



Test Plan



Riferimento	
Versione	1.0
Data	19/01/2022
Destinatario	Prof.ssa F Ferrucci, Prof. F.Palomba
Presentato da	ALESSIA AMATO, ALFONSO MADDALONI, ANTONIO GIORDANO, ENRIQUE CAMACHO GARCÍA, MARIANNA VUJKO, LUCA MORELLI, GIUSEPPE RAGOSTA, MARCO PALMISCIANO
Approvato da	ALESSIA AMATO, ALFONSO MADDALONI, ANTONIO GIORDANO, ENRIQUE CAMACHO GARCÍA, MARIANNA VUJKO, LUCA MORELLI, GIUSEPPE RAGOSTA, MARCO PALMISCIANO



Revision History

Data	Versione	Descrizione	Autori
11/12/2022	v0.1	Test Case	Tutti
13/01/2022	v0.2	Introduzione, Panoramica del sistema e Relazione con altri documenti, Pass/fail criteria	Alessia Amato
13/01/2022	v0.3	Feature da testare/non testare, Approccio	Alessia Amato, Alfonso maddaloni
15/01/2021	v0.4	Modifica e revisione ODL e PDL	Luca Morelli
19/01/2021	v1.0	Revisione documento	tutti



Sommario

Revision History	3
1.Introduzione	5
2. Relazione con altri documenti	5
3. Panoramica del Sistema	5
4. Features da testare/non testare	6
5. Pass/fail criteria	7
6. Approccio	8
Testing di sistema	8
Testing di integrazione	8
Testing di unità	8
Ispezione del codice	8
7. Sospensione e ripristino	9
Criteri di sospensione	9
Criteri di ripristino	9
8. Materiale di Testing	9
9. Test cases	10
ACC_TC_1 Log-in	10
ACC_TC_3 Registrazione	11
IBC_TC_2 Creazione book club	13
ODL_TC_3 Modifica obiettivo di lettura	14
ODL_TC_7 Definizione obiettivo di lettura	15
SLI_TC_1 Ricerca libro	16
INV_TC_1 Invito utente	16
INV_TC_2 Annulla invito	17
PDL_TC_2 Modifica progresso di lettura	17
10. Testing schedule	18



1.Introduzione

Book Club si prefigge lo scopo di avvicinare le persone all'hobby della lettura offrendo una piattaforma utile al confronto e alla discussione letteraria. L'idea nasce a causa della sempre minore propensione dei giovani ad avvicinarsi alla lettura nel tempo libero. Grazie a Book Club, la lettura viene resa più coinvolgente, dinamica e immersa nel sociale con il coinvolgimento di altre persone all'interno dei gruppi di lettura, facilitando ulteriormente gli incontri di chi già prima di usare questo sistema organizzava attività di questo genere.

2. Relazione con altri documenti

Il seguente documento farà riferimento agli altri documenti di testing:

- Requirements Analysis Document (RAD);
- System Design Document (SDD);
- Object Diagram Document (ODD).

3. Panoramica del Sistema

Il sistema farà uso dello stile architetturale repository per realizzare i vari sottosistemi contenenti logiche di gestione dati persistenti con sistema Model View Controller (MVC).

Per quanto riguarda il front-end, sarà utilizzato il framework JavaScript per app mobile **React Native** (<https://reactnative.dev/docs/getting-started>).

Per quanto riguarda il back-end, sarà utilizzato il framework **NestJS** (<https://nestjs.com>). Si basa sul linguaggio **TypeScript**, una modernizzazione del linguaggio JavaScript (<https://www.typescriptlang.org>).

Per la gestione dei dati persistenti si è scelto un database relazionale gestito tramite il DBMS **MySQL**.



4. Features da testare/non testare

Di seguito elencheremo una serie di features che verranno testate:

- Registrazione;
- Login;
- Creazione Book Club;
- Ricerca Libro;
- Modifica Obiettivo Di Lettura;
- Definizione Obiettivo Di Lettura;
- Invito Utente;
- Annulla Invito;
- Modifica Progresso Di Lettura.

Le funzionalità per cui non si andrà a fare testing sono derivate da requisiti funzionali a bassa o media priorità o, comunque, sono funzionalità che non richiedono l'inserimento manuale di parametri da parte dell'utente (ad esempio operazioni di visualizzazione).



5. Pass/fail criteria

Le attività di testing hanno lo scopo di individuare degli errori all'interno del sistema in modo da correggerli prima che il sistema arrivi nelle mani del cliente.

Tramite quello che viene definito oracolo, ovvero il risultato che ci aspettiamo dall'esecuzione di quel test case, valutiamo il test case in questione.

Se il risultato di un test darà come esito un valore diverso dall'oracolo, esso avrà fallito (fail); se invece l'esito del test darà un valore uguale all'oracolo, esso avrà successo (pass).

Il testing sarà ritenuto valido se le seguenti condizioni verranno rispettate:

- testare tutti i requisiti funzionali ad alta priorità
- ogni qualvolta vengono aggiunte o modificate caratteristiche del sistema, si dovranno effettuare test di regressione
- branch coverage da raggiungere $\geq 75\%$



6. Approccio

Il testing di tutto il sistema consta di 3 fasi: testing di unità, testing di integrazione e testing di sistema.

La progettazione dei casi di test di sistema avverrà prima dell'implementazione secondo il modello black-box; dopodiché verranno perfezionati. Inoltre, durante l'implementazione verranno progettati ed eseguiti i test di unità e di integrazione.

Testing di sistema

Per il testing di sistema verrà utilizzato il tool Selenium IDE Chrome. Per poterlo utilizzare si eseguirà in localhost l'applicazione, in particolare sul web browser Chrome. Così, Selenium IDE Chrome, ci consentirà di registrare le azioni dell'utente all'interno dell'applicazione in esecuzione su browser con lo scopo di implementare ed eseguire i test case di sistema.

Testing di integrazione

Si utilizzerà Postman per fare richieste HTTP direttamente alle API in modo da testare i controller in integrazione con i service del sistema back-end.

Postman si servirà di Visual Studio Code e del database scritto in MySQL.

Testing di unità

Verrà usato Jest per testare le componenti contenenti business logics delle API, ovvero i service, in maniera indipendente dalle richieste web.

Per ogni funzione verrà creato un test in un altrettanto file di tipo ".spec.ts" dove si verificherà se la funzione non dà fault.

Ispezione del codice

Per garantire una qualità del codice accettabile e standardizzata in tutto il team, si è scelto come strumento di code formatting ESLint. Si è configurato ESLint in modo da seguire degli standard che permettono una programmazione altamente accessibile e contemporaneamente leggibile del codice.



7. Sospensione e ripristino

In questa sezione verranno elencati i criteri di sospensione del test e le attività che verranno ripetute quando avviene un ripristino.

Criteri di sospensione

Il testing non verrà sospeso finché non sarà terminato, anche se dovessero esserci failures. Durante l'esecuzione il test potrà essere sospeso momentaneamente, nel caso vi sia un errore nella definizione di uno dei test in questione.

Criteri di ripristino

Il testing verrà ripreso dopo aver risolto gli errori individuati.

8. Materiale di Testing

L'hardware utilizzato per il testing è un comune computer.



9. Test cases

La category partition verrà utilizzata per definire i test frames. Per l'output si farà uso di un oracolo umano, a causa dell'assenza di specifiche formali/semi-formali.

ACC_TC_1 Log-in

Parametro: E-Mail	
Formato: [a-zA-Z0-9._%~]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,4}	
Categorie	Scelte
Lunghezza LE	<ol style="list-style-type: none">1. lunghezza < 6 [errore]2. 6 <= lunghezza <= 100 [property LunghezzaLE_ok]3. lunghezza > 100 [errore]
Formato FE	<ol style="list-style-type: none">1. Non rispetta il formato [if LunghezzaLE_ok] [errore]2. Rispetta il formato [if LunghezzaLE_ok] [property LunghezzaFE_ok]
Esiste EE	<ol style="list-style-type: none">1. Non esiste nel db [if LunghezzaLE_ok and FormatoFE_ok] [errore]2. Esiste nel db [if LunghezzaLE_ok and FormatoFE_ok] [property EsisteEE_ok]

Parametro: Password	
Formato: (?=.*[0-9])(?=.*[a-zA-Z]){8+}	
Categorie	Scelte
Lunghezza LP	<ol style="list-style-type: none">1. lunghezza < 8 [errore]2. 8 <= lunghezza <= 32 [property LunghezzaLP_ok]3. lunghezza > 16 [errore]
Formato FP	<ol style="list-style-type: none">1. Non rispetta il formato [if LunghezzaLP_ok] [errore]2. Rispetta il formato [if LunghezzaLP_ok] [property FormatoFP_ok]
Corrisponde CP	<ol style="list-style-type: none">1. Non corrisponde alla password della e-mail indicata [if EsisteEP_ok] [errore]2. Corrisponde alla password della e-mail indicata [if EsisteEP_ok] [property CorrispondeCP_ok]



Codice	Combinazione	Esito
TC_1.1	LE1	FAIL
TC_1.2	LE3	FAIL
TC_1.3	LE2.FE1	FAIL
TC_1.4	LE2.FE2.EE1	FAIL
TC_1.5	LE2.FE2.EE2.LP1	FAIL
TC_1.6	LE2.FE2.EE2.LP3	FAIL
TC_1.7	LE2.FE2.EE2.LP2.FP1	FAIL
TC_1.8	LE2.FE2.EE2.LP2.FP2.CP1	FAIL
TC_1.9	LE2.FE2.EE2.LP3.FP2.CP2	PASS

ACC_TC_3 Registrazione

Parametro: Nome Formato: [A-Z a-z]	
Categorie	Scelte
Lunghezza LN	1. Lunghezza < 2 [errore] 2. 2 <= lunghezza <= 20 [property LunghezzaLN_ok] 3. Lunghezza > 20 [errore]
Formato FN	1. Non rispetta il formato [if LunghezzaLN_ok] [errore] 2. Rispetta il formato [if LunghezzaLN_ok] [property FormatoFN_ok]

Parametro: Cognome Formato: [A-Z a-z]	
Categorie	Scelte
Lunghezza LC	1. lunghezza < 2 [errore] 2. 2 <= lunghezza <= 20 [property LunghezzaLC_ok] 3. lunghezza > 20 [errore]
Formato FC	1. Non rispetta il formato [if LunghezzaLC_ok] [errore] 2. Rispetta il formato [if LunghezzaLC_ok] [property FormatoFC_ok]



Parametro: E-Mail	
Formato: [a-zA-Z0-9._%~]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,4}	
Categorie	Scelte
Lunghezza LE	<ol style="list-style-type: none"> 1. lunghezza < 6 [errore] 2. 6 <= lunghezza <= 100 [property LunghezzaLE_ok] 3. lunghezza < 100 [errore]
Formato FE	<ol style="list-style-type: none"> 1. Non rispetta il formato [if LunghezzaLE_ok] [errore] 2. Rispetta il formato [if LunghezzaLE_ok] [property FormatoFE_ok]
Esiste EE	<ol style="list-style-type: none"> 1. Esiste nel db [if LunghezzaLE_ok and FormatoFE_ok] [errore] 2. Non esiste nel db [if LunghezzaLE_ok and FormatoFE_ok] [property EsisteEE_ok]

Parametro: Password	
Formato: (?=.*[0-9]) (?=.*[a-z A-Z]){8+}	
Categorie	Scelte
Lunghezza LP	<ol style="list-style-type: none"> 1. lunghezza < 8 [errore] 2. 8 <= lunghezza <= 32 [property LunghezzaLP_ok] 3. lunghezza > 16 [errore]
Formato FP	<ol style="list-style-type: none"> 1. Non rispetta il formato [if LunghezzaLP_ok] [errore] 2. Rispetta il formato [if LunghezzaLP_ok] [property FormatoFP_ok]



Codice	Combinazione	Esito
TC_3.1	LN1	FAIL
TC_3.2	LN3	FAIL
TC_3.3	LN2.FN1	FAIL
TC_3.4	LN2.FN2.LC1	FAIL
TC_3.5	LN2.FN2.LC3	FAIL
TC_3.6	LN2.FN2.LC2.FC1	FAIL
TC_3.7	LN2.FN2.LC2.FC2.LE1	FAIL
TC_3.8	LN2.FN2.LC2.FC2.LE3	FAIL
TC_3.9	LN2.FN2.LC2.FC2.LE2.FE1	FAIL
TC_3.10	LN2.FN2.LC2.FC2.LE2.FE2.EE1	FAIL
TC_3.11	LN2.FN2.LC2.FC2.LE2.FE2.EE2.LP1	FAIL
TC_3.12	LN2.FN2.LC2.FC2.LE2.FE2.EE2.LP3	FAIL
TC_3.13	LN2.FN2.LC2.FC2.LE2.FE2.EE2.LP2.FP1	FAIL
TC_3.14	LN2.FN2.LC2.FC2.LE2.FE2.EE2.LP2.FP2	PASS

IBC_TC_2 Creazione book club

Parametro: Nome Formato: [A-Z a-z][0-9]	
Categorie	Scelte
Lunghezza LN	1. Lunghezza < 2 [errore] 2. 2 <= lunghezza <= 20 [property LunghezzaLN_ok] 3. Lunghezza > 20 [errore]
Formato FN	1. Non rispetta il formato [if LunghezzaLN_ok] [errore] 2. Rispetta il formato [if LunghezzaLN_ok] [property FormatoFN_ok]

Codice	Combinazione	Esito
TC_2.1	LN1	FAIL
TC_2.2	LN3	FAIL
TC_2.3	LN2.FN1	PASS
TC_2.4	LN2.FN2	PASS



ODL_TC_3 Modifica obiettivo di lettura

Parametro: book club	
Categorie	Scelte
Selezionato SE	<ol style="list-style-type: none"> 1. Non viene selezionato [errore] 2. Viene selezionato [property SelezionatoSE_ok]

Parametro: Obiettivo di lettura Formato: [0-9]	
Categorie	Scelte
ModificaValore ODLMV	<ol style="list-style-type: none"> 1. valoreAggiornato < valore [if ModificabileODLM_ok] [errore] 2. valoreAggiornato > pagineLibro [if ModificabileODLM_ok] [errore] 3. valoreAggiornato = valore [if ModificabileODLM_ok] [errore] 4. valoreAggiornato = pagineLibro [if ModificabileODLM_ok] [property ModificabileODLMV_ok] 5. valore < valoreAggiornato < pagineLibro [if ModificabileODLM_ok] [property ModificabileODLMV_ok]

Parametro: stato Obiettivo di lettura	
Categorie	Scelte
Modificabile ODLM	<ol style="list-style-type: none"> 1. È in stato "raggiunto" [if SelezionatoSE_ok] [errore] 2. È in stato "non raggiunto" [if SelezionatoSE_ok] [property ModificabileODLM_ok]

Codice	Combinazione	Esito
OLD_TC_3.1	SE1	FAIL
ODL_TC_3.2	SE2.ODLM1	FAIL
ODL_TC_3.3	SE2.ODLM2.ODLMV1	FAIL
ODL_TC_3.4	SE2.ODLM2.ODLMV2	FAIL
OLD_TC_3.5	SE2.ODLM2.ODLMV3	FAIL
ODL_TC_3.6	SE2.ODLM2.ODLMV4	PASS
ODL_TC_3.7	SE2.ODLM2.ODLMV5	PASS



ODL_TC_7 Definizione obiettivo di lettura

Parametro: book club	
Categorie	Scelte
Selezionato SE	1. Non viene selezionato [errore] 2. Viene selezionato [property SelezionatoSE_ok]

Parametro: Obiettivo di lettura Formato: [0-9]	
Categorie	Scelte
NuovoValore ODLNV	1. nuovoValore < valore [if SelezionatoSE_ok] [errore] 2. nuovoValore > pagineLibro [if SelezionatoSE_ok] [errore] 3. nuovoValore = valore [if SelezionatoSE_ok] [errore] 4. valoreAggiornato = pagineLibro [if SelezionatoSE_ok] [property ValoreODLNV_ok] 5. valore < nuovoValore < pagineLibro [if SelezionatoSE_ok] [property ValoreODLNV_ok]

Codice	Combinazione	Esito
OLD_TC_7.1	SE1	FAIL
ODL_TC_7.2	SE2.ODLNV1	FAIL
ODL_TC_7.3	SE2.ODLNV2	FAIL
ODL_TC_7.4	SE2.ODLNV3	FAIL
ODL_TC_7.5	SE2.ODLNV4	PASS
ODL_TC_7.6	SE2.ODLNV5	PASS



SLI_TC_1 Ricerca libro

Parametro: Ricerca libro	
Categorie	Scelte
Selezionato SE	1. Non viene selezionato [errore] 2. Viene selezionato [property SelezionatoSE_ok]
Disponibile DA	1. Il libro non è esistente [if SelezionatoSE_ok] [errore] 2. Il libro è esistente [if SelezionatoSE_ok] [property DisponibileDA_ok]

Codice	Combinazione	Esito
SLI_TC_1.1	SE1	FAIL
SLI_TC_1.2	SE2.DA1	FAIL
SLI_TC_1.3	SE2.DA2	PASS

INV_TC_1 Invito utente

Parametro: email Utente autenticato Formato: [a-zA-Z0-9+_.-]+@[a-zA-Z0-9.-]+	
Categorie	Scelte
Formato FE	1. Non rispetta il formato [errore] 2. Rispetta il formato [property FormatoFE_ok]
Esiste EE	1. Email inesistente nel db [if FormatoFE_ok] [errore] 2. Email esiste nel db [if FormatoFE_ok] [property EsisteEE_ok]
Valida VE	1. Email già presente nel proprio book club [if FormatoFE_ok and EsisteEE_ok] [errore] 2. Email non presente nel proprio book club [if FormatoFE_ok and EsisteEE_ok] [property ValidaVE_ok]

Codice	Combinazione	Esito
INV_TC_1.1	FE1	FAIL
INV_TC_1.2	FE2.EE1	FAIL



INV_TC_1.3	FE2.EE2.VE1	FAIL
INV_TC_1.4	FE2.EE2.VE2	PASS

INV_TC_2 Annulla invito

Parametro: Schermata inviti	
Categorie	Scelte
Selezionato SE	1. Non viene selezionato [errore] 2. Viene selezionato [property SelezionatoSE_ok]

Parametro: Invito	
Categorie	Scelte
Attesa AT	1. L'invito è stato accettato [errore] 2. L'invito è stato rifiutato [errore] 3. L'invito è in attesa [property AttesaAT_ok]

Codice	Combinazione	Esito
INV_TC_2.1	SE1	FAIL
INV_TC_2.2	SE2.AT1	FAIL
INV_TC_2.3	SE2.AT2	FAIL
INV_TC_2.4	SE2.AT3	PASS

PDL_TC_2 Modifica progresso di lettura

Parametro: book club	
Categorie	Scelte
Selezionato SE	1. Non viene selezionato [errore] 2. Viene selezionato [property SelezionatoSE_ok]

Parametro: Progresso di lettura	
Categorie	Scelte
NuovoValore PDLNV	1. nuovoValore < 0 [if SelezionatoSE_ok] [errore] 2. nuovoValore = 0 [if SelezionatoSE_ok] [errore] 3. nuovoValore > 0 [if SelezionatoSE_ok] [property ValorePDLV_ok]



Codice	Combinazione	Esito
PDL_TC_2.1	SE1	FAIL
PDL_TC_2.2	SE2.PDLNV1	FAIL
PDL_TC_2.3	SE2.PDLNV2	FAIL
PDL_TC_2.3	SE2.PDLNV3	PASS

10. Testing schedule

Appena dopo la fase di design necessaria, avverrà la pianificazione del testing.

I test case verranno scritti prima della creazione del codice e la loro esecuzione avverrà sia durante che al termine della compilazione.

Per altre informazioni, ci si può rivolgere ai documenti di testing in cui si monitorerà e analizzerà più approfonditamente tutta la parte di testing.