



TP Test Plan

iLike

Riferimento	
Versione	3.1
Data	13/01/2023
Destinatario	Prof. Carmine Gravino e tutor IS a.a. 2022/23
Presentato da	Costante Luigina, Giorgione Francesco, Lo Conte Simona, Napolillo Marta
Approvato da	



Revision History

Data	Versione	Descrizione	Autori
07/12/2022	0.1	Aggiunte in tutto il documento	Luigina Costante Simona Lo Conte Marta Napolillo
07/12/2022	0.2	Modifica test case	Francesco Giorgione
07/12/2022	0.3	Modifiche varie	Simona Lo Conte
09/12/2022	0.4	Modifica formattazione	Francesco Giorgione
12/12/2022	0.5	Modifiche varie	Tutto il team
14/12/2022	0.6	Aggiunta dei link alla repository di GitHub e al Drive condiviso	Simona Lo Conte
08/01/2023	0.7	Modifiche test cases 9.2	Luigina Costante
08/01/2023	0.8	Modifiche test cases 9.3	Luigina Costante
08/01/2023	1.0	Modifica Relazioni con altri documenti e Approccio	Marta Napolillo
09/01/2023	2.0	Modifica test case 9.2 e 9.3	Francesco Giorgione
09/01/2023	3.0	Modifica test case 9.2 e 9.3	Francesco Giorgione
13/01/2023	3.1	Revisione	Francesco Giorgione



Sommario

Revision History	2
1 Introduzione	4
2 Relazioni con altri documenti	4
3 Panoramica del sistema	4
4 Features da testare/da non testare	5
5 Pass/Fail criteria	5
6 Approccio	5
7 Sospensione e ripristino	6
8 Materiale di testing	6
9 Test cases	7
10 Testing schedule	10



1 Introduzione

L'obiettivo principale di iLike è quello di realizzare una piattaforma unificata che consenta di recensire film, serie TV, libri e album musicali. Lo scopo di questo documento è descrivere ed analizzare le attività di testing, al fine di garantire la corretta esecuzione delle funzionalità del sistema.

2 Relazioni con altri documenti

Al fine di individuare una corretta individuazione dei test case, si fa riferimento ai seguenti documenti:

- **RAD (Requirements Analysis Document)**, in quanto i test case sono elaborati in relazione ai requisiti funzionali e non funzionali.
- **SDD (System Design Document)**, in quanto i test case devono rispettare la suddivisione in sottosistemi presentata nell'SDD.
- **ODD (Object Design Document)**, in quanto i test di unità fanno riferimento alla divisione dei package, riportata nell'implementazione e descritta nell'ODD.

Tali documenti sono reperibili ai link seguenti:

- [Repository GitHub](#)
- [Drive condiviso](#)

3 Panoramica del sistema

Il sistema proposto è basato sullo stile architetturale Three-Tier che organizza i sottosistemi in tre layer:

- Livello di Storage, che si occupa della gestione dei dati persistenti;
- Livello Application, che si occupa della logica di applicazione dove vengono eseguite le funzionalità del sistema;
- Livello Presentation, che si occupa dell'interazione con l'utente.



4 Features da testare/da non testare

Di seguito vengono specificate le feature del sistema che si intende testare, divise in base ai sottosistemi di appartenenza.

Gestione Recensioni

- Pubblicazione recensioni

Gestione Account

- Login (iscritto)

Gestione Segnalazioni

- Cancellazione recensione

Gestione Liste

- Creazione liste

5 Pass/Fail criteria

Lo scopo delle attività di testing è identificare la presenza di faults all'interno del sistema per massimizzare l'affidabilità del sistema prodotto.

L'esito di un test case è valutato mediante un oracolo, inteso come il risultato atteso della sua esecuzione, basandosi sui requisiti. In particolare, per ciascun test case, il risultato può essere

- **fail** se, dato un input al sistema, l'output ottenuto è diverso dall'output atteso (il test ha avuto successo perché ha individuato un fault);
- **pass**: se, dato un input al sistema, l'output ottenuto è uguale all'output atteso (il test è fallito perché non ha individuato alcun fault).

6 Approccio

Il testing realizzato si compone del testing di unità e del testing di sistema.



Testing di sistema

Il testing di sistema, in particolare il testing funzionale, ha lo scopo di testare le funzionalità del sistema proposto al fine di validare i requisiti funzionali presenti nel RAD. Ha lo scopo di trovare le differenze tra i requisiti funzionali e il sistema realizzato.

Testing di unità

Il testing di unità ha lo scopo di testare singoli metodi o moduli del sistema, con l'obiettivo di individuare differenze tra il comportamento atteso e il comportamento effettivo.

7 Sospensione e ripristino

Criteri di sospensione

Il testing potrà essere momentaneamente sospeso solo nel caso di rilevazione di un errore nella definizione di uno dei test. Non sarà sospeso se sarà rilevata una failure.

Criteri di ripristino

Il testing verrà ripreso dopo aver risolto i fault individuati.

8 Materiale di testing

L'hardware utilizzabile per l'attività di testing consiste in uno smartphone oppure un computer con apposito emulatore di dispositivo smartphone Android.



9 Test cases

9.1 Pubblicazione recensioni

Parametro: Testo	
Nome categoria	Scelte per la categoria
Lunghezza [LN]	1. LN < 3 [errore] 2. LN > 1000 [errore] 3. 3 <= LN <= 1000 [LN_OK]

Parametro: Valutazione	
Nome categoria	Scelte per la categoria
Stelle selezionate [RG]	1. RG = 0 2. RG >= 1

Test case ID	Test frame	Esito
TC_1_1	LN_1	Errato: recensione troppo corta
TC_1_2	LN_2	Errato: recensione troppo lunga
TC_1_3	LN_3, RG_1	Errato: valutazione non inserita
TC_1_4	LN_3, RG_2	Corretto

9.2 Login

Parametro: username	
Nome categoria	Scelte per la categoria
Match username [MU]	1. Match con email = false AND match con nickname = false [errore] 2. Match con email = true OR match con nickname = true [MU_OK]

Nota: si ricorda che, come specificato nel RAD, l'iscritto può eseguire il login utilizzando come username sia il nickname che l'email; il gestore, invece, non avendo un nickname, può utilizzare soltanto l'email.



Parametro: password	
Formato richiesto: $^{\wedge}(\text{?}=.^*[A-Za-z])(\text{?}=.^*\backslash d)[A-Za-z\backslash d]\{8,\}\$$	
Nome categoria	Scelte per la categoria
Match password [MP]	<ol style="list-style-type: none">1. Password corretta = false [errore]2. Password corretta = true [MP_OK]

Test case ID	Test frame	Esito
TC_2_1	MU_1	Errato: username non matcha con nessun nickname/email
TC_2_2	MU_2, MP_1	Errato: la password non è corretta
TC_2_3	MU_2, MP_2	Corretto

9.3 Cancellazione recensione

Parametro: Motivazione cancellazione	
Nome categoria	Scelte per la categoria
Lunghezza [LN]	<ol style="list-style-type: none">1. $LN < 1$ [errore]2. $LN > 300$ [errore]3. $1 \leq LN \leq 300$ [LN_OK]

Test case ID	Test frame	Esito
TC_3_1	LN_1	Errato: motivazione cancellazione troppo corta
TC_3_2	LN_2	Errato: motivazione cancellazione troppo lunga



TC_3_3	LN_3	Corretto
--------	------	----------

9.4 Creazione liste

Parametro: Nome lista	
Nome categoria	Scelte per la categoria
Lunghezza [LN]	<ol style="list-style-type: none">1. LN < 1 [errore]2. LN > 50 [errore]3. 1 <= LN <= 50 [LN_OK]
Nome già in uso [NR]	<ol style="list-style-type: none">1. NR = true [errore]2. NR = false [RN_OK]

Test case ID	Test frame	Esito
TC_4_1	LN_1	Errato: nome lista troppo corto
TC_4_2	LN_2	Errato: nome lista troppo lungo
TC_4_3	LN_3, NR_1	Errato: nome della lista già in uso
TC_4_4	LN2, NR_2	Corretto

10 Testing schedule

Le attività di pianificazione del testing saranno prodotte subito dopo la fase di design. L'esecuzione dei test sarà eseguita sia durante che dopo l'implementazione del sistema. Una volta concluso lo sviluppo, tutti i test saranno rieseguiti per garantirne il corretto funzionamento e produrre i report finali.