# Università degli Studi di Salerno

Corso di Ingegneria del Software

## 7Movies Unit Test Design (UTD) Versione 1.0





Data: 24/01/2020

| $C_{\Omega}$ | ordina   | tore | del | nroc | etto.  |
|--------------|----------|------|-----|------|--------|
| υU           | oi uii a | lule | uei | טוטנ | iello. |

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

#### Partecipanti:

| Nome              | Matricola  |
|-------------------|------------|
| Luca Siviero      | 0512104862 |
| Mario Castigliano | 0512104718 |

| Scritto da: | Luca Siviero |
|-------------|--------------|
|             |              |

## **Revision History**

| 1.01.0.0.1 |          |               |              |
|------------|----------|---------------|--------------|
| Data       | Versione | Descrizione   | Autore       |
| 24/01/2020 | 1.0      | Completamento | Luca Siviero |
|            |          |               |              |
|            |          |               |              |
|            |          |               |              |
|            |          |               |              |
|            |          |               |              |
|            |          |               |              |

### **Indice**

- 1. INTRODUZIONE
- 1.1. Identificativo del documento
- 1.2. Scopo
- 1.3. Riferimenti
- 2. DETTAGLI PER IL DESIGN DEL TEST DI UNITÀ
- 2.1. Funzionalità da testare
- 2.2. Approccio
- 2.3. Identificazione del test
- 2.4. Criteri di successo/ fallimento
- 2.5. Prodotti del test
- 3. GENERALI
- 3.1. Glossario

#### 1. INTRODUZIONE

Lo scopo di questo documento è quello di pianificare il design del test di unità del software 7Movies: verranno riportate le strategie adottate per il testing, gli strumenti utilizzati e le funzionalità testate. Nello specifico saranno riportati tutti i metodi esercitati sulle classi prese in considerazione.

#### 1.1. Identificativo del documento

Il seguente documento è identificato in modo univoco dalle seguenti proprietà: data di rilascio, nomi di chi ha partecipato alla stesura e alla revisione del suddetto; tali informazioni sono riportate nel frontespizio.

### 1.2. Scopo

Il documento fa riferimento alla pianificazione del test di unità per il software 7Movies, all'interno del quale si verificano le singole unità di una classe e si controlla che i metodi e le classi abbiano il comportamento atteso. I membri del team hanno testato esclusivamente i Bean delle classi implementate nel sistema.

#### 1.3. Riferimenti

- 7Movies\_RAD\_v\_1.5.doc
- 7Movies\_SDD\_v\_1.1.doc
- 7Movies\_ODD\_v\_1.1.doc
- 7Movies\_TP\_v\_1.1.doc
- 7Movies\_TCS\_v\_1.1.doc

#### 2. DETTAGLI PER IL DESIGN DEL TEST DI UNITA'

In questa sezione del documento verranno forniti i dettagli relativi alle funzionalità e all'approccio utilizzati per eseguire lo unit test design.

#### 2.1. Funzionalità da testare

Il documento si propone di verificare il test di unità effettuato sui Bean presenti nel sistema; in particolare:

- Verrà effettuato il test della classe UserBean.java
- Verrà effettuato il test della classe MovieBean.java
- Verrà effettuato il test della classe ReviewBean.java

### 2.2. Approccio

L'approccio utilizzato per eseguire il test di unità è stato del tipo black-box, in quanto l'attenzione è stata posta sul comportamento I/O del sistema. I test si sono basati, quindi, solo su ciò che il sistema poteva mostrare, senza conoscere l'implementazione delle classi stesse. I Bean sono stati esercitati tenendo conto anche del fatto che i possibili dati di input e output sono stati opportunamente suddivisi in categorie rappresentanti stati validi o non validi per gli stessi; di conseguenza, durante il test ci si è accertati anche della correttezza delle variabili usate. Avendo verificato il corretto funzionamento di queste classi, è possibile procedere al livello successivo; il framework di supporto utilizzato è JUnit.

#### 2.3. Identificazione del test

#### **UTC\_AL:** UserBean.java

| Nome Classe       | UserBean.java    |
|-------------------|------------------|
| ID Unit Test Case | Metodo           |
| UTC_UB_1          | testGetName()    |
| UTC_UB_2          | testSetName()    |
| UTC_UB_3          | testGetSurname() |
| UTC_UB_4          | testSetSurname() |
| UTC_UB_5          | testGetMail()    |
| UTC_UB_6          | testSetMail()    |
| UTC_UB_7          | testGetCart()    |
| UTC_UB_8          | testSetCart()    |
| UTC_UB_9          | testGetAdmin()   |
| UTC_UB_10         | testSetAdmin()   |

### **UTC\_AMM:** Amministratore.java

| Nome Classe       | ReviewBean.java |
|-------------------|-----------------|
| ID Unit Test Case | Metodo          |
| UTC_RB_1          | testGetId()     |
| UTC_RB_2          | testSetId()     |
| UTC_RB_3          | testGetName()   |
| UTC_RB_4          | testSetName()   |
| UTC_RB_5          | testGetMail()   |
| UTC_RB_6          | testSetMail()   |
| UTC_RB_7          | testGetTitle()  |
| UTC_RB_8          | testSetTitle()  |
| UTC_RB_9          | testGetReview() |
| UTC_RB_10         | testSetReview() |
| UTC_RB_11         | testGetDate()   |
| UTC_RB_12         | testSetDate()   |

## UTC\_ART: MovieBean.java

| Nome Classe       | MovieBean.java      |
|-------------------|---------------------|
| ID Unit Test Case | Metodo              |
| UTC_MB_1          | testGetId()         |
| UTC_MB_2          | testSetId()         |
| UTC_MB_3          | testGetTitle()      |
| UTC_MB_4          | testSetTitle()      |
| UTC_MB_5          | testGetDirector()   |
| UTC_MB_6          | testSetDirector()   |
| UTC_MB_7          | testGetYear()       |
| UTC_MB_8          | testSetYear()       |
| UTC_MB_9          | testGetLangs()      |
| UTC_MB_10         | testSetLangs()      |
| UTC_MB_11         | testGetPrice()      |
| UTC_MB_12         | testSetPrice()      |
| UTC_MB_13         | testGetCategories() |
| UTC_MB_14         | testSetCategories() |
| UTC_MB_15         | testGetImgpath()    |
| UTC_MB_16         | testSetImgpath()    |

| UTC_MB_17 | testGetFramepath()   |
|-----------|----------------------|
| UTC_MB_18 | testSetFramePath()   |
| UTC_MB_19 | testGetDescription() |
| UTC_MB_20 | testSetDescription() |

#### 2.4. Criteri di successo/ fallimento

Nel caso in cui siano stati riscontrati errori durante la fase di testing di unità, si è proceduto con la correzione dei fault intervenendo direttamente sulle porzioni di codice che generano il problema. Si è iterata, dunque, nuovamente la fase di testing per verificare che la modifica non avesse impattato su altre componenti del sistema, ma, al contrario, ne avesse migliorato il comportamento.

#### 2.5. Prodotti del test

Si faccia riferimento al documento 7Movies\_UTR\_v\_1.0.doc.

#### 3. GENERALI

#### 3.1. Glossario

#### Definizioni:

- **Bean**: componente software riusabile, usata per rappresentare le informazioni di interesse.
- ODD: Documento che riporta e analizza gli oggetti che compongono il sistema analizzando le componenti a più basso livello, riportandole così come saranno implementate.
- RAD: Documento di Raccolta e analisi dei Requisiti che contiene l'elenco dei requisiti funzionali e non funzionali individuati in fase di individuazione dei stessi e la loro analisi sotto forma di scenari e casi d'uso. I mock-up mostrano una possibile implementazione dell'interfaccia del sistema.
- **SDD:** Documento che riporta la progettazione del sistema come risultato di una prima fase di modellazione: contiene una suddivisione ad alto livello del sistema nei sottosistemi che lo comporranno.
- TCS: Documento che specifica i casi di test in tutti i loro dettagli.
- TP: Documento che descrive il piano di testing adottato nel progetto e la definizione dei casi di test.

#### Acronimi:

- ODD: Object Design Document;
- RAD: Requirement Analysis Document;
- SDD: System Design Document;
- SPMP: Software Project Management Plan;
- TCS: Test case specification;
- TP: Test Plan;