

# Basi di Dati, Progetto di Laboratorio

Federico Serra, matricola 898925

Alessandro Nocera, matricola 886732

## Indice

<b>1</b>	<b>Progettazione concettuale</b>	<b>1</b>
1.1	Requisiti iniziali . . . . .	1
1.2	Glossario dei termini . . . . .	2
1.3	Requisiti rivisti e strutturati in gruppi di frasi omogenee . . . . .	3
1.4	Schema E-R + business rules . . . . .	5
1.4.1	Scherma E-R . . . . .	5
1.4.2	Business rules . . . . .	6
<b>2</b>	<b>Progettazione logica</b>	<b>7</b>
2.1	Tavola dei volumi . . . . .	7
2.2	Tavola delle operazioni . . . . .	8
2.3	Ristrutturazione dello schema E-R . . . . .	9
2.3.1	Analisi delle ridondanze . . . . .	9
2.3.2	Eliminazione delle generalizzazioni . . . . .	10
2.3.3	Partizionamento/accorpamento di entità e associazioni . . . . .	12
2.3.4	Scelta degli identificatori principali . . . . .	12
2.3.5	Schema E-R ristrutturato + business rules . . . . .	13
2.3.6	Schema relazionale . . . . .	14
<b>3</b>	<b>Implementazione</b>	<b>19</b>
3.1	DDL di creazione del database . . . . .	19

## 1 Progettazione concettuale

### 1.1 Requisiti iniziali

Si vuole realizzare una base di dati per la gestione di una piattaforma che fornisce informazioni su film in uscita nei cinema e su serie e programmi TV in onda, liberamente ispirato a piattaforme come ComingSoon.

Un numero limitato di utenti della redazione si occupa di aggiornare le informazioni sui contenuti disponibili inserendo i dati di film e serie, comprese le date di uscita e programmazione in sala e in TV. Degli utenti si memorizzano nome utente, password e indirizzo email. Per i redattori si tiene traccia anche della data di inizio collaborazione.

I contenuti hanno una serie di caratteristiche ~~come evidenziato in Figura 1~~, quali: data di uscita, genere, anno, regia, paese, durata, distribuzione, sceneggiatura, fotografia, musiche e produzione. In particolare, a ogni film e serie TV è associata una lista di attori, con l'indicazione del personaggio interpretato. Notare che alcune informazioni tipiche dei film, ad esempio la regia, per le serie TV sono specifiche dei singoli episodi. Attori e registi hanno una propria scheda (~~esempio in Figura 2~~) dove vengono visualizzati foto, nome, cognome, dati anagrafici e biografia. Qui può essere anche mostrata una lista dei contenuti più recenti a cui la persona ha partecipato.

I film vengono proiettati in svariati cinema in tutto il paese. Per ogni cinema, si tiene traccia del suo nome, contatti, e localizzazione (regione, provincia e indirizzo). La proiezione avviene a degli orari specifici in diverse date, in una delle sale disponibili al cinema, e comporta il pagamento di un biglietto il cui prezzo è indicato sul sito.

Programmi e serie TV vanno invece in onda su diverse piattaforme (es. Netflix, Rai Gulp, ...), che possono essere canali TV tradizionali o piattaforme di streaming video. Le serie TV sono organizzate per genere, come i film, es. commedia, drammatico, etc. In più, i serial sono composti da episodi, racchiusi in una o più stagioni, dove ogni episodio ha uno specifico titolo, durata, regista, e sceneggiatore. Anche per i programmi sono memorizzate alcune informazioni, come l'anno di messa in onda, genere, paese, durata e una descrizione testuale.

I visitatori del sito, previa registrazione tramite email e password, possono lasciare un voto (da 1 a 5 stelline) su qualunque contenuto. La media dei voti ricevuti da film e programmi viene mostrata nelle rispettive schede. Inoltre, gli utenti possono cliccare su un link "inserisci tra i preferiti" presente nelle schede dei contenuti, in modo da poter visualizzare in seguito i contenuti salvati. Nel momento della cancellazione di un utente dal sito, i suoi preferiti possono essere rimossi dal database, mentre i voti espressi dal visitatore rimangono in memoria.

## 1.2 Glossario dei termini

<i>Termine</i>	<i>Descrizione</i>	<i>Sinonimi</i>	<i>Collegamento</i>
<b>Piattaforme</b>	Servizi informatici che rendono disponibili agli utenti le stagioni di una serie	Piattaforme di streaming video	Serie
<b>Contenuti</b>	Insieme di produzioni audiovisive (film, serie e programmi)		Film, serie, programmi
<b>Utente</b>	Persona registrata sul sito	Visitatori	Voto
<b>Redattore</b>	Utente che aggiorna informazioni riguardanti i contenuti presenti sul sito	Utenti della redazione	Voto
<b>Serie</b>	Serie TV disponibili sul sito	Serie TV, serial	Stagione, episodio, piattaforma
<b>Stagione</b>	Raggruppamento in episodi di una serie		Episodio, serie
<b>Episodio</b>	Singola unità di una stagione		Stagione, attore, regista
<b>Film</b>	Film presenti sul sito		Cinema, attore, regista
<b>Attore</b>	Colui che recita in un contenuto interpretando un ruolo	Persona	Contenuto
<b>Regista</b>	Colui che si occupa della regia dei contenuti	Regia, persona	Contenuto
<b>Cinema</b>	Luogo in cui è proiettato un film		Film
<b>Voto</b>	Voto numerico (da 1 a 5 stelle) espresso da un utente del sito		Contenuto, utente

### 1.3 Requisiti rivisti e strutturati in gruppi di frasi omogenee

- **Frasi di carattere generale:** Si vuole realizzare una base di dati per la gestione di un sito che fornisce informazioni su film in uscita nei cinema e su serie e programmi TV in onda, liberamente ispirato a piattaforme come ComingSoon.
- **Frasi relative agli utenti:** Gli utenti possono essere di due tipologie: iscritti o redattori. Entrambi si registrano tramite email e password. Possono votare e aggiungere ai preferiti qualunque contenuto. Nel momento della cancellazione di un utente dal sito, i suoi preferiti possono essere rimossi dal database, mentre i voti espressi rimangono in memoria. Per gli utenti si memorizzano nome utente, password e indirizzo email.
- **Frasi relative ai redattori:** I redattori sono utenti che si occupano di aggiornare le informazioni sui contenuti disponibili inserendo i dati dei contenuti, comprese le date di uscita e programmazione nei cinema e nei canali televisivi. Per i redattori si registra anche della data di inizio collaborazione.
- **Frasi relative ai contenuti:** I contenuti possono essere di tre tipi: serie, film e programmi. Per i contenuti rappresentiamo: titolo, data di uscita, genere, paese, durata, distribuzione, produzione e i membri della troupe (attori con relativo personaggio interpretato, regista, sceneggiatori, i direttori della fotografia, i responsabili

delle musiche).

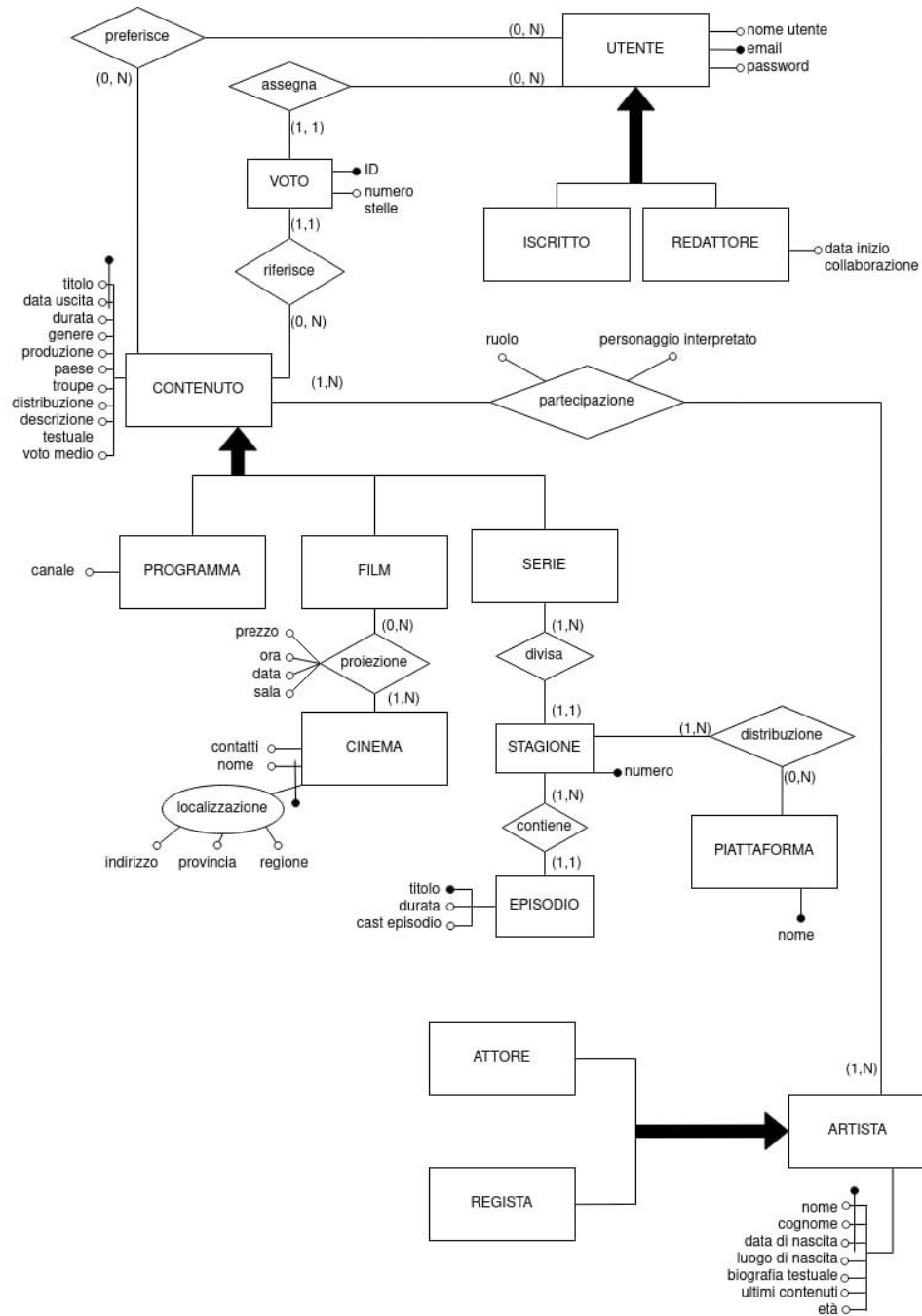
Per i programmi memorizziamo anche il canale dove è disponibile il contenuto.

Per i film memorizziamo invece anche il cinema dove saranno disponibili.

- **Frasi relative alle serie:** Le serie sono contenuti divisi in stagioni. Le stagioni vanno in onda su siti di streaming online e sono divise in episodi. Per ogni episodio si memorizza titolo, durata, regista, sceneggiatore e la lista di attori che vi partecipano.
- **Frasi relative ai cinema:** Per i cinema rappresentiamo nome, contatti, e localizzazione (regione, provincia e indirizzo). La proiezione dei film è caratterizzata da: una sala del cinema, una data e un'ora, il prezzo del biglietto.
- **Frasi relative agli artisti:** Gli artisti possono essere attori o registi. Per gli artisti è registrata una scheda personale contenente: foto, nome, cognome, età, data e luogo di nascita, biografia e gli ultimi contenuti a cui ha partecipato.

## 1.4 Schema E-R + business rules

### 1.4.1 Scherma E-R



### 1.4.2 Business rules

- Quando un utente viene cancellato i voti espressi rimangono, dati personali e preferenze vengono rimossi.
- Il valore del voto è compreso tra 1 e 5 stelle.
- L'attributo voto medio di un contenuto indica la media dei voti assegnati dagli utenti.
- Per il singolo artista viene mantenuta una lista dei contenuti a cui ha partecipato.
- I redattori aggiornano le informazioni relative ai contenuti.
- L'attributo ruolo dell'associazione partecipazione può assumere i valori regista o attore. Nel primo caso l'attributo personaggio interpretato avrà valore NULL, nel secondo invece avrà come valore il nome del personaggio interpretato dall'attore all'interno del contenuto.

## 2 Progettazione logica

### 2.1 Tavola dei volumi

<i>Concetto</i>	<i>Tipo</i>	<i>Volume</i>	<i>Motivazione</i>
<b>Utente</b>	Entità	50.000	Approssimazione della somma tra Iscritti e Redattori
<b>Iscritto</b>	Entità	50.000	Ci si basa sulla piattaforma italiana mymovies che contava nel 2018 una media di 500.000 utenti unici giornalieri, si suppone che il 10% sia registrato.
<b>Redattore</b>	Entità	10	Mymovies conta 5 content manager, più altri collaboratori. Si sceglie 10 come numero di profili abilitati alla modifica calcolandone uno per ogni content manager più altri 5 divisi tra i collaboratori.
<b>Preferisce</b>	associazione	250.000	Si considera una media di 5 film preferiti per utente
<b>Assegna</b>	associazione	5	Media di 5 voti per utente ottenuta dividendo i voti totali per gli utenti
<b>Voti</b>	Entità	250.000	Da mymovies si evince che ci sono circa 200.00 recensioni totali, si considera un voto per recensione più un voto per utente che non lascia il commento
<b>Voto contenuto</b>	associazione	500	Si considera che il 1% di utenti vota un film.
<b>Contenuti</b>	Entità	40.000	Basandosi su dati di un articolo su mymovies
<b>Partecipazione</b>	associazione	20	Si stima un media di partecipazione a 20 contenuti per artista
<b>Programma</b>	Entità	500	Approssimazione dati mymovies
<b>Film</b>	Entità	10.000	Approssimazione dati mymovies
<b>Proiezione</b>	associazione	420.000	Si stima che in ogni cinema ci siano 4 spettacoli al giorno, moltiplicati per i 30 giorni (un film rimane in programmazione un mese), moltiplicati per i cinema in Italia
<b>Cinema</b>	Entità	3.500	Numero di cinema presenti su mymovies
<b>Serie</b>	Entità	1.750	Approssimazione basata sul fatto che sono 600 le serie presenti su Netflix, considerando che non è presente 1 serie su 3.
<b>Divisa</b>	associazione	3.500	Stesso valore di Stagione.
<b>Stagione</b>	Entità	3.500	Media di 2 stagioni a serie.
<b>Contiene</b>	associazione	29.500	Stesso valore di Episodio.
<b>Distribuzione</b>	associazione	600	Si stima che per ogni piattaforma è presente il circa il 30% delle serie.
<b>Episodio</b>	Entità	29.500	Media di 8-9 episodi a stagione
<b>Piattaforma</b>	Entità	10	Si considerano le 10 principali piattaforme in Italia
<b>Attore</b>	Entità	3.000	Gli artisti rimanenti togliendo i registi
<b>Regista</b>	Entità	2.000	Si considera una media di 20 contenuti girati per regista
<b>Artisti</b>	Entità	5.000	Dati approssimati ComingSoon

## 2.2 Tavola delle operazioni

<i>Operazione</i>	<i>Tipo</i>	<i>Frequenza</i>	<i>Motivazione</i>
<b>Registrazione utente</b>	Interattivo	10/giorno	Si considera il numero di utenti totali attuali dopo 15 anni di attività
<b>Cancellazione utente</b>	Interattivo	1/giorno	Si considera che l'eliminazione avvenga meno raramente della registrazione, si ottiene così una media di +9 utenti/giorno coerente con i valori riportati nella tavola dei volumi
<b>Modifica dati utente</b>	Interattivo	5/mese	Operazione fatta raramente
<b>Assegnazione di un voto a contenuto</b>	Interattivo	45/giorno	Si considera il numero di voti dopo 15 anni di attività
<b>Aggiornamento voto medio di un contenuto</b>	Batch	45/giorno	Uguale al numero di assegnazioni di voto
<b>Aggiunta di un contenuto ai preferiti</b>	Interattivo	45/giorno	Numero totale dei preferiti suddivisi in 15 anni di attività
<b>Eliminazione di un contenuto tra i preferiti</b>	Interattivo	1/giorno	Molto inferiore al numero di aggiunta dei preferiti
<b>Aggiunta di un contenuto</b>	Interattivo	3/giorno	Si considerano film e serie presenti sulla piattaforma dopo 15 anni di attività
<b>Modifica contenuto</b>	Interattivo	1/mese	Si suppone di dover modificare un contenuto molto raramente
<b>Aggiornamento cinema di proiezione del film</b>	Interattivo	90/mese	1/mese per i film attualmente al cinema, contando una media di 1000 film prodotti all'anno
<b>Aggiunta stagione alla serie</b>	Interattivo	1/mese	Si considera la media uscita di una nuova serie di 1 al mese
<b>Aggiunta artista</b>	Interattivo	1/mese	Si considera che l'aggiunta della scheda di un artista avvenga raramente, più raramente della modifica
<b>Modifica/aggiornamento artista</b>	Interattivo	3/giorno	Legato all'aggiunta di un contenuto
<b>Visualizzazione preferiti</b>	Interattivo	800/giorno	Si considera che un utente guarda i propri preferiti raramente, una volta ogni due mesi
<b>Visualizzazione contenuto</b>	Interattivo	500.000/giorno	Ci si basa sui dati di mymovies, è l'operazione in assoluto più frequente
<b>Visualizzazione artista</b>	Interattivo	1.000/giorno	Minore della visualizzazione del contenuto, si stima $\frac{1}{5}$ delle visualizzazioni del contenuto

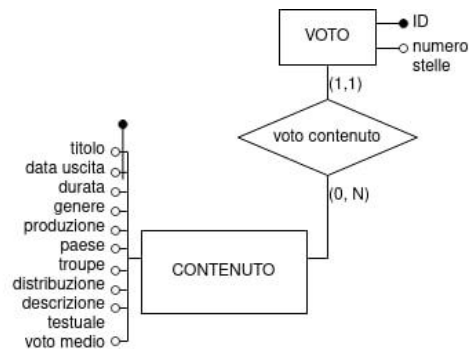


## 2.3 Ristrutturazione dello schema E-R

### 2.3.1 Analisi delle ridondanze

1. Attributo “età” di “artista” (attributo derivabile): l’attributo “età” è derivabile considerando l’anno di nascita e la data odierna. Il mantenimento di questo attributo comporta l’aggiornamento costante di un dato (“età”) secondo la data di nascita dell’artista. Per il precedente motivo si è scelto di eliminare la ridondanza, diminuendo gli aggiornamenti dei dati relativi all’artista.
2. Attributo “Voto medio” di “contenuto” (attributo derivabile da entità e conteggio): l’attributo “voto medio” è derivabile dal conteggio delle occorrenze dell’entità “voto” facente riferimento ad un dato “contenuto”. In questa somma, se si tiene anche conto del valore dei punteggi (il numero di stelle), si può facilmente derivare che  $\frac{\text{totalepunteggi}}{\text{cardinalitàvoto}} = \text{votomedio}$ .

Si è scelto di analizzare la seconda ridondanza in quanto ritenuta più significativa.



#### Operazione 1 - assegnazione di un voto a un contenuto -

*Accessi con ridondanza:*

Concetto	Costrutto	Accessi	Tipo
<b>Voto</b>	Entità	1	Scrittura
<b>Voto</b>	Entità	1	Lettura
<b>Voto contenuto</b>	associazione	1	Scrittura
<b>Contenuto</b>	Entità	1	Scrittura

Il costo è di  $(45 \times 2) + 45 + (45 \times 2) + (45 \times 2) = 315$  accessi al giorno, contando gli accessi in scrittura come doppi.

*Accessi senza ridondanza:*

Concetto	Costrutto	Accessi	Tipo
<b>Voto</b>	Entità	1	Scrittura
<b>Voto contenuto</b>	Entità	1	Scrittura

Il costo è di  $(45 \times 2) + (45 \times 2) = 180$  accessi al giorno, contando gli accessi in scrittura come doppi.

## Operazione 2 - visualizzazione contenuto (include il voto medio) -

Accessi con ridondanza:

Concetto	Costrutto	Accessi	Tipo
<b>Contenuto</b>	Entità	1	Lettura

Il costo è di 500000 accessi al giorno.

Accessi senza ridondanza:

Concetto	Costrutto	Accessi	Tipo
<b>Voto</b>	Entità	1	Lettura
<b>Voto contenuto</b>	associazione	1	Lettura
<b>Contenuto</b>	Entità	1	Lettura

Il costo è di  $500000 + 500000 + 500000 = 1500000$  accessi al giorno.

## Operazione 1 + operazione 2 - costi totali -

Con ridondanze (operazione 1 + operazione 2):

Costo totale in numero di accessi  $315 + 500000 = 500315$ .

Senza ridondanze (operazione 1 + operazione 2):

Costo totale in numero di accessi  $180 + 1500000 = 1500180$ .

Costi aggiuntivi in termini di spazio:

Ipotesi: si utilizzano 4byte per memorizzare il valore del voto medio.

Spazio totale necessario:  $4 \times 40000 = 160000 = 160Kbyte$

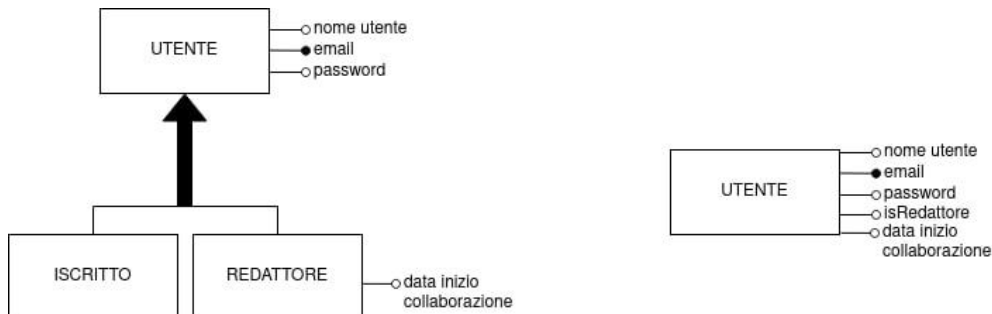
	Con ridondanza	Senza ridondanza
Numero accessi	500315	1500180
Spazio aggiuntivo	160Kbyte	0Kbyte

### Decisione:

Considerata la differenza di circa 1000000 di accessi e lo spreco di memoria non ingente, si è scelto di lasciare la ridondanza, essendo anche l'operazione 2 la più frequente sulla base di dati.

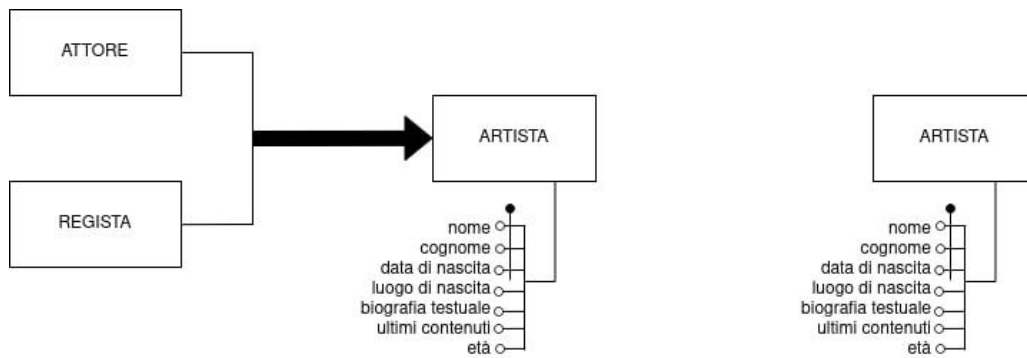
## 2.3.2 Eliminazione delle generalizzazioni

### 1. Utente - Iscritto, Redattore (generalizzazione totale ed esclusiva)



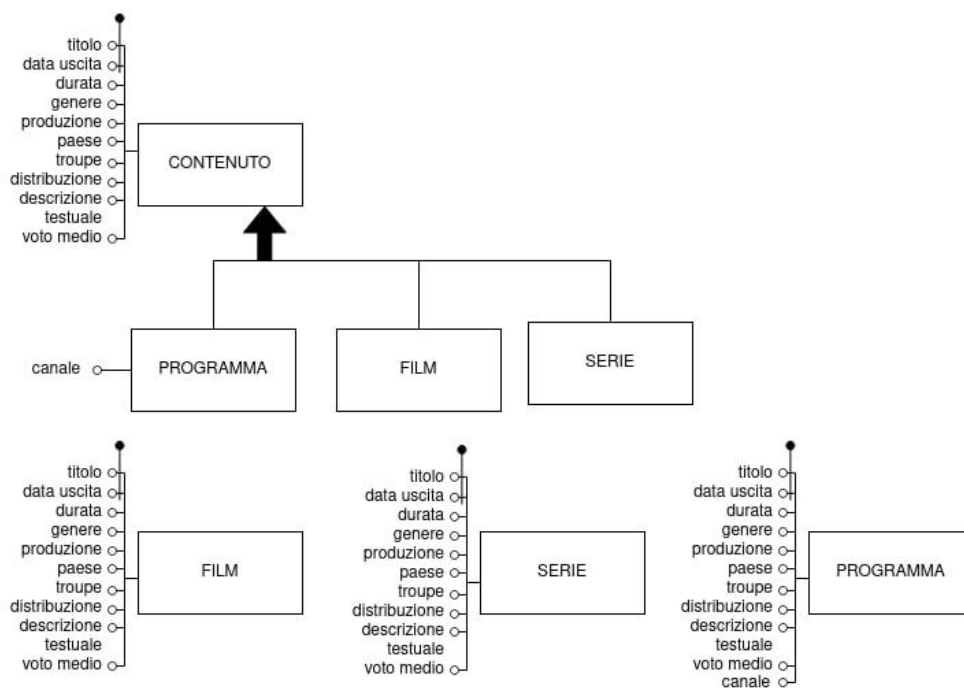
Motivazione: si è scelto di accorpare le entità figlie della generalizzazione nell'entità padre, semplificando così la base di dati. Si è aggiunto a tale scopo un flag *isRedattore* per indicare se l'utente è un iscritto o un redattore.

## 2. Artista - Attore, Regista (generalizzazione totale ed esclusiva)



Motivazione: si è scelto di accorpare le entità figlie della generalizzazione nell'entità padre, in quanto gli accessi alle tabelle sono per la maggior parte contestuali. Se osserviamo la tavola delle operazioni infatti notiamo che la visualizzazione di un artista avviene 1000 volte in un giorno, la visualizzazione di un contenuto (che richiede l'accesso ad entrambe le tabelle) invece occorre 500000 volte nello stesso tempo. Per questo accorpamento non è stata necessaria l'aggiunta di attributi, in quanto la differenza tra attore e regista viene già espressa dall'attributo ruolo della associazione partecipazione.

## 3. Contenuto - Programma, Film, Serie (generalizzazione totale ed esclusiva)



Motivazione: si è scelto di accorpare la entità padre della generalizzazione nelle entità figlie, in quanto gli accessi alla tabelle figlie sono distinti.

### **2.3.3 Partizionamento/accorpamento di entità e associazioni**

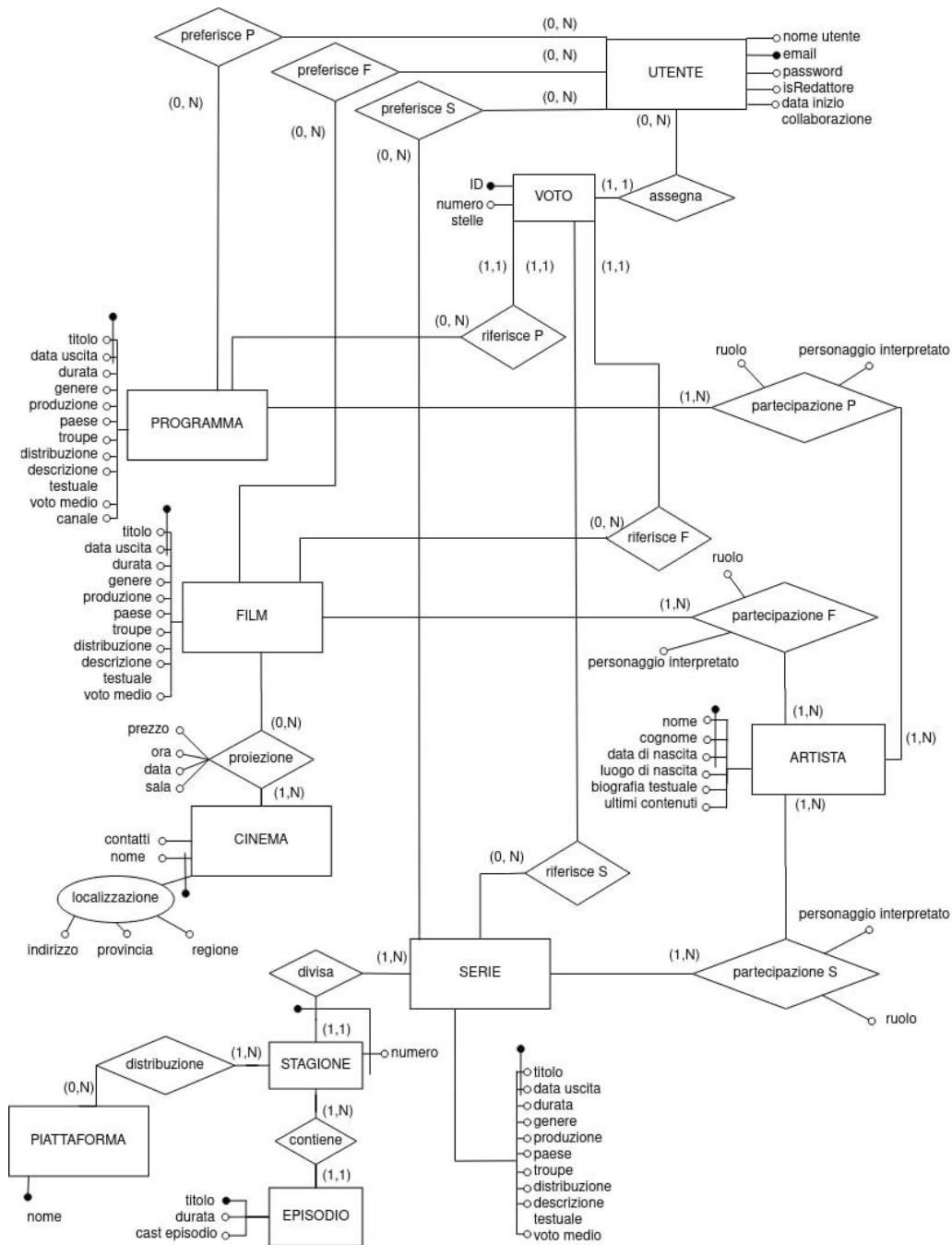
Non si è valutato necessario il partizionamento e/o l'accorpamento di entità e associazioni a fronte delle operazioni previste sulla base di dati, descritte nella tavola delle operazioni.

### **2.3.4 Scelta degli identificatori principali**

L'attributo ID di Voto ha lo scopo di mantenere i voti degli utenti eventualmente rimossi dalla base di dati (necessari per il calcolo del voto medio di film, serie e programmi). Nonostante fosse già presente nello **schema E-R iniziale** si è pensato di spiegarlo per maggiore chiarezza.

La descrizione degli altri identificatori principali non è stata trattata in quanto si ritiene sufficientemente auto-esplicativa a partire dallo schema E-R + business-rules.

### 2.3.5 Schema E-R ristrutturato + business rules



#### Business rules:

- Se un utente ha il flag `isRedattore = TRUE` allora l'attributo `data inizio collaborazione` IS NOT NULL.

- Se viene modificato il flag `isRedattore` a `FALSE` l'attributo `data inizio collaborazione` assume valore `NULL`.
- Se viene modificato il flag `isRedattore` a `TRUE` l'attributo `data inizio collaborazione` assume come valore la data del momento in cui viene modificato il flag (il giorno corrente).
- Il valore del voto è compreso tra 1 e 5 stelle.
- L'attributo voto medio di film indica la media dei voti assegnati dagli utenti.
- L'attributo voto medio di serie indica la media dei voti assegnati dagli utenti.
- L'attributo voto medio di programma indica la media dei voti assegnati dagli utenti.
- Per il singolo artista viene mantenuta una lista dei contenuti a cui ha partecipato.
- Gli utenti con flag `isRedattore` = `TRUE` aggiornano le informazioni relative a: film, serie, programmi.
- L'attributo ruolo dell'associazione partecipazione F può assumere i valori regista o attore. Nel primo caso l'attributo personaggio interpretato avrà valore `NULL`, nel secondo invece avrà come valore il nome del personaggio interpretato dall'attore all'interno del film.
- L'attributo ruolo dell'associazione partecipazione S può assumere i valori regista o attore. Nel primo caso l'attributo personaggio interpretato avrà valore `NULL`, nel secondo invece avrà come valore il nome del personaggio interpretato dall'attore all'interno della serie.
- L'attributo ruolo dell'associazione partecipazione P può assumere i valori regista o attore. Nel primo caso l'attributo personaggio interpretato avrà valore `NULL`, nel secondo invece avrà come valore il nome del personaggio interpretato dall'attore all'interno del programma.

### 2.3.6 Schema relazionale

Utente (Email, Nome utente, Password, isRedattore, Data inizio collaborazione)

Preferisce F (Email Utente, Titolo Film, Data uscita Film)

Preferisce F.Email Utente references Utente.Email

Preferisce F.Titolo Film references Film.Titolo

Preferisce F.Data uscita Film references Film.Data uscita

Preferisce S (Email Utente, Titolo Serie, Data uscita Serie)

Preferisce S.Email Utente references Utente.Email

Preferisce S.Titolo Serie references Serie.Titolo

Preferisce S.Data uscita Serie references Serie.Data uscita

Preferisce P (Email Utente, Titolo Programma, Data uscita Programma)

Preferisce P.Email utente references Utente.Email

Preferisce P.Titolo programma references Programma.Titolo

Preferisce P.Data uscita Programma references Programma.Data uscita

Voto (ID, Numero stelle, Email Utente)

Voto.Email Utente references Utente.Email

Riferisce F (ID voto, Titolo film, Data uscita Film)

Riferisce.ID Voto references Voto.ID

Riferisce F.Titolo Film references Film.Titolo

Riferisce F.Data uscita Film references Film.Data uscita

Riferisce S (ID voto, Titolo serie, Data uscita serie)

Riferisce.ID Voto references Voto.ID

Riferisce F.Titolo Serie references Serie.Titolo

Riferisce F.Data uscita Serie references Serie.Data uscita

Riferisce P (ID voto, Titolo programma, Data uscita programma)

Riferisce.ID Voto references Voto.ID

Riferisce F.Titolo Programma references Programma.Titolo

Riferisce F.Data uscita Programma references Programma.Data uscita

Distribuzione (Nome Piattaforma, Numero Stagione, Titolo Serie, Data uscita Serie)

Distribuzione.Nome Piattaforma references Piattaforma.Nome

Distribuzione.Numero Stagione references Stagione.Numero

Distribuzione.Titolo Serie references Stagione.Titolo Serie

Distribuzione.Data uscita Stagione references Stagione.Data uscita Serie

Film (Titolo, Data uscita, Durata, Genere, Produzione, Paese, Troupe, Distribuzione,  
Descrizione testuale, Voto medio)

Serie (Titolo, Data uscita, Durata, Genere, Produzione, Paese, Troupe, Distribuzione,  
Descrizione testuale, Voto medio)

Programma (Titolo, Data uscita, Durata, Genere, Produzione, Paese, Troupe,  
Distribuzione, Descrizione testuale, Voto medio, Canale)

Proiezione (Titolo Film, Data uscita Film, Nome Cinema, Indirizzo Cinema, Provincia Cinema,  
Regione Cinema, Prezzo, Ora, Data, Sala})

Proiezione.Titolo Film references Film.Titolo

Proiezione.Data uscita Film references Film.Data uscita

Proiezione.Nome Cinema references Cinema.Nome

Proiezione.Indirizzo Cinema references Cinema.Indirizzo

Proiezione.Provincia Cinema references Cinema.Provincia

Proiezione.Regione Cinema references Cinema.Regione

Cinema (Nome, Indirizzo, Provincia, Regione, Contatti)

Artista (Nome, Cognome, Data di nascita, Luogo di nascita, Biografia testuale,  
Ultimi contenuti)

Partecipazione F (Nome Artista, Cognome Artista, Data di nascita Artista, Titolo Film,  
Data uscita Film, Ruolo, Personaggio interpretato)

Partecipazione F.Nome Artista references Artista.Nome



Partecipazione F.Cognome Artista references Artista.Cognome  
Partecipazione F.Data di nascita Artista references Artista.Data di nascita  
Partecipazione F.Titolo Film references Film.Titolo  
Partecipazione F.Data uscita Film references Film.Data uscita

Partecipazione S (Nome Artista, Cognome Artista, Data di nascita Artista, Titolo Serie,  
Data uscita Serie,Ruolo, Personaggio interpretato)

Partecipazione S.Nome Artista references Artista.Nome  
Partecipazione S.Cognome Artista references Artista.Cognome  
Partecipazione S.Data di nascita Artista references Artista.Data di nascita  
Partecipazione S.Titolo Serie references Serie.Titolo  
Partecipazione S.Data uscita Serie references Serie.Data uscita

Partecipazione P (Nome Artista, Cognome Artista, Data di nascita Artista, Titolo Programma,  
Data uscita Programma, Ruolo, Personaggio interpretato)

Partecipazione P.Nome Artista references Artista.Nome  
Partecipazione P.Cognome Artista references Artista.Cognome  
Partecipazione P.Data di nascita Artista references Artista.Data di nascita  
Partecipazione P.Titolo Programma references Programma.Titolo  
Partecipazione P.Data uscita Programma references Programma.Data uscita

Stagione (Numero, Titolo Serie, Data uscita Serie)

Stagione.Titolo uscita Serie references Stagione.Titolo Serie  
Stagione.Data uscita Serie references Stagione.Data uscita

Divisa (Titolo Serie, Data uscita Serie, Numero Stagione)

Divisa.Titolo Serie references Serie.Titolo  
Divisa.Data uscita Serie references Serie.Data uscita  
Divisa.Numero Serie references Stagione.Numero

Contiene (Numero Stagione, Titolo Serie, Data uscita Serie, Titolo Episodio)

Contiene.Numero Stagione references Stagione.Numero

Contiene.Titolo Serie references Stagione.Titolo Serie

Contiene.Data uscita Serie references Stagione.Data uscita Serie

Contiene.Titolo Episodio references Stagione.Titolo

Episodio (Titolo, Durata, Cast episodio)

Piattaforma (Nome)

## 3 Implementazione

### 3.1 DDL di creazione del database

```
create table UTENTE (
    Email varchar(64) NOT NULL CHECK (Email LIKE '%_@_%._%'),
    Nome_utente varchar(32) not null,
    PasswordU varchar(32) not null check(length>PasswordU)>=8),
    isRedattore boolean not null default false,
    Data_inizio_collaborazione date default null,
    constraint UTENTE_P_KEY primary key(Email)
);

create table FILM(
    Titolo varchar(64) not null,
    Data_uscita date not null,
    Durata int CHECK(Durata>0),
    Genere varchar(128),
    Produzione varchar(128),
    Paese varchar(128),
    Troupe varchar(1000),
    Distribuzione varchar(128),
    Descrizione_testuale varchar(1000),
    Voto_medio decimal(2,1),
    constraint FILM_P_KEY primary key (Titolo, Data_uscita)
);

create table SERIE(
    Titolo varchar(64) not null,
    Data_uscita date not null,
    Durata int check(Durata>0),
    Genere varchar(128),
    Produzione varchar(128),
    Paese varchar(128),
    Troupe varchar(1000),
    Distribuzione varchar(128),
    Descrizione_testuale varchar(1000),
    Voto_medio decimal(2,1),
    constraint SERIE_P_KEY primary key (Titolo, Data_uscita)
);

create table STAGIONE(
    Numero int not null,
    Titolo_Serie varchar(128) not null,
    Data_uscita_Serie date not null,
    constraint STAGIONE_P_KEY primary key (Numero, Titolo_Serie, Data_uscita_Serie),
    constraint STAGIONE_F_KEY_SERIE foreign key (Titolo_Serie, Data_uscita_Serie) references SER.
```

```

create table PROGRAMMA(
    Titolo varchar(64) not null,
    Data_uscita date not null,
    Durata int check(Durata>0),
    Genere varchar(128),
    Produzione varchar(128),
    Paese varchar(128),
    Troupe varchar(1000),
    Distribuzione varchar(128),
    Descrizione_testuale varchar(1000),
    Voto_medio decimal(2,1),
    Canale varchar(128),
    constraint PROGRAMMA_P_KEY primary key (Titolo, Data_uscita)
);

create table VOTO (
    Id serial,
    Numero_stelle int check (Numero_stelle >=1 AND Numero_stelle <=5),
    Email_Utente varchar (64),
    constraint VOTO_F_KEY_UTENTE foreign key (Email_Utente)
    references UTENTE(Email),
    constraint VOTO_P_KEY primary key (Id)
);

create table PREFERISCE_F (
    Email_Utente varchar (64),
    Titolo_Film varchar (64),
    Data_uscita_Film date,
    constraint PREFERISCE_F_P_KEY primary key (Email_Utente, Titolo_Film, Data_uscita_Film),
    constraint PREFERISCE_F_KEY_UTENTE foreign key (Email_Utente) references UTENTE(Email),
    CONSTRAINT PREFERISCE_F_KEY_FILM foreign key (Data_uscita_Film, Titolo_Film)references FILM
);

create table PREFERISCE_S (
    Email_Utente varchar (64),
    Titolo_Serie varchar (64),
    Data_uscita_Serie date,
    constraint PREFERISCE_S_P_KEY primary key (Email_Utente, Titolo_Serie, Data_uscita_Serie),
    constraint PREFERISCE_S_KEY_UTENTE foreign key (Email_utente) references UTENTE(Email),
    constraint PREFERISCE_S_KEY_SERIE foreign key (Data_uscita_Serie, Titolo_Serie) references S
);

create table PREFERISCE_P (
    Email_Utente varchar (64),
    Titolo_Programma varchar (64),
    Data_uscita_Programma date,
    constraint PREFERISCE_P_P_KEY primary key (Email_Utente, Titolo_Programma, Data_uscita_Programma),
    constraint PREFERISCE_P_KEY_UTENTE foreign key (Email_utente) references UTENTE(Email),

```

```

        constraint REFERISCE_P_KEY_PROGRAMMA foreign key (Data_uscita_Programma, Titolo_Programma)
    );

create table RIFERISCE_F (
    Id_Voto serial,
    Titolo_Film varchar (64),
    Data_uscita_Film date,
    constraint riferisce_F_P_KEY primary key (Id_Voto, Titolo_Film, Data_uscita_Film),
    constraint riferisce_F_KEY_VOTO foreign key (Id_Voto) references VOTO(Id),
    constraint riferisce_F_KEY_FILM foreign key (Data_uscita_Film, Titolo_Film) references FILM
);

create table RIFERISCE_P (
    Id_Voto serial,
    Titolo_programma varchar (64),
    Data_uscita_programma date,
    constraint riferisce_P_P_KEY primary key (Id_Voto, Titolo_programma, Data_uscita_programma),
    constraint riferisce_P_KEY_VOTO foreign key (Id_Voto) references VOTO(Id),
    constraint riferisce_P_KEY_programma foreign key (Data_uscita_programma, Titolo_programma)
);

create table RIFERISCE_S (
    Id_Voto serial,
    Titolo_serie varchar (64),
    Data_uscita_serie date,
    constraint riferisce_S_P_KEY primary key (Id_Voto, Titolo_serie, Data_uscita_serie),
    constraint riferisce_S_KEY_VOTO foreign key (Id_Voto) references VOTO(Id),
    constraint riferisce_S_KEY_serie foreign key (Data_uscita_serie, Titolo_serie) references s
);

create table CINEMA(
    Nome varchar(128) not null,
    Indirizzo varchar(128) not null,
    Provincia varchar(128) not null,
    Regione varchar(128) not null,
    Contatti varchar(1000),
    constraint CINEMA_P_KEY primary key (Nome, Indirizzo, Provincia, Regione)
);

create table ARTISTA(
    Nome varchar(64) not null,
    Cognome varchar(64) not null,
    Data_di_nascita date not null,
    Luogo_di_nascita varchar(128),
    Biografia_testuale varchar(1000),
    Ultimi_contenuti varchar(1000),
    constraint ARTISTA_P_KEY primary key (Nome, Cognome, Data_di_nascita)
);

```

```

create table EPISODIO(
    Titolo varchar(128) not null,
    Durata int,
    Cast_episodio varchar(1000),
    constraint EPISODIO_P_KEY primary key (Titolo)
);

create table PIATTAFORMA(
    Nome varchar(128) not null,
    constraint PIATTAFORMA_P_KEY primary key (Nome)
);

create table PARTECIPAZIONE_F(
    Nome_Artista varchar(128) not null,
    Cognome_Artista varchar(128) not null,
    Data_di_nascita_Artista date not null,
    Titolo_Film varchar(128) not null,
    Data_uscita_Film date not null,
    Ruolo varchar(64),
    Personaggio_interpretato varchar(64),
    constraint PARTECIPAZIONE_F_P_KEY primary key( Nome_Artista, Cognome_Artista,
    Data_di_nascita_Artista, Titolo_Film, Data_uscita_Film),
    constraint PARTECIPAZIONE_F_F_KEY_ARTISTA foreign key (Nome_Artista,
    Cognome_Artista,Data_di_nascita_Artista) references ARTISTA(Nome,Cognome,
    Data_di_nascita),
    constraint PARTECIPAZIONE_F_F_KEY_FILM foreign key (Titolo_Film,Data_uscita_Film)
references FILM(Titolo, Data_uscita)
);

create table PARTECIPAZIONE_S(
    Nome_Artista varchar(128) not null,
    Cognome_Artista varchar(128) not null,
    Data_di_nascita_Artista date not null,
    Titolo_Serie varchar(128) not null,
    Data_uscita_Serie date not null,
    Ruolo varchar(64),
    Personaggio_interpretato varchar(64),
    constraint PARTECIPAZIONE_S_P_KEY primary key( Nome_Artista, Cognome_Artista,
    Data_di_nascita_Artista, Titolo_Serie, Data_uscita_Serie),
    constraint PARTECIPAZIONE_F_F_KEY_ARTISTA foreign key (Nome_Artista,
    Cognome_Artista,Data_di_nascita_Artista) references ARTISTA(Nome,Cognome,
    Data_di_nascita),
    constraint PARTECIPAZIONE_S_KEY_FILM foreign key (Titolo_Serie, Data_uscita_Serie)
references SERIE(Titolo, Data_uscita)
);

create table PARTECIPAZIONE_P(

```

```

Nome_Artista varchar(128) not null,
Cognome_Artista varchar(128) not null,
Data_di_nascita_Artista date not null,
Titolo_Programma varchar(128) not null,
Data_uscita_Programma date not null,
Ruolo varchar(64),
Personaggio_interpretato varchar(64),
constraint PARTECIPAZIONE_P_P_KEY primary key( Nome_Artista, Cognome_Artista,
Data_di_nascita_Artista, Titolo_Programma, Data_uscita_Programma),
constraint PARTECIPAZIONE_P_F_KEY_ARTISTA foreign key (Nome_Artista,
Cognome_Artista,Data_di_nascita_Artista) references ARTISTA(Nome,Cognome,
Data_di_nascita),
constraint PARTECIPAZIONE_F_F_KEY_PROGRAMMA foreign key (Titolo_Programma,
Data_uscita_Programma) references PROGRAMMA(Titolo, Data_uscita)
);

create table DISTRIBUZIONE(
Nome_Piattaforma varchar(128) not null,
Numero_Stagione int not null,
Titolo_Serie varchar(128) not null,
Data_uscita_Serie date not null,
constraint DISTRIBUZIONE_P_KEY primary key(Nome_Piattaforma, Numero_Stagione,
Titolo_Serie, Data_uscita_Serie),
constraint DISTRIBUZIONE_F_KEY_PIATTAFORMA foreign key(Nome_Piattaforma)
references PIATTAFORMA(Nome),
constraint DISTRIBUZIONE_F_KEY_STAGIONE foreign key(Numero_Stagione,
Titolo_Serie, Data_uscita_Serie) references STAGIONE(Numero, Titolo_Serie,
Data_uscita_Serie)
);

create table DIVISA(
Titolo_Serie varchar(128) not null,
Data_uscita_Serie date not null,
Numero_Stagione int not null,
constraint DIVISA_P_KEY primary key (Titolo_Serie, Data_uscita_Serie, Numero_Stagione),
constraint DIVISA_F_KEY_SERIE foreign key(Titolo_Serie, Data_uscita_Serie) references SERIE
constraint DIVISA_F_KEY_STAGIONE foreign key (Numero_Stagione, Titolo_Serie, Data_uscita_Serie)
);

create table CONTIENE(
Numero_Stagione int not null,
Titolo_Stagione varchar(64) not null,
Data_uscita_Stagione date not null,
Titolo_Episodio varchar(128) not null,
constraint CONTIENE_P_KEY primary key(Numero_Stagione, Titolo_Stagione, Data_uscita_Stagione,
constraint CONTIENE_F_KEY_STAGIONE foreign key (Numero_Stagione, Titolo_Stagione, Data_uscita_Stagione)
constraint CONTIENE_F_KEY_EPISODIO foreign key(Titolo_Episodio) references EPISODIO(Titolo)
);

```

```

create table PROIEZIONE(
    Titolo_Film varchar(64) not null,
    Data_uscita_Film date not null,
    Nome_Cinema varchar(128) not null,
    Indirizzo_Cinema varchar(128) not null,
    Provincia_Cinema varchar(128) not null,
    Regione_Cinema varchar(128) not null,
    Prezzo int,
    Ora time,
    Data_C date,
    Sala int,
    constraint PROIEZIONE_P_K primary key(Titolo_Film, Data_uscita_Film, Nome_Cinema, Indirizzo_Cinema),
    constraint PROIEZIONE_F_K_FILM foreign key(Titolo_Film, Data_uscita_Film) references FILM(Titolo_Film, Data_uscita_Film),
    constraint PROIEZIONE_F_K_CINEMA foreign key(Nome_Cinema, Indirizzo_Cinema, Provincia_Cinema, Regione_Cinema),
);

INSERT INTO UTENTE (Email,Nome_utente>PasswordU,isRedattore>Data_inizio_collaborazione)
VALUES
('volutpat.Nulla.dignissim@inceptos.com','Galvin Valenzuela','DIcVn7997KIV',true,'2020-27-08'),
('nec.cursus@Phasellus.org','Remedios Ward','MIKKz9C55HIx',false,'2010-08-27'),
('ligula.consectetuer@Morbi.co.uk','Amery Parks','BIaGr8269ZIJ',false,'2011-06-21'),
('ut.aliquam.iaculis@Sed.com','Kelly Cox','TIEGp8A37FIM',false,'2020-01-29'),
('tempus@blandit.ca','Robin Hernandez','BIPKc1413FIa',false,'2011-10-08');

INSERT INTO VOTO (Id,Numero_stelle>Email_Utente)
VALUES
(1,5,'volutpat.Nulla.dignissim@inceptos.com'),
(2,1,'nec.cursus@Phasellus.org'),
(3,4,'ligula.consectetuer@Morbi.co.uk'),
(4,2,'ut.aliquam.iaculis@Sed.com'),
(5,4,null);

INSERT INTO PIATTAFORMA (Nome)
VALUES
('Netflix'),
('Disney+'),
('AmazonPrimeVideo'),
('Infinity'),
('Sky Go');

INSERT INTO CINEMA (Nome, Indirizzo, Provincia, Regione, Contatti)
VALUES
('Cinema Multisala Lux','Via Massaciuccoli, 33, 00199 Roma', 'RM', 'Lazio', '+390686391361'),
('Cinema Lumière', 'Via Azzo Gardino, 65, 40122 Bologna', 'BO', 'Emilia-Romagna', '+390512195311'),
('La Casa del Cinema','Salizada San Stae, 1990, 30135 Venezia', 'VE', 'Veneto', '+390412747140'),
('Cinema Massimo','Via Giuseppe Verdi, 18, 10124 Torino', 'TO', 'Piemonte', '+390118138574'),
('Arcadia', 'Via Martiri della Libertà, 5, 20066 Melzo', 'MI', 'Lombardia', '+390295416445');

```



```

insert into PROGRAMMA (Titolo, Data_uscita, Durata, Genere, Produzione, Paese, Troupe, Distribuzione)
values
('Blob','1989-01-12' , '20', 'Satirico', 'Rai', 'Italia', 'Enrico Ghezzi, Marco Giusti, Angelo Gualerzi', 'Rai'),
('Antonino Chef Academy', ' 2019-10-03', '90', ' Enogastronomico', 'Sky', 'Italia', 'Antonino Chef Academy', 'Sky'),
('Fratelli Crozza', '2017-02-12', '70', 'Show', 'Andrea Zalone', 'Italia', '', 'Rai', 'Nel corso della storia'),
('Alessandro Borghese - 4 ristoranti', '2015-12-10', '60', 'Talent Show', 'Sky', 'Italia', 'Alessandro Borghese', 'Sky'),
('Forum', '1985-01-02', '120', 'Show', 'Mediaset', 'Italia', 'Catherine Spaak, Paola Perego, Rita Barberis', 'Mediaset'),

insert into SERIE values('I Medici','2016-03-07',50, 'Drammatico', 'Big Light Productions, Lux Video Service', 'USA'),
insert into SERIE values('The Queen''s Gambit', '2020-02-04',46, 'Drammatico', 'Marcus Loges, M. J. Loges', 'USA'),
insert into SERIE values('The Mandalorian', '2019-07-05', 42, 'Fantascienza', 'Lucasfilm', 'USA'),
insert into SERIE values('Breaking Bad', '2008-02-28', 47, 'Thriller', 'High Bridge Entertainment', 'USA'),
insert into SERIE values('Il trono di spade', '2011-11-03', 60 , 'Fantasy', 'Television 360', 'USA'),

insert into STAGIONE values(1,'The Queen''s Gambit', '2020-02-04');
insert into STAGIONE values(2,'Breaking Bad', '2008-02-28');
insert into STAGIONE values(1,'Breaking Bad', '2008-02-28');
insert into STAGIONE values(3,'Il trono di spade', '2011-11-03');
insert into STAGIONE values(1,'Il trono di spade', '2011-11-03');

insert into EPISODIO values('Valar Dohaeris', 55, 'Ciarán Hinds, Michael McElhatton, Ian McElhinney', 'USA'),
insert into EPISODIO values('The Climb', 55, ' Gwendoline Christie, Ian McElhinney');
insert into EPISODIO values('Crazy Handful of Nothin'', 55, ' Max Arciniega, John Koyama');
insert into EPISODIO values('Gray Matter', 45, 'Roberta Marquez, Allan Pacheco ');
insert into EPISODIO values('"Exchanges"', 30,'Anya Taylor-Joy');

insert into FILM values('Matrix','1999-04-02', 133, 'Fantascienza', 'Lana Wachowski, Lilly Wachowski', 'USA'),
insert into FILM values ('Teenage Mutant Ninja Turtles', '2014-05-20', 101, 'Azione', 'Paramount Pictures', 'USA'),
insert into FILM values ('Il cosmo sul como', '2008-07-15', 94, 'Commedia', 'Medusa Film', 'Italia'),
insert into FILM values ('Ghost Rider - Spirito di vendetta', '2012-03-23', 95, 'Azione', 'Medusa Film', 'Italia'),
insert into FILM values ('Il gigante di ferro', '1999-08-14', 86, 'Animazione', 'Warner Bros.', 'USA'),

INSERT INTO Preferisce_F (Email_Utente,Titolo_Film,Data_uscita_Film)
VALUES
('volutpat.Nulla.dignissim@inceptos.com','Matrix','1999-04-02'),
('nec.cursus@Phasellus.org','Teenage Mutant Ninja Turtles', '2014-05-20');

INSERT INTO Preferisce_S (Email_Utente,Titolo_Serie,Data_uscita_Serie)
VALUES
('ligula.consectetuer@Morbi.co.uk','The Mandalorian', '2019-07-05'),
('ut.aliquam.iaculis@Sed.com','Breaking Bad', '2008-02-28',);

INSERT INTO Preferisce_P (Email_Utente,Titolo_Programma,Data_uscita_Programma)
VALUES
('tempus@blandit.ca','Blob','1989-01-12');

INSERT INTO Riferisce_F (Id_Voto,Titolo_Film,Data_uscita_Film)

```

```

VALUES
('volutpat.Nulla.dignissim@inceptos.com','Il gigante di ferro', '1999-08-14');

INSERT INTO Riferisce_S (Id_Voto,Titolo_Serie,Data_uscita_Serie)
VALUES
('ligula.consectetuer@Morbi.co.uk','Il trono di spade', '2011-11-03'),
('ut.aliquam.iaculis@Sed.com','I Medici','2016-03-07');

INSERT INTO Riferisce_P (Id_Voto,Titolo_Programma,Data_uscita_Programma)
VALUES
('ut.aliquam.iaculis@Sed.com','Alessandro Borghese - 4 ristoranti', '2015-12-10');

INSERT INTO Distribuzione (Nome_Piattaforma, Numero_Stagione, Titolo_Serie, Data_uscita_Serie)
VALUES
('Netflix', 1, 'The Queen''s Gambit', '2020-02-04'),
('AmazonPrimeVideo', 2, 'Breaking Bad', '2008-02-28'),
('Infinity', 1, 'Breaking Bad', '2008-02-28'),
('Infinity', 3, 'Il trono di spade', '2011-11-03');
('Infinity', 3, 'Il trono di spade', '2011-11-03');

INSERT INTO Divisa (Titolo_Serie, Data_uscita_Serie, Numero_Stagione)
VALUES
('The Queen''s Gambit', '2020-02-04', 1),
('Breaking Bad', '2008-02-28', 2),
('Breaking Bad', '2008-02-28', 1),
('Il trono di spade', '2011-11-03', 3),
('Il trono di spade', '2011-11-03', 3);

insert into ARTISTA (Nome, Cognome, Data_di_nascita, Luogo_di_nascita, Biografia_testuale, Ultimo_nome)
values
('Anya Josephine Marie', 'Taylor-Joy','1996-04-16', 'Miami','Anya Taylor-Joy nasce a Miami, in Florida', 'Taylor-Joy')
insert into ARTISTA values
('Maurizio', 'Crozza', '1959-12-5', 'Genova', 'Nato a Genova nel quartiere di Borgoratti, primo figlio di un'operaio', 'Crozza')
insert into ARTISTA values
('Amethyst', 'Bentley','1989-09-07','Villafranca d''Asti', 'Famosa per la sua interpretazione di Luce', 'Bentley')
insert into ARTISTA values
('Mara', 'Herring','1940-05-13','Ajmer', 'Nata da madre polacca e padre turco si appassiona fin da piccola al cinema', 'Herring')
insert into ARTISTA values
('Echo', 'Petty','1977-08-28','Lancaster', 'Famosa per il suo talento nel recitare la parte del gatto', 'Petty')

insert into PARTECIPAZIONE_F (Nome_Artista, Cognome_Artista, Data_di_nascita_Artista, Titolo_Film, Anno, Ruolo)
values
('Anya Josephine Marie', 'Taylor-Joy','1996-04-16', 'Matrix','1999-04-02', 'Regista', null),
('Mara', 'Herring','1940-05-13', 'Ghost Rider - Spirito di vendetta', 'Attore', 'Sedia');
insert into PARTECIPAZIONE_S (Nome_Artista, Cognome_Artista, Data_di_nascita_Artista, Titolo_Serie, Anno, Ruolo)
values
('Amethyst', 'Bentley','1989-09-07', 'Breaking Bad', '2008-02-28', 'Attore', 'Walter Jr');
insert into PARTECIPAZIONE_P (Nome_Artista, Cognome_Artista, Data_di_nascita_Artista, Titolo_Programma, Anno, Ruolo)

```

```

values
('Maurizio', 'Crozza', '1959-12-5', 'Fratelli Crozza', '2017-02-12', 'Attore', 'Crozza'),
('Amethyst', 'Bentley', '1989-09-07', 'Alessandro Borghese - 4 ristoranti', '2015-12-10', 'Regista', 'Bentley'),
('Echo', 'Petty', '1977-08-28', 'Antonino Chef Academy', '2019-10-03', 'Attore', 'Mestolo');

insert into CONTIENE values (1, 'The Queen''s Gambit', '2020-02-04', 'Exchanges');
insert into CONTIENE values (1, 'Breaking Bad', '2008-02-28', 'Crazy Handful of Nothin');
insert into CONTIENE values (2, 'Breaking Bad', '2008-02-28', 'Gray Matter');
insert into CONTIENE values (3, 'Il trono di spade', '2011-11-03', 'Valar Dohaeris');
insert into CONTIENE values (1, 'Il trono di spade', '2011-11-03', 'The Climb');

insert into Proiezione (Titolo_Film, Data_uscita_Film, Nome_Cinema, Indirizzo_Cinema, Provincia, Data_inizio, Data_fine)
values
('Matrix', '1999-04-02', 'Cinema Multisala Lux', 'Via Massaciuccoli, 33, 00199 Roma', 'RM', 'Lazio', '1999-04-02', '1999-04-02'),
('Teenage Mutant Ninja Turtles', 'Cinema Multisala Lux', 'Via Massaciuccoli, 33, 00199 Roma', 'RM', 'Lazio', '1999-04-02', '1999-04-02'),
('Il cosmo sul como', '2008-07-15', 'Cinema Multisala Lux', 'Via Massaciuccoli, 33, 00199 Roma', 'RM', 'Lazio', '2008-07-15', '2008-07-15'),
('Il cosmo sul como', '2008-07-15', 'Cinema Lumière', 'Via Azzo Gardino, 65, 40122 Bologna', 'BO', 'Emilia-Romagna', '2008-07-15', '2008-07-15'),
('Il gigante di ferro', '1999-08-14', 'La Casa del Cinema', 'Salizada San Stae, 1990, 30135 Venezia', 'VE', 'Veneto', '1999-08-14', '1999-08-14');

```