



Cinema Multisala

Simone Parente Martone Mario Penna
N86004297 N86003308
19/10/2022

Indice

1	Requisiti identificati	3
1.1	Analisi dei requisiti	4
2	Progettazione concettuale	5
2.1	Class Diagram	5
2.2	Analisi della ristrutturazione del Class Diagram	5
2.2.1	Analisi delle ridondanze	6
2.2.2	Analisi degli identificativi	6
2.2.3	Rimozione degli attributi multipli	7
2.2.4	Rimozione degli attributi composti	7
2.2.5	Partizione/Accorpamento delle associazioni	7
2.2.6	Rimozione delle gerarchie	7
2.3	Class Diagram ristrutturato	7
2.4	Dizionario delle classi	8
2.5	Dizionario delle associazioni	8
3	Schema logico	9

Capitolo 1

Requisiti identificati

Si provvederà alla progettazione ed allo sviluppo di una base di dati dedicata alla gestione di un Cinema Multisala, nello specifico:

Gestione di Proiezioni di Film, di Spot pubblicitari e Anteprime di Film all'interno di Sale.

1.1 Analisi dei requisiti

Sono state individuate le seguenti entità:

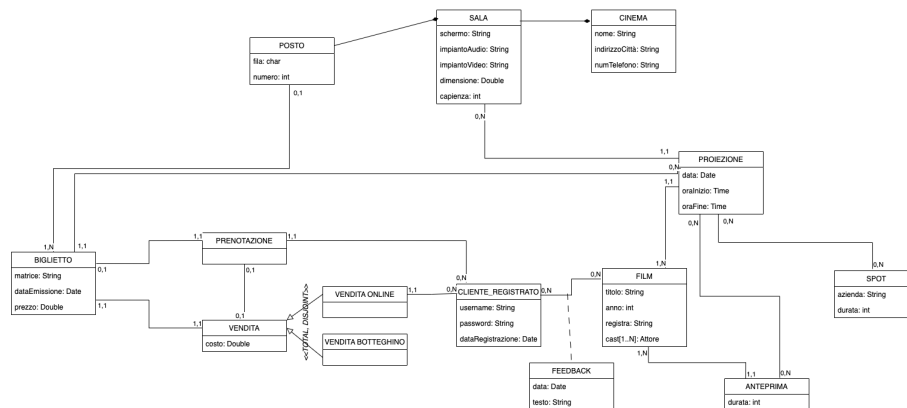
Cinema	Sede di un cinema
Sala	Caratteristiche tecniche di una sala
Posto	Entità atta a identificare univocamente una poltrona
Proiezione	Data e ora di una proiezione
Film	Entità che descrive le generalità di un film
Anteprima	Trailer di un film.
Spot	Entità che identifica univocamente messaggi pubblicitari da parte di aziende
Biglietto	Entità che si occupa della vendita di un posto in sala
Vendita	Generalizzazione di una vendita online e una vendita al botteghino
Vendita online	Specializzazione di una vendita
Vendita botteghino	Specializzazione di una vendita
Prenotazione	Entità che identifica un biglietto prenotato ma non ancora pagato (?)
Cliente	Entità atta a identificare univocamente una persona che ha acquistato un biglietto online
Feedback	Recensione di un film da parte di un cliente

Tabella 1.1: Entità

Capitolo 2

Progettazione concettuale

2.1 Class Diagram



2.2 Analisi della ristrutturazione del Class Diagram

In questa fase ci occupiamo di rendere il Class Diagram idoneo alla traduzione in schemi relazionali, in particolare verranno rimosse:

- Le Generalizzazioni e Specializzazioni
- Gli attributi multipli
- Gli attributi strutturati

2.2.1 Analisi delle ridondanze

L'unica ridondanza che abbiamo ritenuto fosse il caso di rimuovere è l'entità **Vendita** che ha come unico attributo *costo*, anche presente nell'entità **Biglietto**.

2.2.2 Analisi degli identificativi

Nell'analisi degli identificativi provvederemo a scegliere uno o più attributi che garantiranno il rispetto dell'integrità referenziale per ogni tupla.

In particolare notiamo che:

- **Cinema**: si è scelto di aggiungere un attributo ID_Cinema e renderlo chiave primaria.
- **Sala**: è già presente un attributo Numero che rappresenta una potenziale chiave primaria insieme alla chiave esterna ID_Cinema.
- **Posto**: sono già presenti degli attributi Fila e Numero che insieme alle chiavi esterne *Cinema* e *Sala* permettono di identificare univocamente ogni poltrona.¹
- **Proiezione**: si è scelto di aggiungere un attributo ID_Proiezione così da non dover utilizzare una chiave primaria composta da tutti gli attributi dell'entità.
- **Film**: si è scelto di aggiungere un attributo ID_Film e renderlo chiave primaria.
- **Trailer**: si è scelto di aggiungere un attributo ID_Trailer per identificare univocamente i singoli trailer.
- **Spot**: abbiamo ritenuto che il nome di un'azienda non fosse abbastanza per identificare una pubblicità, è stato quindi aggiunto un attributo ID_Spot ed è stato reso chiave primaria.
- **Biglietto**: si è scelto di modificare il nome dell'attributo *matrice* e farlo diventare ID_Biglietto e di rendere lo stesso chiave primaria.
- **Cliente**: è già presente un attributo Username che rappresenta una potenziale chiave primaria.
- **Recensione**: è stato aggiunto un attributo ID_Recensione per identificare univocamente le recensioni.

¹Sarebbe utile creare un attributo *ID_Poltrona*?

2.2.3 Rimozione degli attributi multipli

In questa sezione ci occuperemo di rimuovere eventuali attributi multivalore:

In questo Class Diagram è presente un solo attributo multivalore:

- *Attori*, all'interno dell'entità **Film** che abbiamo provveduto a trasformare in una entità a sé stante avente come attributi Nome, Cognome e Data di nascita.

2.2.4 Rimozione degli attributi composti

In questo Class Diagram non sono presenti attributi composti.

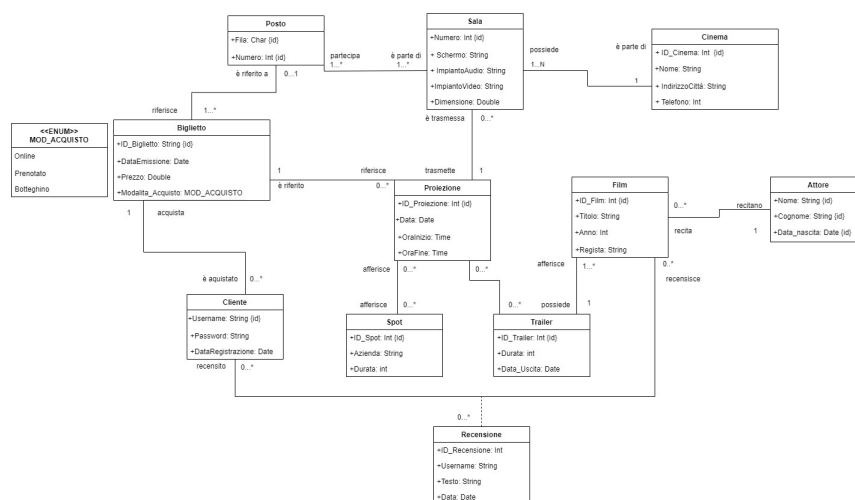
2.2.5 Partizione/Accorpamento delle associazioni

Non si è ritenuto necessario effettuare alcun accorpamento di entità.

2.2.6 Rimozione delle gerarchie

L'unica generalizzazione presente era **Vendita**, che aveva come entità figlie Vendita Online e Vendita Botteghino, che sono state rimosse nel punto 2.2.1 e sostituite dall'attributo *Modalita_Acquisto* di tipo MOD_ACQUISTO inserito come enumerazione nell'entità **Biglietto**.

2.3 Class Diagram ristrutturato



2.4 Dizionario delle classi

2.5 Dizionario delle associazioni

Capitolo 3

Schema logico

Cinema: ID_Cinema - Nome - indirizzoCittà - NumTelefono

Sala: ID_Sala - Numero - Schermo - ImpiantoAudio - ImpiantoVideo - Dimensione - Cinema

Cinema \hookrightarrow Cinema(ID_Cinema)

Posto: Fila - Numero - Sala

Sala \hookrightarrow Sala(ID_Sala)

Biglietto: ID_Biglietto - DataEmissione - Prezzo - Modalità_acq - Fila - Numero - Sala - Proiezione - Username

Fila \hookrightarrow Posto(Fila)

Numero \hookrightarrow Posto(Numero)

Sala \hookrightarrow Posto(Numero)

Username \hookrightarrow Cliente(Username)

Cliente: Username - Password - DataRegistrazione

Recensione: ID_Recensione - Data - Testo - Username - Film

Username \hookrightarrow Cliente(Username)

Film \hookrightarrow Film(ID_Film)

Spot: ID_Spot - Azienda - Durata - Proiezione

Proiezione \hookrightarrow Proiezione(ID_Proiezione)

Trailer: ID_Trailer - Durata - DataUscita - Proiezione - Film

Proiezione \hookrightarrow Proiezione(ID_Proiezione)

Film \hookrightarrow Film(ID_Film)

Film: ID_Film - Titolo - Anno - Regista

Attore: Nome - Cognome - DataNascita - Film

Film \hookrightarrow Film(ID_Film)

Proiezione: ID_Proiezione - Data - OraInizio - OraFine - Sala - Film

Sala \hookrightarrow Sala(ID_Sala)

Film \hookrightarrow Film(ID_Film)