**Modifiche e note**

1. La classe Gestore è stata sostituita dalla classe SQLDatabase
2. La classe Cinema è stata eliminata in quanto poco utilizzata

* Nel caso d’uso “Inserire un film” è di conseguenza omessa la selezione del cinema, supponendo quindi di organizzare la programmazione dei film relativo ad un solo cinema del circuito

1. Nel scenario alternativo “La sala è già occupata ”del caso d’uso “Modificare una programmazione di film” è stata omessa per semplificazione l’opzione dello scambio delle due programmazioni
2. Nel caso d’uso “Ricercare un film” oltre la scheda è mostrato l’elenco completo della programmazione relativa al film cercato
3. Nel casi d’uso “Modificare uno sconto/abbonamento” sono stati aggiunti come scenari alternativi ulteriori controlli sulla correttezza dei nuovi dati
4. Ogni volta che si effettua un nuovo test tramite Test.java è creato un file test.db, quando si effettuano nuovi test all’interno di Test.java occorre eliminare la vecchia versione di test.db
5. Nel caso d’uso “Inserire un film” è aggiunto come scenario alternativo il controllo sulla sala (se è eventualmente già occupata da un’altra proiezione)
6. Siccome non è possibile l’interazione utente nella fase di testing, sono state ridefinite le funzioni di update di programmazione/sconto/abbonamento, in particolare ogni update è suddiviso in più metodi e ciascun metodo è dedicato al campo di programmazione/sconto/abbonamento che si vuole modificare:

* updateProgrammazione(p) si è suddiviso in updateDataProgrammazione(p,data) e updateDataProgrammazione(p,sala), dove data e sala sono i nuovi campi che l’utente vuole immettere
* updateSconto(s) si è suddiviso in updateNomeSconto(s,nome), updatePrezzoSconto(s,prezzo), updateBeneficiariSconto(s,beneficiari) e updateScadenzaSconto(s,scadenza), dove nome, prezzo, beneficiari e scadenza sono i nuovi campi che l’utente vuole immettere
* updateAbbonamento(a) si è suddiviso in updateNomeAbbonamento(a,nome), updatePrezzoAbbonamento(a,prezzo), updateBeneficiariAbbonamento(a,beneficiari) e updateScadenzaAbbonamento(a,scadenza), dove nome, prezzo, beneficiari e scadenza sono i nuovi campi che l’utente vuole immettere

1. Nella fase di testing non è garantito il 100% di copertura nella classe SQLDatabase per i seguenti motivi:

* Nella funzione create() le tabelle sono effettivamente create solo se il file test.db è creato per la prima volta, altrimenti tale fase è saltata onde evitare sovrascrizione delle tabelle
* Analogamente nella funzione insertFilm() il film è effettivamente creato solo se non è stato inserito precedentemente, altrimenti tale fase è saltata
* Le funzioni di update e i relativi scenari alternativi sono coperti in base alla natura degli input