1 Rispetta il formato [if lunghezza ok] [ok] 2 Non rispetta il formato [errore]

Parametro: categoria	
Formato: [a-z A-Z] {1-30}	
Categorie	Scelte
Lunghezza lc	1 Lunghezza >= 1 && lunghezza <= 30 [ok] 2 Lunghezza <=0 lunghezza >30 [errore]
Formato fc	1 Rispetta il formato [if lunghezza ok] [ok] 2 Non rispetta il formato [errore]

Parametro: autore	
Formato: [a-z A-Z] {1-20}	
Categorie	Scelte
Lunghezza la	1 Lunghezza >= 1 && lunghezza <= 20 [ok] 2 Lunghezza <=0 lunghezza >20 [errore]
Formato fn	1 Rispetta il formato [if lunghezza ok] [ok] 2 Non rispetta il formato [errore]

Parametro: casa editrice	
Formato: [a-z A-Z] {1-20}	
Categorie	Scelte
Lunghezza lce	1 Lunghezza >= 1 && lunghezza <= 20 [ok] 2 Lunghezza <=0 lunghezza >20 [errore]
Formato fce	1 Rispetta il formato [if lunghezza ok] [ok] 2 Non rispetta il formato [errore]

Codice	Combinazione
TF_8.1	Lis2,lt1,ft1,ll1,fl1,lap1,fap1,lc1,fc1,la1,fa1,lce1,fce1
TF_8.2	Lis1,eis1,lt2,ll1,fl1,lap1,fap1,lc1,fc1,la1,fa1,lce1,fce1
TF_8.3	Lis1,eis1,lt1,ft1,ll2,lap1,fap1,lc1,fc1,la1,fa1,lce1,fce1
TF_8.4	Lis1,eis1,lt1,ft1,ll1,fl1,lap2,lc1,fc1,la1,fa1,lce1,fce1
TF_8.5	Lis1,eis1,lt1,ft1,ll1,fl1,lap1,fap1,lc2,la1,fa1,lce1,fce1
TF_8.6	Lis1,eis1,lt1,ft1,ll1,fl1,lap1,fap1,lc1,fc1,la2,lce1,fce1
TF_8.7	Lis1,eis1,lt1,ft1,ll1,fl1,lap1,fap1,lc1,fc1,la1,fa1,lce2
TF_8.8	Lis1,eis1,lt1,ft1,ll1,fl1,lap1,fap1,lc1,fc1,la1,fa1,lce1,fce1

11.9 TC_9 Ricerca di un libro

Parametro: titolo	
Formato: [a-z A-Z] {1-50}	
Categorie	Scelte
Lunghezza lt	1 Lunghezza >= 1 && lunghezza <= 50 [ok] 2 Lunghezza <=0 lunghezza >50 [errore]
Formato ft	3 Rispetta il formato [if lunghezza ok] [ok] 4 Non rispetta il formato [errore]

Codice	Combinazione
TF_9.1	Lt2
TF_9.2	Lt1,ft1

11.10 TC_10 invio prestito

Codice	Combinazione
TF_10.1	

11.11 TC_11 visualizza catalogo

Codice	Combinazione
TF_11.1	

11.12 TC_12 visualizza prestiti

Codice	Combinazione
TF_12.1	

11.13 TC_13 visualizza carrello

Codice	Combinazione
TF_13.1	

11.14 TC_14 logout

Codice	Combinazione
TF_14.1	

12.TEST CASE SPECIFICATION:

12.1 TCS_1 Login

Test Frame ID	TF_1.1	
	Input	Valore
	Username	t00001

	Password	Password errata
Oracolo	Viene visualizzato un messaggio di errore “username e password sbagliati”. Non viene effettuato il login	

Test Frame ID	TF_1.2	
	Input	Valore
	Username	t00001
	Password	asdf
Oracolo	Avviene correttamente il login	

12.2 TCS_2 Registrazione

Test Frame ID	TF 2.1	
	Input	Valore
	Numero Tessera	(vuoto)
	Password	
	Nome	
	Cognome	
	Data di nascita	
	Indirizzo	
Oracolo	L'amministratore visualizza il messaggio “compila questo campo”. Registrazione non effettuata.	

Test Frame ID	TF 2.2	
	Input	Valore
	Numero Tessera	t00006
	Password	(vuoto)
	Nome	
	Cognome	
	Data di nascita	
	Indirizzo	
Oracolo	L'amministratore visualizza il messaggio “compila questo campo”.	

	Registrazione non effettuata.
--	-------------------------------

Test Frame ID	TF 2.3	
	Input	Valore
	Numero Tessera	t00006
	Password	pollo
	Nome	(vuoto)
	Cognome	
	Data di nascita	
	Indirizzo	
Oracolo	L'amministratore visualizza il messaggio "compila questo campo". Registrazione non effettuata.	

Test Frame ID	TF 2.4	
	Input	Valore
	Numero Tessera	t00006
	Password	pollo
	Nome	Marco
	Cognome	(vuoto)
	Data di nascita	
	Indirizzo	
Oracolo	L'amministratore visualizza il messaggio "compila questo campo". Registrazione non effettuata.	

Test Frame ID	TF 2.5	
	Input	Valore
	Numero Tessera	t00006
	Password	pollo
	Nome	Marco
	Cognome	D'Amore
	Data di nascita	(vuoto)
	Indirizzo	
Oracolo	L'amministratore visualizza il messaggio "compila questo campo". Registrazione non effettuata.	

Test Frame ID	TF 2.6	
	Input	Valore
	Numero Tessera	t00006
	Password	pollo
	Nome	Marco
	Cognome	D'Amore
	Data di nascita	11/11/11
	Indirizzo	(vuoto)
Oracolo	L'amministratore visualizza il messaggio "compila questo campo". Registrazione non effettuata.	

Test Frame ID	TF 2.7	
	Input	Valore
	Numero Tessera	t00006
	Password	pollo
	Nome	Marco
	Cognome	D'Amore
	Data di nascita	11/11/11
	Indirizzo	Via Mazzini
Oracolo	L'amministratore visualizza il messaggio "registrazione avvenuta con successo". Registrazione effettuata.	

12.3 TCS_3 Cancellazione libro dal catalogo

Test Frame ID	TF_3.1
Oracolo	L'amministratore cancella un libro dal catalogo

12.4 TCS_4 Chiusura prestito

Test Frame ID	TF_4.1
Oracolo	L'amministratore chiude un prestito

12.5 TCS_5 Cancellazione utente

Test Frame ID	TF_5.1
Oracolo	L'amministratore cancella un utente dalla lista utenti

12.6 TCS_6 Aggiunta libro al carrello

Test Frame ID	TF_6.1
Oracolo	L'utente aggiunge un libro al carrello

12.7 TCS_7 Rimozione libro dal carrello

Test Frame ID	TF_7.1
Oracolo	L'utente rimuove un libro dal carrello

12.8 TCS_8 Aggiunta libro al catalogo

Test Frame ID	TF_8.1	
	Input	Valore
	ISBN	(vuoto)
	Titolo	Mistborn
	Lingua	Italiano
	Anno Pubblicazione	2015
	Categoria	Fantasy
	Autore	Sanderson
	Casa Editrice	Fanucci
Oracolo	Viene notificato un messaggio di errore. Non viene effettuato l'inserimento	

Test Frame ID	TF_8.2	
	Input	Valore
	ISBN	3482930492837
	Titolo	(vuoto)
	Lingua	Italiano
	Anno Pubblicazione	2015
	Categoria	Fantasy
	Autore	Sanderson
	Casa Editrice	Fanucci
Oracolo	Viene notificato un messaggio di errore. Non viene effettuato l'inserimento	

Test Frame ID	TF_8.3	
	Input	Valore
	ISBN	3482930492837
	Titolo	Mistborn
	Lingua	(vuoto)
	Anno Pubblicazione	2015
	Categoria	Fantasy
	Autore	Sanderson
	Casa Editrice	Fanucci
Oracolo	Viene notificato un messaggio di errore. Non viene effettuato l'inserimento	

Test Frame ID	TF_8.4	
	Input	Valore
	ISBN	3482930492837
	Titolo	Mistborn
	Lingua	italiano
	Anno Pubblicazione	(vuoto)
	Categoria	Fantasy
	Autore	Sanderson
	Casa Editrice	Fanucci
Oracolo	Viene notificato un messaggio di errore. Non viene effettuato l'inserimento	

Test Frame ID	TF_8.5	
	Input	Valore
	ISBN	3482930492837
	Titolo	Mistborn
	Lingua	italiano
	Anno Pubblicazione	2015
	Categoria	(vuoto)
	Autore	Sanderson
	Casa Editrice	Fanucci
Oracolo	Viene notificato un messaggio di errore. Non viene effettuato l'inserimento	

Test Frame ID	TF_8.6	
	Input	Valore
	ISBN	3482930492837
	Titolo	Mistborn
	Lingua	italiano
	Anno Pubblicazione	2015
	Categoria	Fantasy
	Autore	(vuoto)
	Casa Editrice	Fanucci
Oracolo	Viene notificato un messaggio di errore. Non viene effettuato l'inserimento	

Test Frame ID	TF_8.7	
	Input	Valore
	ISBN	3482930492837
	Titolo	Mistborn
	Lingua	italiano
	Anno Pubblicazione	2015
	Categoria	Fantasy
	Autore	Sanderson
	Casa Editrice	(vuoto)
Oracolo	Viene notificato un messaggio di errore. Non viene effettuato l'inserimento	

Test Frame ID	TF_8.8	
	Input	Valore
	ISBN	3482930492837
	Titolo	Mistborn
	Lingua	italiano
	Anno Pubblicazione	2015
	Categoria	Fantasy
	Autore	Sanderson
	Casa Editrice	Fanucci
Oracolo	Viene effettuato l'inserimento	

12.9 TCS_9 Ricerca libro nel catalogo

Test Frame ID	TF_9.1	
	Input	Valore
	Titolo	(vuoto)
Oracolo	Non viene effettuata la ricerca	

Test Frame ID	TF_9.2	
	Input	Valore
	Titolo	Mistborn
Oracolo	Viene effettuata la ricerca	

12.10 TCS_10 Invio prestito

Test Frame ID	TF_10.1	
Oracolo	L'utente invia il prestito	

12.11 TCS_11 Visualizza Catalogo

Test Frame ID	TF_11.1	
Oracolo	L'utente (o l'amministratore) visualizza il catalogo	

12.12 TCS_12 Visualizza Prestiti

Test Frame ID	TF_12.1	
Oracolo	L'utente (o l'amministratore) visualizza la pagina prestiti	

12.13 TCS_13 Visualizza carrello

Test Frame ID	TF_13.1	
Oracolo	L'utente visualizza il carrello	

12.14 TCS_14 Logout

Test Frame ID	TF_14.1
Oracolo	L'utente (o l'amministratore) effettua il logout

13. TESTING SCHEDULE:

13.1 Gestione dei rischi

I possibili rischi generati dalle attività di testing sono stati minimizzati diminuendo le componenti del sistema da implementare e, quindi, testare. Inoltre, effettuando un testing di tipo funzionale viene limitato lo sviluppo di stub e driver per il testing delle singole componenti e quindi l'introduzione di nuovi errori nel nuovo codice di cui si sarebbero composti.

D'altro canto il testing di funzionalità rallenta l'individuazione di errori qualora un caso di test avesse esito positivo, poiché l'utilizzo di più componenti per il test di una singola funzionalità estranea dall'ipotesi di totale correttezza di ogni componente interessata. Il risultato atteso è di riscontrare al più un errore parziale per funzionalità: le componenti interagiscono fra loro e la funzionalità viene eseguita ma non in modo completamente corretto.

Qualora la fase di testing evidenziasse un numero di errori maggiore rispetto alla media attesa, viene pianificato un impegno maggiore dei membri del team sulle attività di testing ed in casi estremi l'abbandono delle altre attività finché errori gravi (funzionalità non corretta, risultati errati, modifiche apportate in modo errato) non vengano risolti.

13.2 Organizzazione delle attività

Le attività di testing devono svolgersi sulle singole funzionalità divise nei livelli di suddivisione del sistema, rispettando le direttive indicate dal documento di system design.

13.3 Schedulazione delle attività

Le attività di testing sono previste per il seguente periodo:

- inizio testing: 9 gennaio;
- fine testing: 11 gennaio.