

Università degli Studi di Salerno
Corso di Ingegneria del Software

CERICinema
Integration Test Design (ITD)
Versione 1.0



Partecipanti:

| Nome | Matricola |
|-------------------|------------|
| Edilio Massaro | 0512103218 |
| Ildo Tiberio | 0512103134 |
| Raffaele Squeglia | 0512103122 |
| Chiara Bufalino | 0512102894 |
| | |
| | |

| | |
|-------------|--|
| Scritto da: | |
|-------------|--|

Revision History

[illegible]

Indice

| | | |
|------|---|---|
| 1. | INTRODUZIONE | 4 |
| 1.1. | <i>Identificativo del documento</i> | 4 |
| 1.2. | <i>Scopo</i> | 4 |
| 2. | DETTAGLI PER IL DESIGN DEL TEST DI INTEGRAZIONE | 5 |
| 2.1. | <i>Funzionalità da testare</i> | 5 |
| 2.2. | <i>Approccio</i> | 5 |
| 2.3. | <i>Identificazione del test</i> | 5 |
| 2.4. | <i>Criteri di successo/fallimento</i> | 6 |
| 2.5. | <i>Prodotto del test</i> | 6 |
| 3. | GENERALI | 7 |
| 3.1. | <i>Glossario</i> | 7 |

1. INTRODUZIONE

Lo scopo di questo documento è quello di pianificare il design del test di integrazione del software CERiCinema: 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, i nomi di chi ha partecipato alla stesura e alla versione del suddetto; tali informazioni sono riportate nel frontespizio.

1.2. *Scopo*

Lo scopo del documento è quello di fornire una presentazione dettagliata della pianificazione del design per il test di integrazione per il software CERiCinema. Nel test si sono integrate le varie componenti del sistema e si è verificato che il software risultasse funzionante.

1.3. *Riferimenti*

- RAD_CeriCinema.docx
- SDD_CERiCinema.docx
- ODD_CERiCinema.docx
- TP_CERiCinema.docx
- TCS_CERiCinema.docx

2. DETTAGLI PER IL DESIGN DEL TEST DI INTEGRAZIONE

In questa sezione del documento verranno forniti i dettagli relativi alle funzionalità utilizzate per eseguire il test di integrazione.

2.1. *Funzionalità da testare*

I sottosistemi che sono stati sottoposti al test di integrazione in questa fase sono i seguenti :

- Verrà effettuato il test della classe DBConnector.java

2.2. *Approccio*

L'approccio utilizzato per eseguire il test di integrazione è stato di tipo Bottom-up (dal basso verso l'alto). Ciò comporta il test e l'integrazione delle singole componenti partendo dal livello più basso, ovvero lo Storage, per poi proseguire con l'Application Layer. Sono stati testati in una prima fase i bean dei vari packages e, verificato il loro corretto funzionamento, si è passati al livello successivo testando i manager, grazie all'utilizzo di opportuni driver e del framework JUnit. Pertanto, il testing di questa fase è vincolato ad un database.

2.3. *Identificazione del test*

ITC_DBC: DBConnector.java

| Nome Classe | DBConnector.java |
|-------------------|-------------------------|
| ID Unit Test Case | |
| ITC_ DB _1 | testUtenti () |
| ITC_ DB _2 | testPrenotazioni () |
| ITC_ DB _3 | testProiezioni() |
| ITC_ DB _3 | testFilm() |

2.4. *Criteri di successo/fallimento*

Il test della classe non ha causato failure e non sono stati riscontrati problemi, di conseguenza non è stato necessario intervenire nelle porzioni di codice dedicate.

2.5. *Prodotto del test*

Si faccia riferimento al documento ITR_CERICinema.docx.

3. GENERALI

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;
- **TCS:** Test case specification;
- **TP:** Test Plan;