
Secondo Appello - Prova di Livello Base-Intermedio

Cognome e Nome: _____ Matricola: _____

Tempo a disposizione: 3 ore

È necessario scrivere un'applicazione che simula il frammento di un sistema informativo per la gestione di un archivio di film. Ogni film è descritto da (a) nome univoco (b) la data di uscita (c) il regista (d) il genere. Ad esempio un possibile film è {" Oppenheimer", 23 Agosto 2023, "Christopher Nolan", "drammatico"}.

Nel sistema sono inoltre catalogati una serie di attori. Ogni attore ha un (a) nome e cognome (b) nazionalità (c) anno di nascita. Ad esempio un possibile attore è {" Cillian Murphy", "Irlanda", 1976}.

Nel sistema bisogna tener traccia degli attori presenti in ciascun film. Ad esempio, Cillian Murphy è presente nel film Oppenheimer.

L'applicazione deve consentire di svolgere i seguenti casi d'uso:

"Utente carica un archivio"

- l'utente carica l'archivio *NOTA: per velocizzare le operazioni, si suggerisce di utilizzare un "mock object" per simulare il funzionamento del DAO che carica i dati dal disco*

"Utente cerca film"

- l'utente sceglie la data di uscita del film e un criterio, che può essere "data crescente" o "data decrescente". *NOTA: per la scelta del criterio, utilizzare JRadioButton*
- il sistema mostra in una tabella la lista di tutti i film usciti prima della data scelta dall'utente

"Utente inserisce nuovo attore"

- l'utente seleziona un film dai risultati
- il sistema mostra, in un nuovo pannello, i dati del film
- il sistema calcola e mostra inoltre quanti attori ci sono per ogni nazionalità. I dati saranno mostrati in una tabella ordinata per numero di attori decrescente. Un esempio di tabella è la seguente

Nazione	Numero di attori
U.S.A.	4 attori
Irlanda	1 attore

- l'utente può decidere di inserire i dati di un nuovo attore, specificando tutti i dati. Dopo aver effettuato l'inserimento, il sistema aggiorna i dati della tabella precedente

Scenario alternativo

- dati scorretti: il sistema mostra un messaggio di errore all'utente

"Utente verifica archivio"

- selezionando una voce di menu, l'utente verifica se in **tutto il sistema** esistono attori duplicati. Un attore è duplicato se ha lo stesso nome e cognome e anno di nascita
- il sistema mostra un messaggio con il risultato calcolato al passo precedente

"Utente ricerca attori frequenti" – Per la prova di livello intermedio

- il sistema calcola, per ogni regista, l'attore più frequente presente nei suoi film
- se c'è un attore che è il più frequente per più di un regista, il sistema mostra il messaggio del tipo "Cristian Bale è l'attore preferito da Christopher Nolan e da Tim Burton"
- **Nota:** è necessario sviluppare i relativi test di regressione

"Utente carica un archivio JSON" – Per la prova di livello intermedio

- Modificare il caso d'uso "Utente carica un archivio" per permettere all'utente di caricare l'archivio da un file json, facendolo selezionare dal disco
- **Nota:** per inizializzare il file si consiglia di scrivere un test per salvare il contenuto dell'archivio mock in un file json

Sviluppare l'applicazione che implementa i casi d'uso elencati, seguendo il processo di sviluppo descritto a lezione, e in particolare le seguenti operazioni:

- costruire e documentare il modello concettuale dell'applicazione
- sviluppare il frammento di applicazione che implementa i casi d'uso descritti
- sviluppare l'interfaccia grafica in **Java Swing** utilizzando l'**architettura MVC** presentata a lezione, rispettando i requisiti per l'accessibilità e l'usabilità.
- sviluppare i test di regressione utilizzando la libreria JUnit
- per effettuare operazioni di logging utilizzare le librerie SLF4J e LogBack
- utilizzare **Gradle** come sistema di costruzione del codice

Per le librerie, se necessario, utilizzare le seguenti dipendenze

- org.slf4j:slf4j-api:2.0.1
- ch.qos.logback:logback-classic:1.3.1
- ch.qos.logback:logback-core:1.3.1
- org.junit.jupiter:junit-jupiter:5.8.2

Per utilizzare Lombok, applicare il seguente plugin gradle

- id "io.freefair.lombok" version "6.5.1"

E' possibile trovare il materiale necessario nelle seguenti cartelle

- materiale didattico: **c:\documentazione\POO1 e POO2**
- file di esempio di logback.xml: **c:\documentazione\POO1 - materiale software aggiuntivo\java**