

Progetto di basi di dati

TV SHOW TIME



A.A 2017-2018
DESIDERIO MATTIA - GIBELLINI GIORGIA

INDICE

Descrizione della realtà da analizzare	p.3
Glossario	p.5
Schema scheletro	p.6
Progettazione concettuale	p.7
Diagramma ER completo	p.13
Progettazione logica	p.14
Eliminazione delle gerarchie ISA	p.14
Selezione delle chiavi primarie ed eliminazione delle chiavi esterne	p.15
Trasformazione degli attributi e/o composti	p.18
Traduzioni di entità e associazioni in schema di relazioni	p.19
Verifica della normalizzazione	p.23
Vincoli aggiuntivi	p.24
Query di creazione	p.25
Query di inserimento	p.29
Query di interrogazione	p.36
Query di modifica	p.38
Query di eliminazione	p.39
Triggers	p.40
Test delle operazioni previste ed esiti	p.42
Studio di dati derivati	p.45

DESCRIZIONE DELLA REALTÀ DA ANALIZZARE

L'applicazione "Tv Show Time" rende possibile agli utenti la gestione della visione delle serie tv e si compone di vari elementi:

Le persone sono caratterizzate da un codice fiscale univoco, nome, cognome, data di nascita, luogo di nascita e la nazionalità.

Le tipologie di persone che l'applicazione considera sono utenti e attori, qualsiasi persona deve rientrare in una delle due categorie e non può appartenere contemporaneamente ad entrambe; un utente quindi non può essere anche un attore e viceversa.

Gli utenti oltre alle caratteristiche di persona hanno anche le seguenti: e-mail, nickname, ultimo accesso e password.

Un utente all'interno dell'applicazione può guardare puntate di serie tv.

Un utente può dare un voto da 1 a 10 a ogni puntata, la puntata può essere votata una sola volta. Un utente può votare una puntata solo se l'ha già vista.

Un utente può fare una sola recensione per ogni serie tv e solo se ne ha visto almeno un episodio.

Un utente può scegliere uno o più generi di serie tv preferiti di cui è rilevante solo conoscere il nome (storico, fantasy, fantascienza, etc.).

Una recensione è caratterizzata da un titolo e dal testo della recensione; un utente può recensire più serie tv ma ognuna una sola volta.

Un attore può avere come ulteriore caratteristica rispetto a persona una breve biografia.

Un attore fa parte del cast di una serie tv in veste di uno o più personaggi.

Un attore può essere nel cast di più serie tv.

Ogni puntata è caratterizzata da un titolo e da una durata (può cambiare da episodio a episodio). Ogni puntata è identificata univocamente dal suo numero, dal numero della stagione e dal titolo e anno della serie tv a cui appartiene.

Una certa puntata appartiene ad una ed una sola stagione mentre una stagione ha uno o più episodi.

Ogni puntata viene trasmessa da un canale in una certa data e ora. Sullo stesso canale non possono essere trasmesse in contemporanea più puntate. Un episodio può essere trasmesso su vari canali nello stesso momento o in momenti diversi.

Ciascuna stagione di una serie tv è identificata univocamente dal numero della stagione, dal titolo della serie tv e dall'anno di inizio della serie tv. Caratteristiche della stagione sono l'anno di uscita e il numero di episodi. Una certa stagione appartiene a una ed una sola serie tv, mentre una serie tv ha da una a più stagioni.

Una serie tv è identificata univocamente dal titolo e l'anno di inizio, ha inoltre come caratteristiche: una o più case produttrici, una breve trama e il numero di stagioni complessivo.

Ogni serie tv è caratterizzata da uno o più generi.

ANNOTAZIONI:

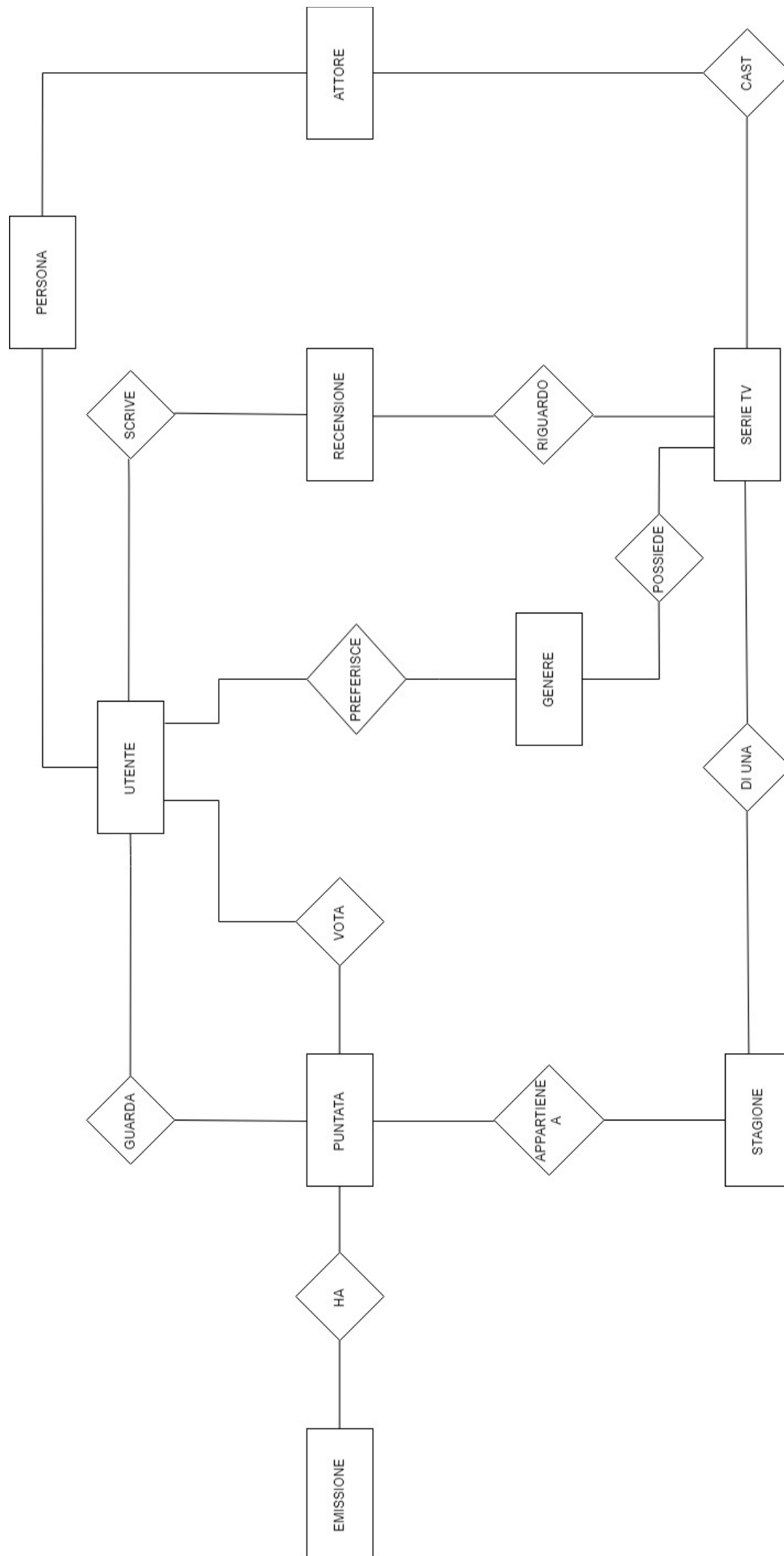
Se una puntata viene vista più volte, vengono aggiornate sia la data di ultima visione che il dispositivo tramite il quale la visione è stata fatta.

Si vuole tenere conto delle ore di visioni totali di serie tv dell'utente con la seguente particolarità: la visione ripetuta di uno stesso episodio viene conteggiata una sola volta nel complessivo delle ore di serie tv guardate.

GLOSSARIO

Termine	Descrizione	Sinonimi	Legami
Persona	Codice fiscale Nome Cognome Luogo di nascita Nazionalità		Utente Attore
Utente	Email Password Nickname Ultimo accesso	User Persona	Puntata Genere Recensione Persona
Attore	Biografia		Serie tv Persona
Puntata	Titolo Durata Numero puntata	Episodio	Stagione Utente Emissione
Stagione	Numero di episodi Numero della stagione Anno		Puntata Serie tv
Serie tv	Titolo Numero delle stagioni Casa produttrice Trama	Serie	Stagione Genere Recensione Attore
Genere	Nome	Tipo	Utente Serie tv
Recensione	Titolo Testo		Utente Serie tv
Emissione	Canale Data		Puntata

SCHEMA SCHELETRO



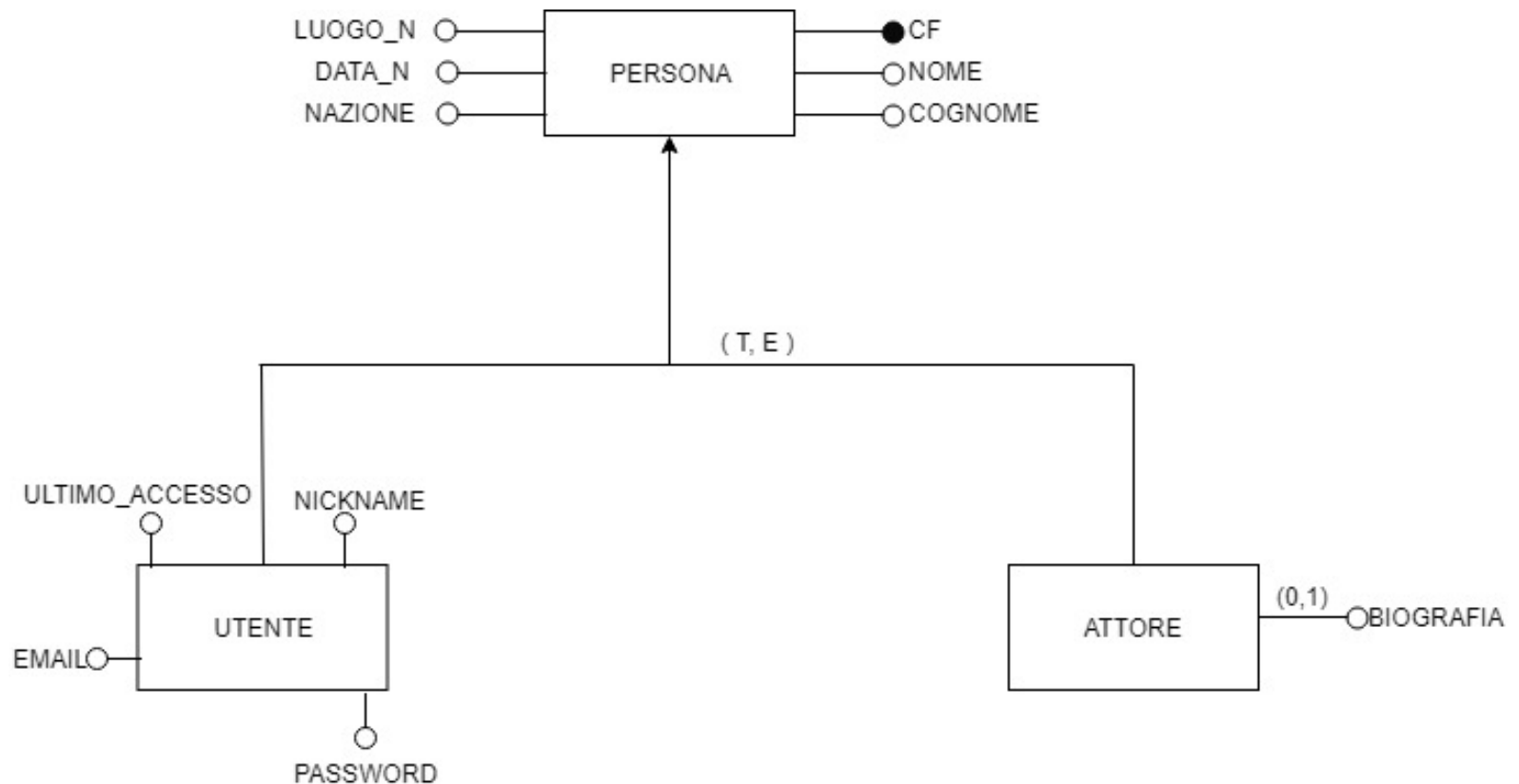
PROGETTAZIONE CONCETTUALE

"[...] Le persone sono caratterizzate da un codice fiscale univoco, nome, cognome, data di nascita, luogo di nascita e la nazionalità.

Le tipologie di persone che l'applicazione considera sono utenti e attori, qualsiasi persona deve rientrare in una delle due categorie e non può appartenere contemporaneamente ad entrambe; un utente quindi non può essere anche un attore e viceversa.

Gli utenti oltre alle caratteristiche di persona hanno anche le seguenti: e-mail, nickname, ultimo accesso e password.

[...]Un attore può avere come ulteriore caratteristica rispetto a persona una breve biografia.
[...]"



Abbiamo deciso di rappresentare le due entità utente e attore come entità figlie di persona; ciò è stato fatto perchè le due entità hanno in comune i dati anagrafici tipici delle persone quali nome, cognome, data e luogo di nascita, nazionalità e codice fiscale. La gerarchia in questione è totale perché ogni persona che rientra nella base di dati deve far parte o degli utenti oppure degli attori delle serie tv, è inoltre esclusiva poiché un attore non può essere anche un utente e viceversa.

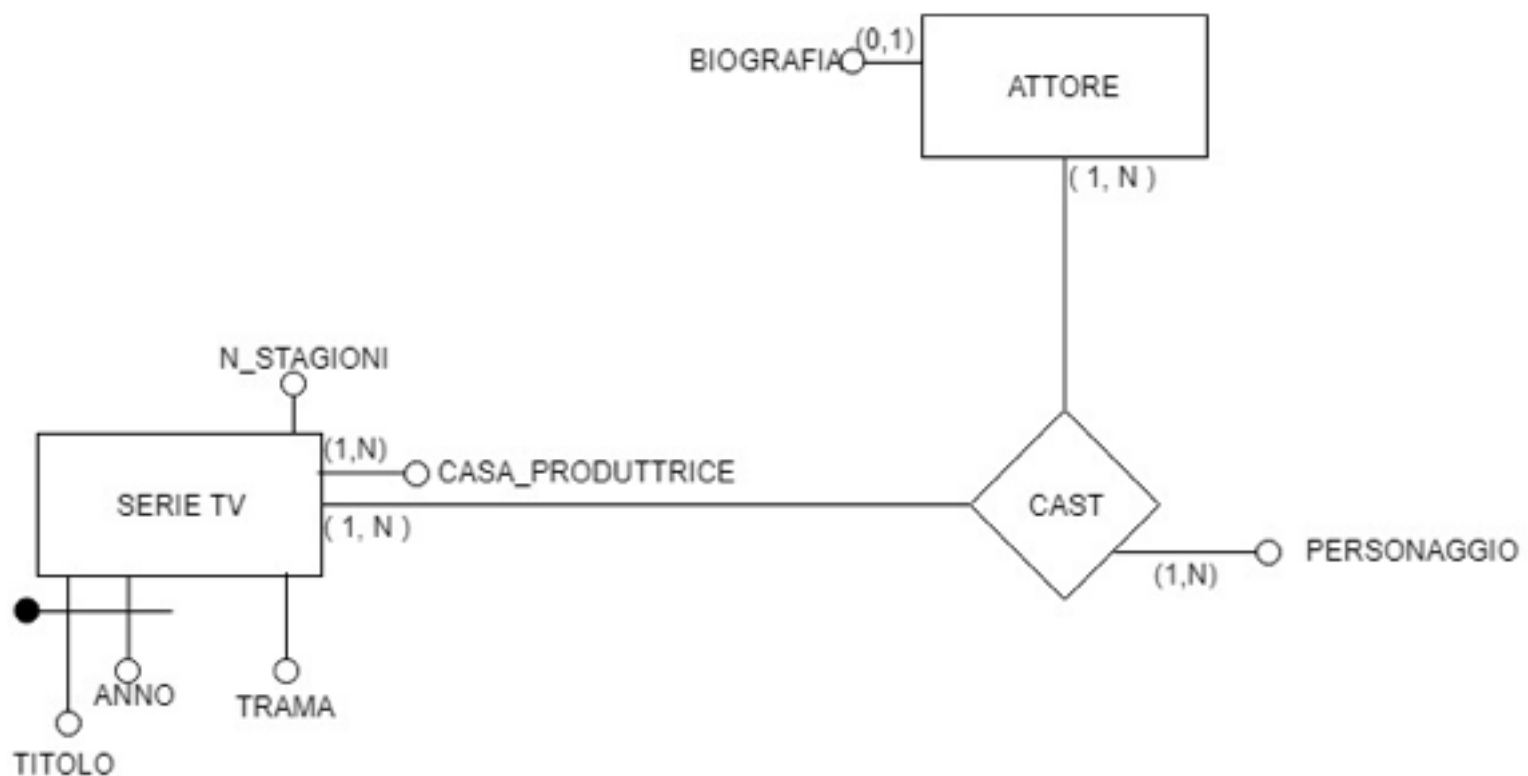
Inizialmente avevamo pensato di creare un attributo "ore visione" su utente per registrare il monte ore totale di serie tv viste da un utente, ma facendo alcune osservazioni sui costi è

emerso che questo dato sarebbe stato molto più costoso rispetto a calcolarlo direttamente in una query; per questo motivo abbiamo eliminato il suddetto dato derivato.

“ [...] Un attore fa parte del cast di una serie tv in veste di uno o più personaggi.

Un attore può essere nel cast di più serie tv. [...]”

[...] Una serie tv è identificata univocamente dal titolo e l’anno di inizio, ha inoltre come caratteristiche: una o più case produttrici, una breve trama e il numero di stagioni complessivo. [...]”



Una caratteristica particolare delle serie tv è che durante l’evoluzione della serie tv possono aggiungersi case produttrici oltre a quelle iniziali. Le case produttrici che si sono succedute durante lo svolgersi della serie tv non hanno importanza nello specifico associate alla stagione ma le abbiamo percepite come un’informazione relativa alla serie tv nella sua globalità.

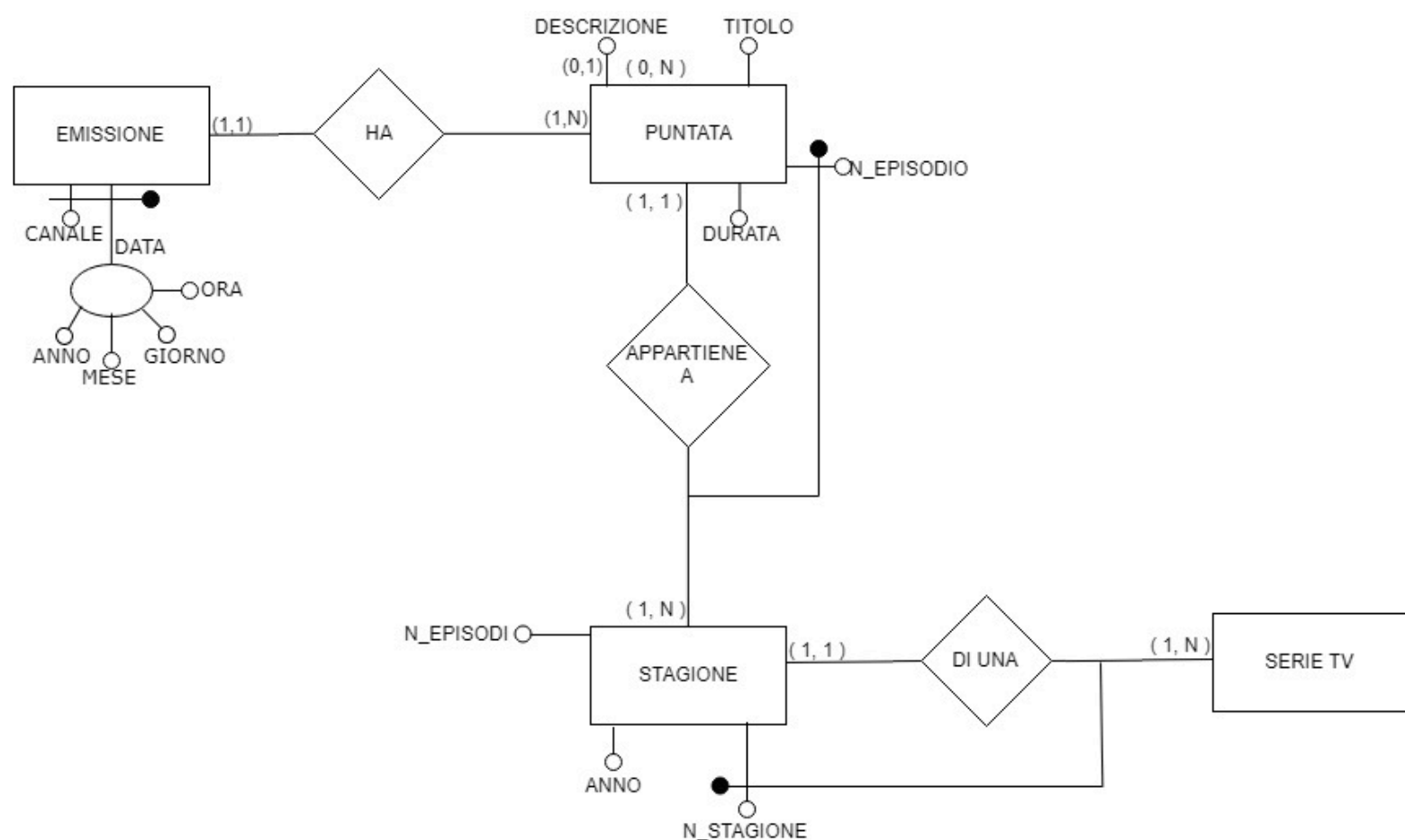
Per riuscire a rappresentare anche il caso di serie tv omonime abbiamo usato la chiave composta da “titolo” e “anno” per l’entità “serie tv” in modo tale che se due serie tv hanno lo stesso nome (ad esempio nel caso di un remake cfr. “Skins”) la loro chiave differisce per l’anno; infatti è pressoché impossibile che due serie tv omonime vengano prodotte lo stesso anno.

"[...]Ogni puntata è caratterizzata da un titolo e una durata (può cambiare da episodio a episodio). Ogni puntata all'interno di una stagione di una serie tv è identificata univocamente dal suo numero, dal numero della stagione e dal titolo e anno della serie tv a cui appartiene.

Una certa puntata appartiene ad una ed una sola stagione mentre una stagione ha uno o più episodi.

Ogni puntata viene trasmessa da un canale in una certa data e ora. Sullo stesso canale non possono essere trasmesse in contemporanea più puntate. Un episodio può essere trasmesso su vari canali nello stesso momento o in momenti diversi.

Ciascuna stagione di una serie tv è identificata univocamente dal numero della stagione, dal titolo della serie tv e dall'anno di inizio della serie tv. Caratteristiche della stagione sono l'anno di uscita e il numero di episodi. Una certa stagione appartiene a una ed una sola



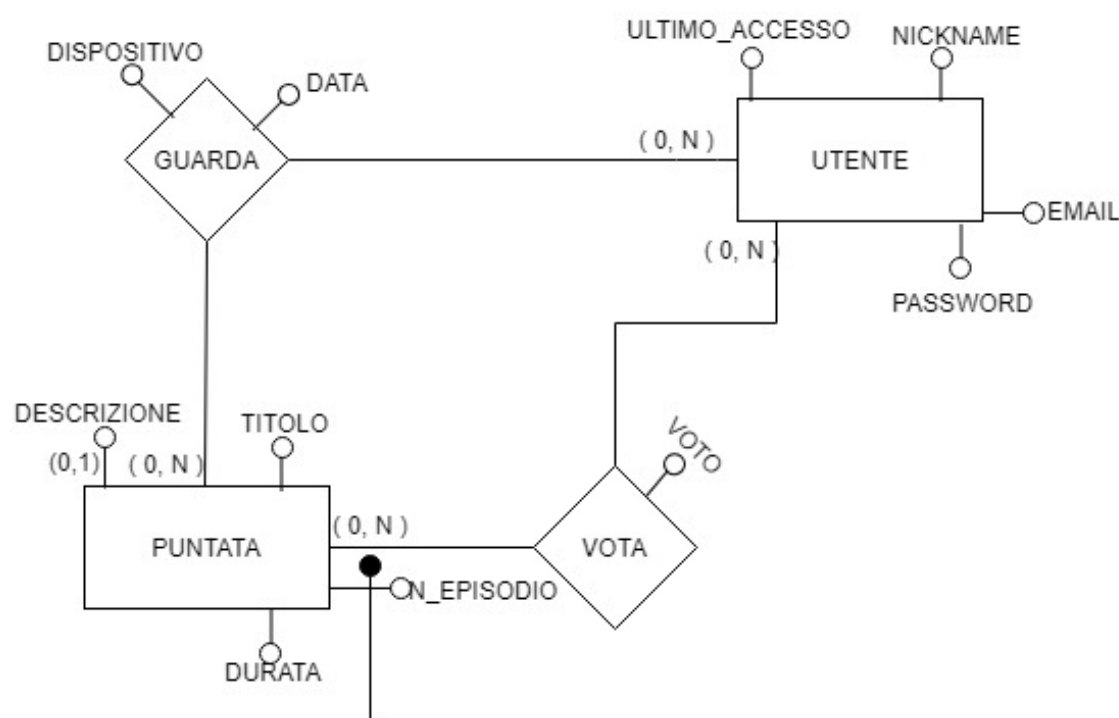
serie tv, mentre una serie tv ha da una a più stagioni. [...]"

Abbiamo preso la decisione di rendere "durata" un attributo dell'entità "puntata" anziché di "stagione" poiché in ogni puntata la durata può essere leggermente diversa e dato che nella nostra visione del problema essa è un dato rilevante (viene usato in due interrogazioni SQL) abbiamo preferito fare in questo modo anche se siamo consapevoli che potranno esserci valori molto simili se non uguali delle durate dei vari episodi.

Un altro elemento che all'inizio avevamo pensato di porre relativo a "stagione" ma che poi abbiamo trattato diversamente è quello relativo alla data e al canale di visione/emissione. Infatti se avessimo messo un attributo composto "data" (formato da giorno, mese, anno e ora) e un attributo "canale" relativi a "stagione" per indicare le informazioni di trasmissione in diretta televisiva su un certo canale di quella serie tv per quella stagione non avremmo avuto modo di gestire il palinsesto ed evitare la sovrapposizione di più episodi sullo stesso canale alla stessa ora. Inoltre può capitare che nonostante una serie tv per quella determinata stagione sia trasmessa usualmente in un giorno e ora della settimana su uno specifico canale in caso di giorni di festività o interruzione della serie tv ciò potrebbe non accadere e portare a inconsistenza dei dati. Per aderire meglio alla realtà dei fatti abbiamo deciso quindi di creare un'entità "emissione" contenente il canale di emissione e la data, attributo composto formato da: anno, mese, giorno e ora. Per rispettare il vincolo di non contemporaneità di più episodi su uno stesso canale nella stessa data (intesa come anno, mese, giorno, ora) la cardinalità tra "emissione" e "episodio" è (1,1) mentre per consentire la trasmissione di uno stesso episodio su vari canali, volendo anche nella stessa data, la cardinalità tra "episodio" e "emissione" è (1,N).

Per quanto concerne l'entità "puntata" la scelta di porre come attributo opzionale "descrizione" deriva dal fatto che spesso serie tv di minor rilevanza non vengono curate così accuratamente da avere una descrizione relativa ad ogni puntata e per questo ci sembrava più appropriato sottolineare l'opzionalità del suddetto attributo.

"[...] Un utente all'interno dell'applicazione può guardare puntate di serie tv.



Un utente può dare un voto da 1 a 10 a ogni puntata, la puntata può essere votata una sola volta. Un utente può votare una puntata solo se l'ha già vista.[...] "

Per quanto riguarda la gestione della visione molteplice di uno stesso episodio abbiamo optato per questa soluzione, ovvero nel caso di visione ripetuta sovrascrittura degli attributi "data" e "dispositivo" e non conteggio delle ore nelle ore di visione totale, poiché è più attinente alla applicazione reale a cui ci siamo ispirati. Inoltre di norma è difficile che un utente riguardi la stessa puntata più volte a meno che si tratti di un maratona e, in quel caso, guardando tutti gli episodi di seguito non ha molta utilità a quale episodio si è arrivati per ricordarselo.

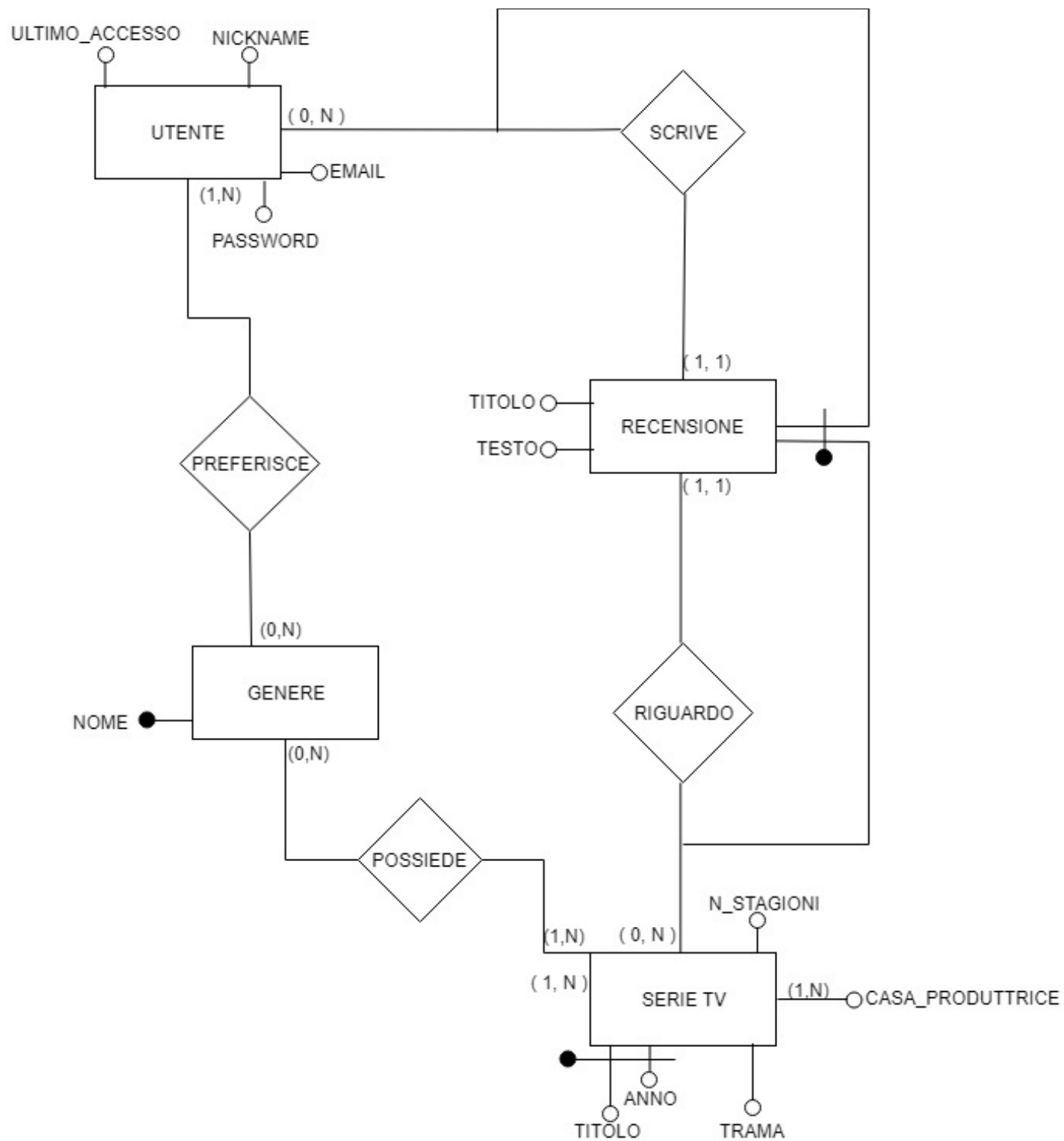
La facoltà di un utente di votare con un punteggio da 1 a 10 un episodio è espressa tramite la relazione "vota" che possiede l'attributo "voto". Abbiamo dato per assunto che un utente possa votare solo le puntate che ha già visto, per questo abbiamo fatto un trigger che crea un'eccezione nel caso in cui un utente cerchi di votare una puntata non vista. Questa soluzione ha come aspetto negativo il fatto che il vincolo che lega la visione di un episodio alla possibilità di votarlo non è espresso esplicitamente nel modello ER, inoltre è abbastanza oneroso a livello di costo; aspetto positivo è che in questo modo si ha la sicurezza che non vengano immessi voti non validi e si consente all'utente di dare un voto alla puntata anche a posteriori rispetto alla visione dell'episodio. Un'altra soluzione che abbiamo valutato rispetto a quella precedentemente esposta consisteva nell'integrare l'attributo "voto" sulla relazione "guarda"; in questa maniera il vincolo di poter votare solo se la puntata è stata vista dall'utente viene mantenuto. Punto debole di questa soluzione è che, se molti utenti non hanno l'abitudine di votare l'episodio, nella colonna "voto" della tabella "guarda" ci sarà un enorme spreco di spazio di memoria dovuto ai valori nulli. Inoltre posizionando l'attributo "voto" nella relazione "guarda" l'utente si vedrebbe costretto a dover votare nel momento della visione cosa che risulterebbe molto scomoda e poco realistica in un'applicazione reale.

"[...] Un utente può fare una sola recensione per ogni serie tv e solo se ne ha visto almeno più di un episodio.

Un utente può scegliere uno o più generi di serie tv preferiti di cui è rilevante solo conoscere il nome (storico, fantasy, fantascienza, etc.).

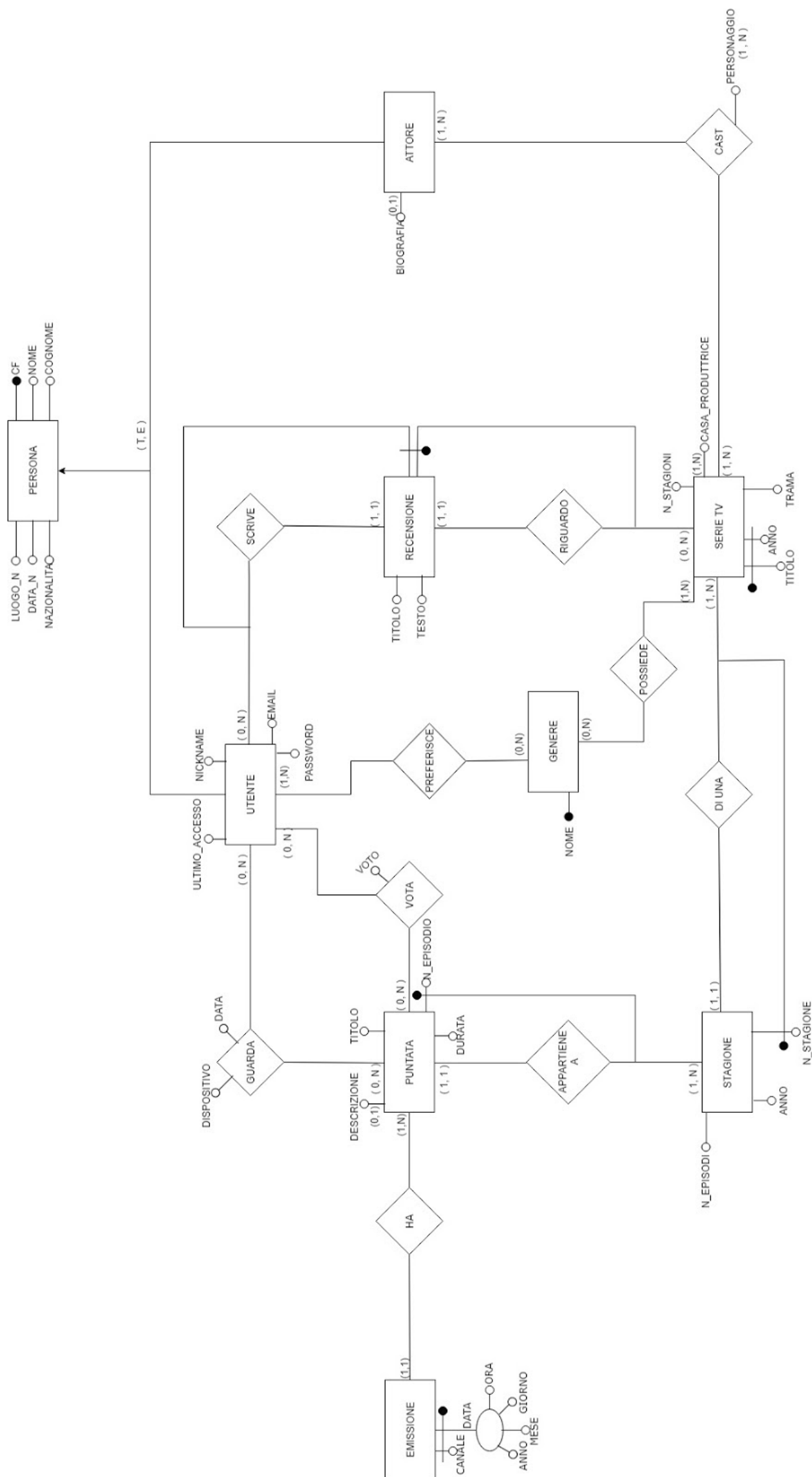
Una recensione è caratterizzata da un titolo e dal testo della recensione; un utente può recensire più serie tv ma ognuna una sola volta.

Ogni serie tv è caratterizzata da uno o più generi. "



Per riuscire a rendere al meglio il vincolo imposto sulle recensioni, ovvero che un utente può fare una sola recensione riguardo una specifica serie tv ma può recensire nessuna, una o più serie tv diverse tra loro, abbiamo usato sia le cardinalità che le chiavi esterne. Infatti per far rispettare in maniera totale il vincolo la cardinalità da "utente" a "recensione" e da "serie tv" a "recensione" deve essere (0,N) così da permettere a un utente di fare da 0 a N recensioni e alla serie tv di avere da 0 a N recensioni. La sola cardinalità però permetteva a un dato utente di recensire più volte la stessa serie tv aspetto contrastante alle caratteristiche volute. Per risolvere questo problema abbiamo reso chiave primaria dell'entità "recensione" la coppia primary key di "utente" e primary key di "serie tv"; in questo modo a un codice fiscale utente e una coppia titolo serie tv, anno serie tv

corrisponde una ed una sola recensione, il vincolo imposto viene quindi rispettato in



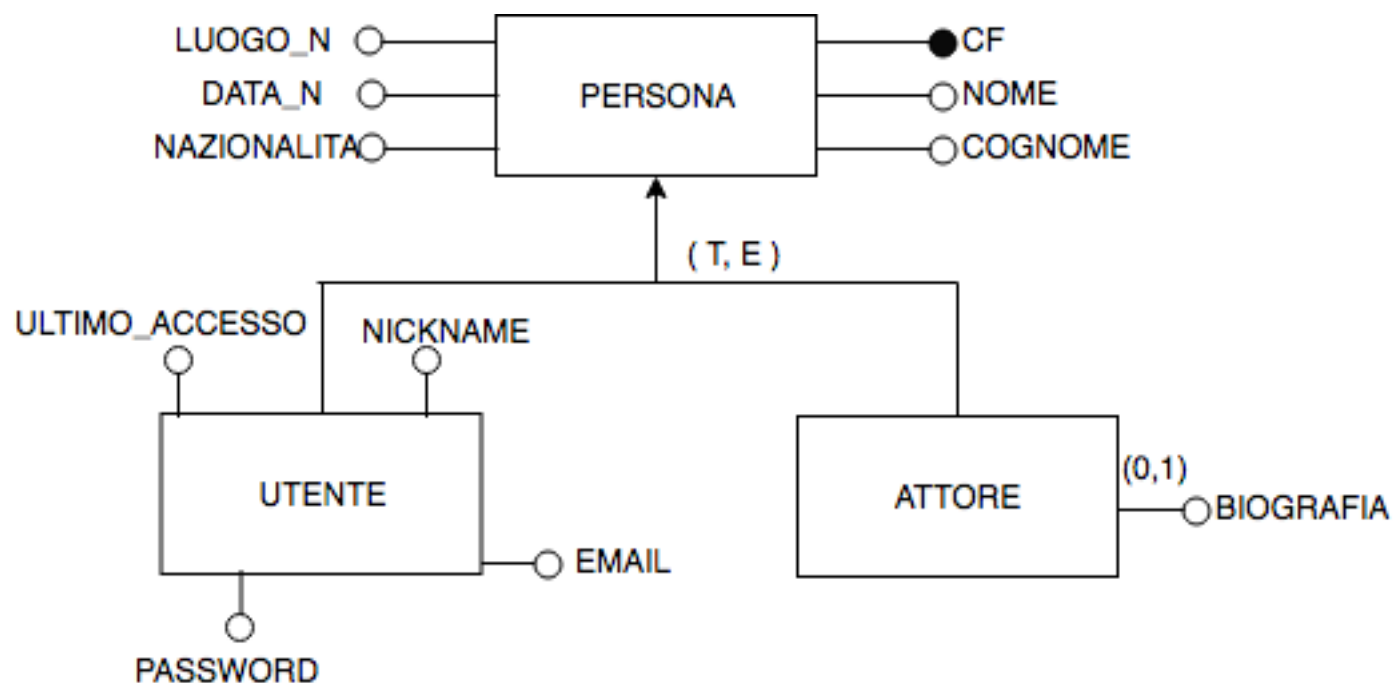
maniera totale.

DIAGRAMMA ER COMPLETO

PROGETTAZIONE LOGICA

La progettazione concettuale costituisce una descrizione sintetica e visiva della base di dati. Per poter però far sì che il DBMS sia in grado di operare su questi concetti è necessario trasformare lo schema ER in uno schema logico relazionale attraverso le seguenti fasi:

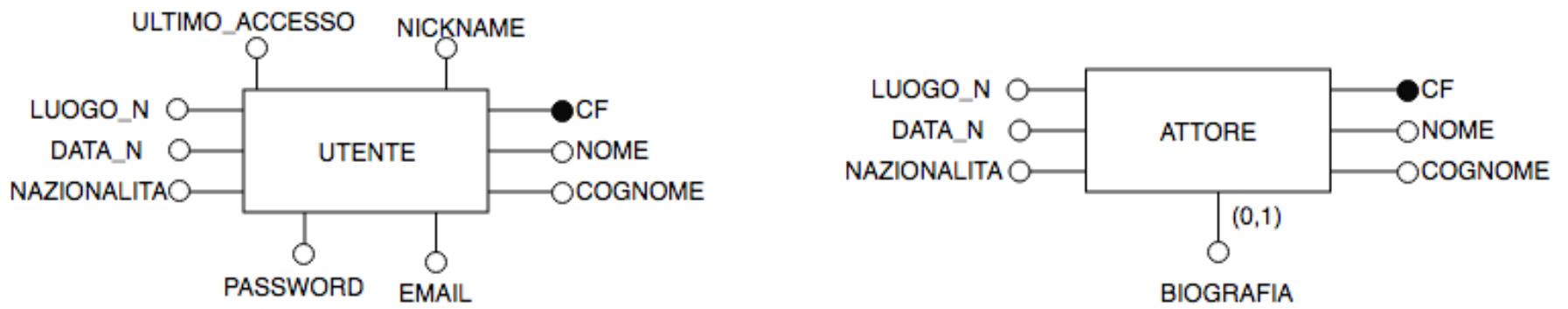
1. Eliminazione delle gerarchie ISA.
2. Selezione delle chiavi primarie e eliminazione delle identificazioni esterne.
3. Trasformazione degli attributi composti e/o multipli.
4. Traduzione di entità e associazioni in schemi di relazioni
5. Verifica della normalizzazione.



1 - ELIMINAZIONE DELLE GERARCHIE ISA

Prima di tutto è necessario rimuovere le gerarchie.

Si è ritenuto più conveniente effettuare il collasso verso il basso in quanto l'entità padre non si interfaccia con nessuna entità della base di dati e la copertura è totale ed esclusiva. Le richieste di informazioni, infatti, ricadono interamente sulle entità figlie.

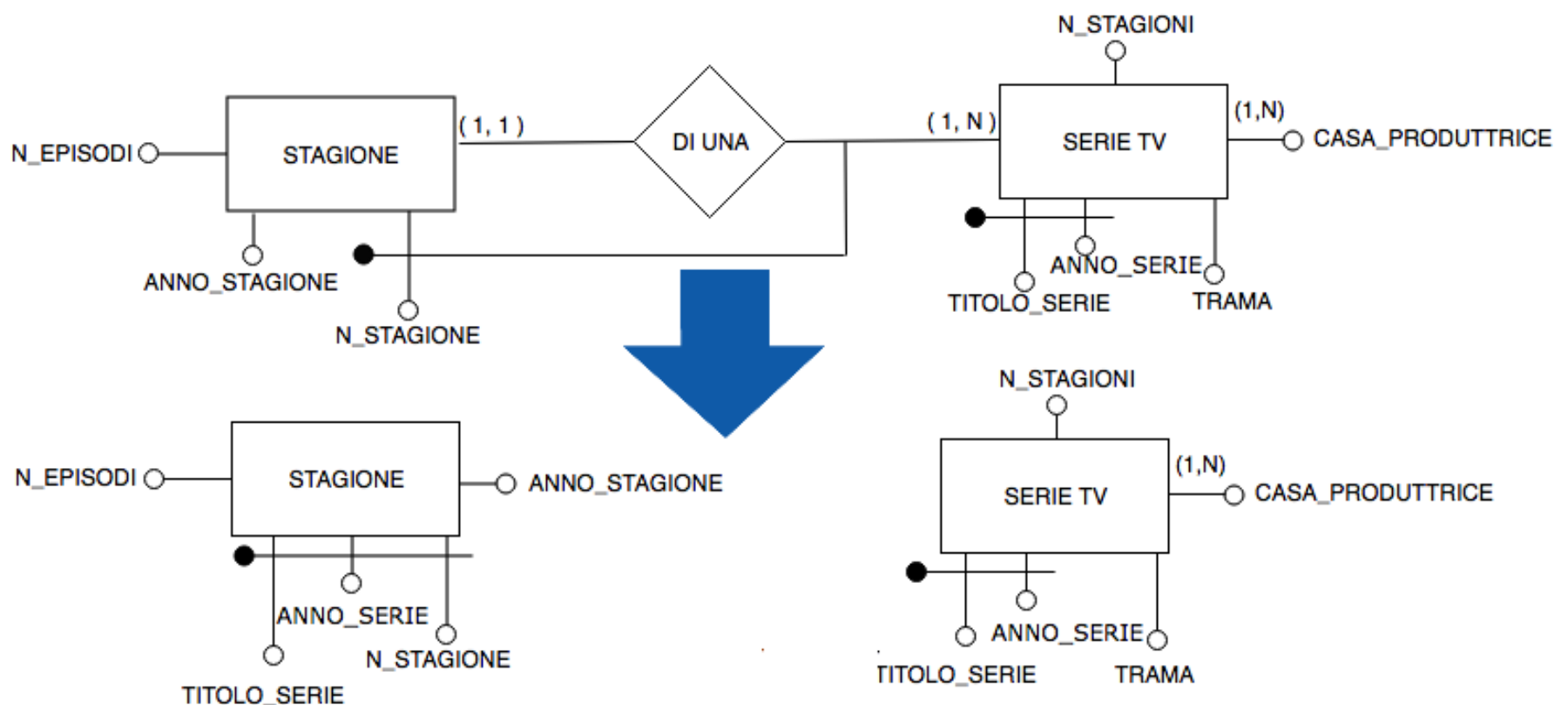


L'entità padre "Persona" viene quindi eliminata e si trasferiscono gli attributi sulle entità figlie "Utente" e "Attore". Questa scelta favorisce le operazioni in cui si accede separatamente alle entità figlie.

Il nickname dell'utente è una valida alternative key, in quanto è unico e non può essere nullo.

2 - SELEZIONE DELLE CHIAVI PRIMARIE ED ELIMINAZIONE DELLE CHIAVI ESTERNE

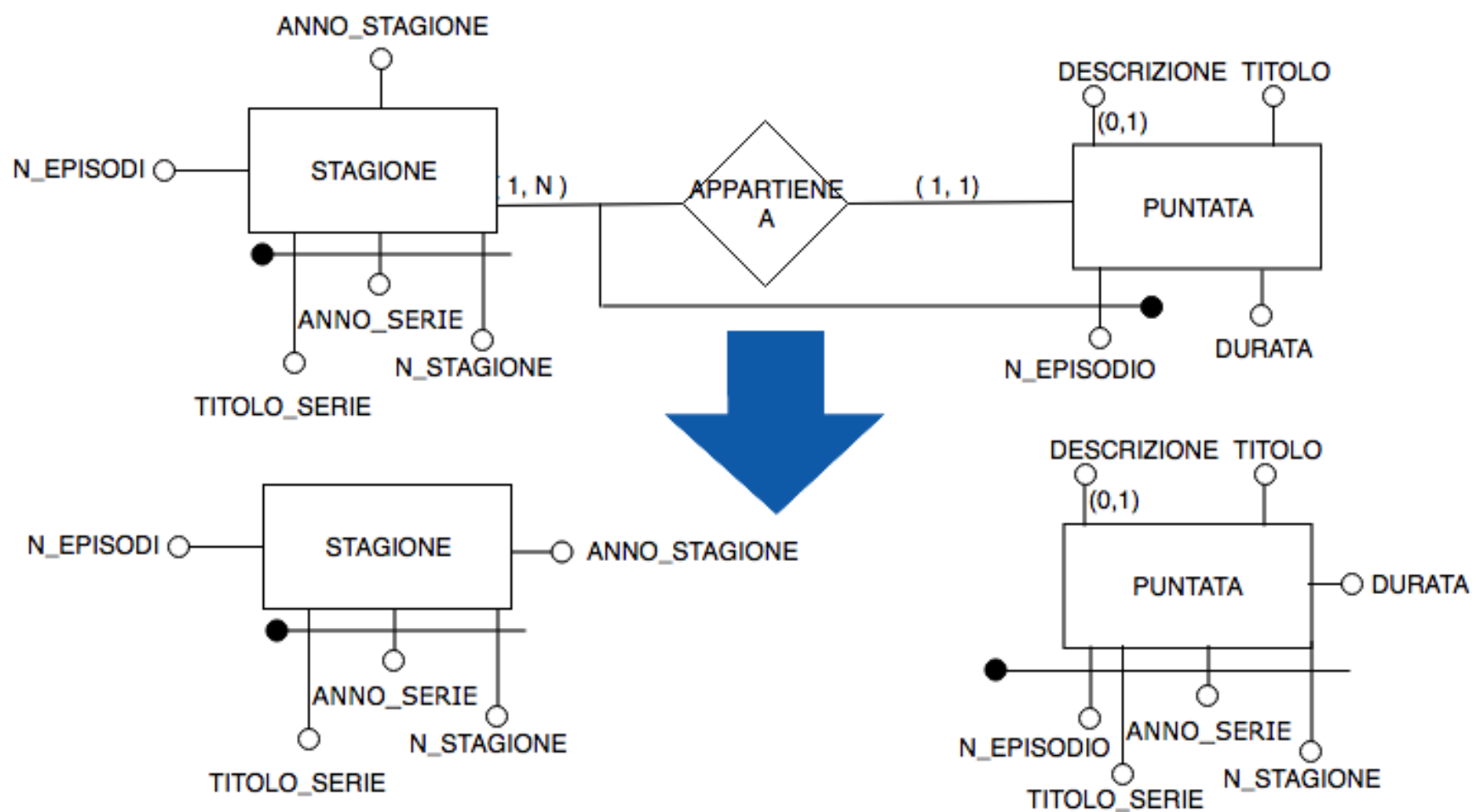
L'entità Stagione è identificata dal numero della stagione e dal titolo ed anno della serie tv



a cui si riferisce tramite l'associazione "Di una".

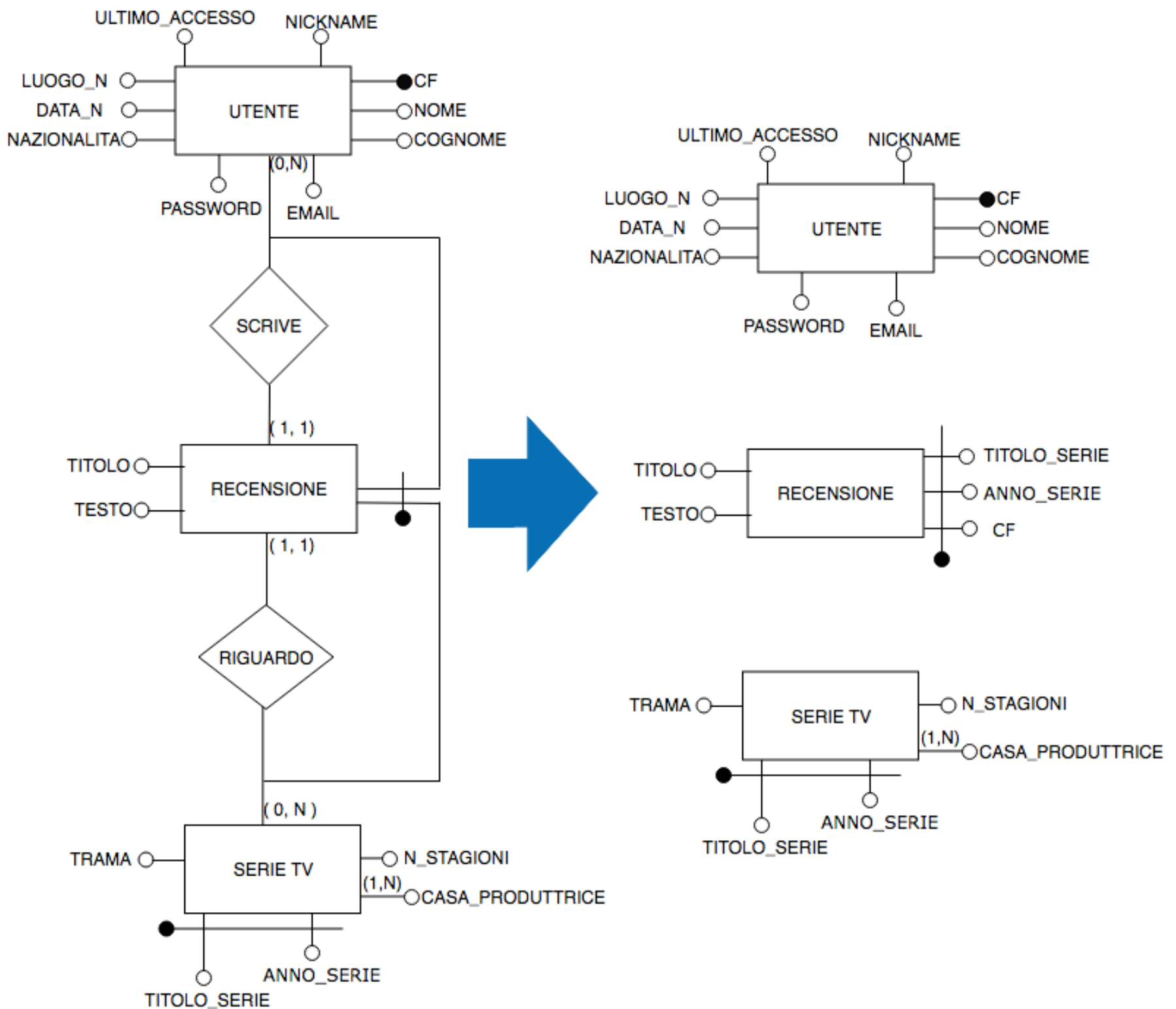
Una componente di identificazione esterna della entità "Stagione" da una entità "Serie tv" attraverso una associazione "Di una" comporta il trasporto della chiave primaria di "Serie tv" su "Stagione":

L'entità "*Puntata*" ha una componente di identificazione esterna dall'entità "*Stagione*".
L'entità Puntata è quindi identificata dal numero della puntata e dal numero della stagione



a cui si riferisce (oltre che al titolo e l'anno della serie tv a cui appartiene). Trasportiamo dunque la chiave primaria di "*Stagione*" su "*Puntata*".

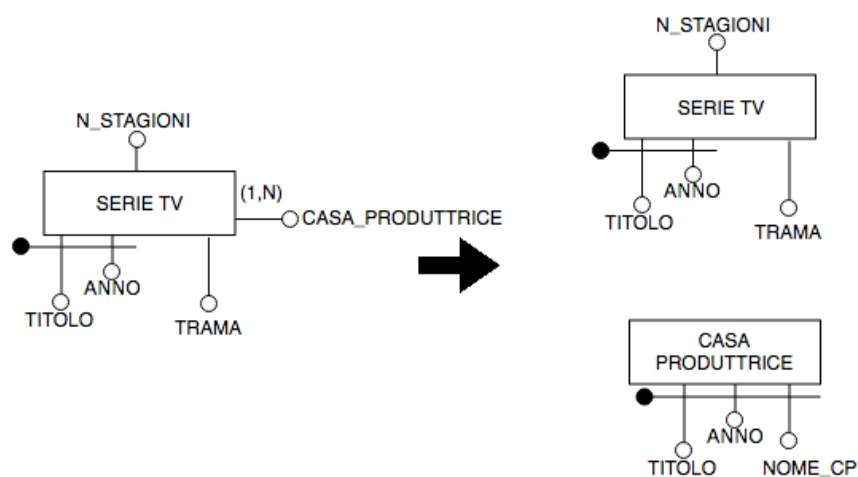
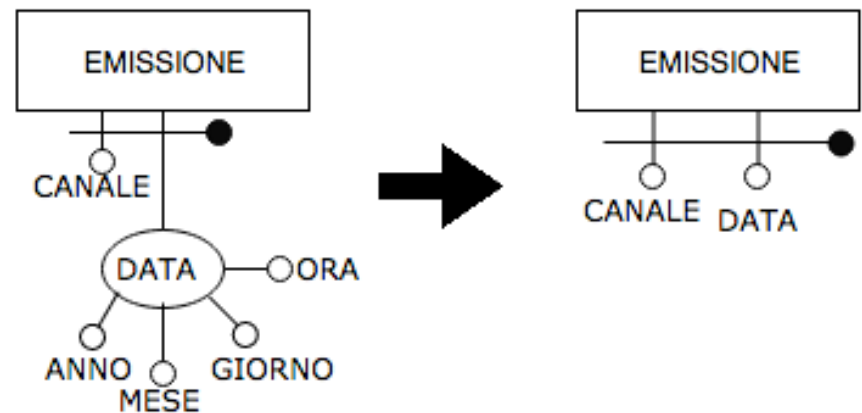
L'entità "Recensione" ha delle componenti di identificazioni esterne dalle entità "Utente" e "Serie tv". L'entità Recensione è quindi identificata dal codice fiscale dell'utente e dal titolo/anno della serie tv.



3 - TRASFORMAZIONE DEGLI ATTRIBUTI COMPOSTI O MULTIPLI

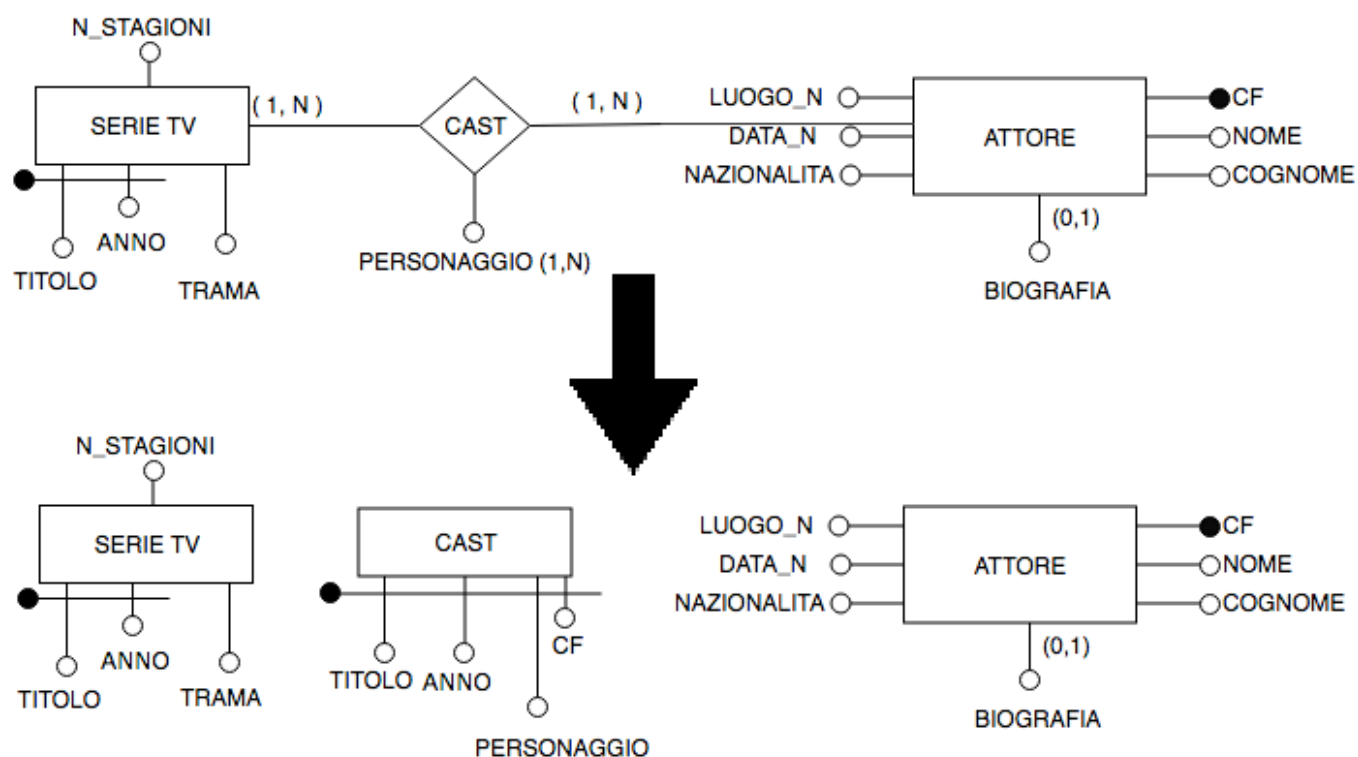
Siccome le relazioni non possono contenere attributi composti o ripetuti, bisogna trasformare gli attributi composti o in attributi atomici, o in entità.

Per l'entità "Emissione" abbiamo eliminato i componenti e considerato l'attributo come semplice.



L'entità "Serie TV" ha un attributo ripetuto, ovvero "Casa produttrice". Creiamo quindi un'entità a parte che contenga l'attributo e che si colleghi a "Serie TV".

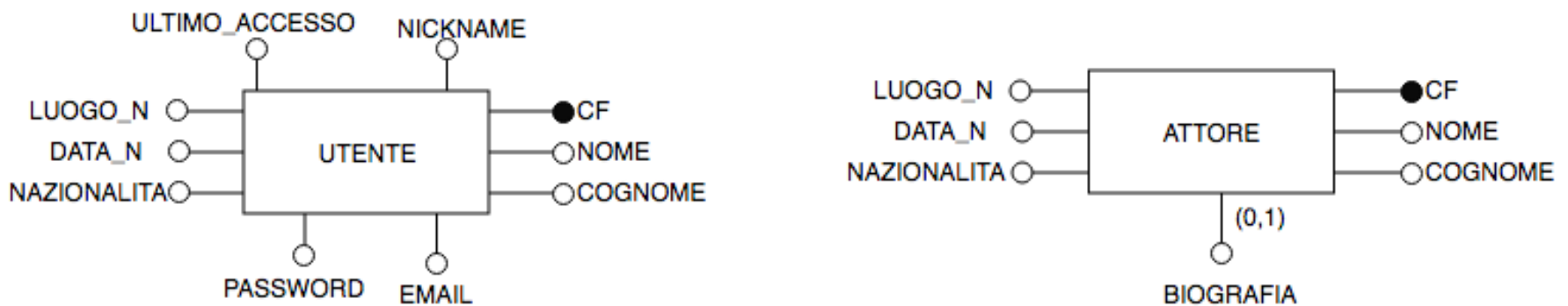
Un altro attributo ripetuto è "**Personaggi**" nell'associazione Cast. Cast non è incorporabile perchè la cardinalità è N:N. Potrei creare un'entità a parte per i personaggi, però non conviene in quanto non ci direbbe nulla in più.



Personaggio non è un attributo semplice, in quanto può avere più valori. Mi limito quindi a trasformare "**Cast**" in una entità e ad aggiungere Personaggio alla chiave.

4 - TRASFORMAZIONE DI ENTITÀ E ASSOCIAZIONI IN SCHEMI DI RELAZIONI

Si vogliono tradurre le entità ed associazioni in schemi relazionali.

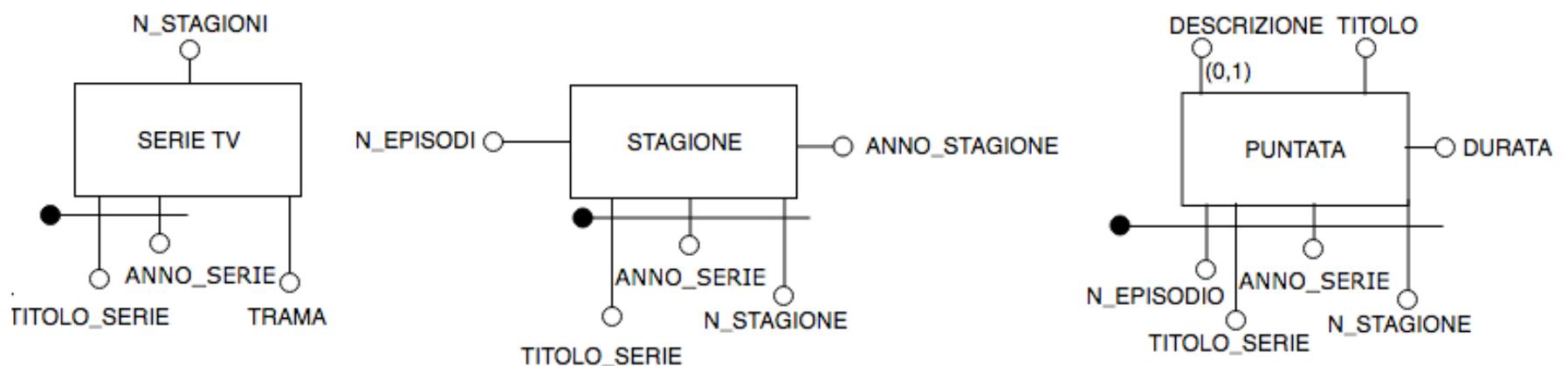


Utente (CF, Nickname, Nome, Cognome, Data_n, Nazionalita, Luogo_n, Email, Ultimo_accesso, Password)

AK: Nickname

AK: Email

Attore (CF, Nome, Cognome, Data_n, Nazionalita, Luogo_n, Biografia)



SerieTV (Titolo_Serie, Anno_Serie, N_Stagioni, Trama)

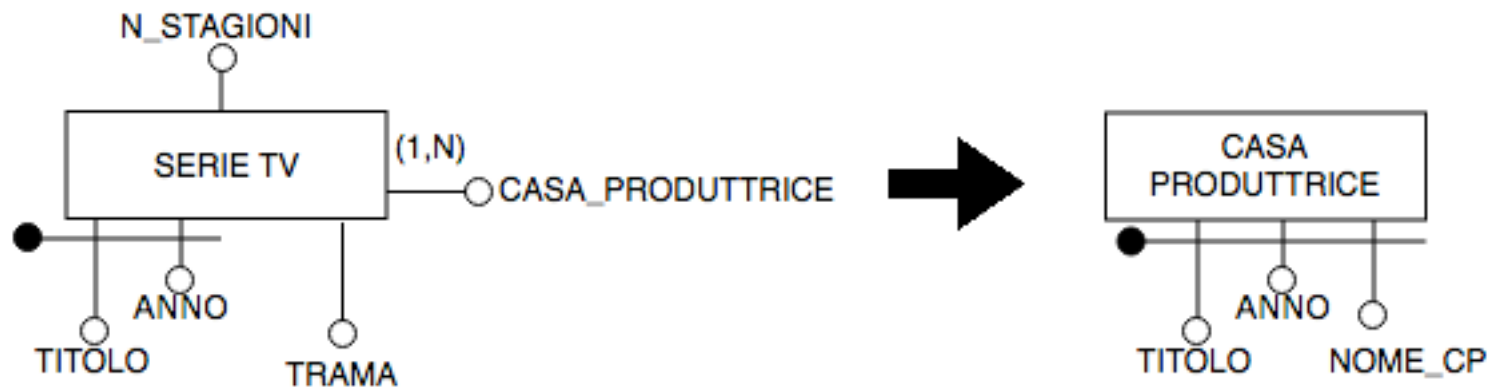
Stagione (N_Stagione, Titolo_Serie, Anno_Serie, Anno_Stagione, N_Episodi)

FK: Titolo_Serie, Anno_Serie REFERENCES SerieTV

Puntata (N_Episodio, N_Stagione, Titolo_Serie, Anno_Serie, Titolo_Puntata, Durata, Descrizione)

FK: Titolo_Serie, Anno_Serie REFERENCES SerieTV

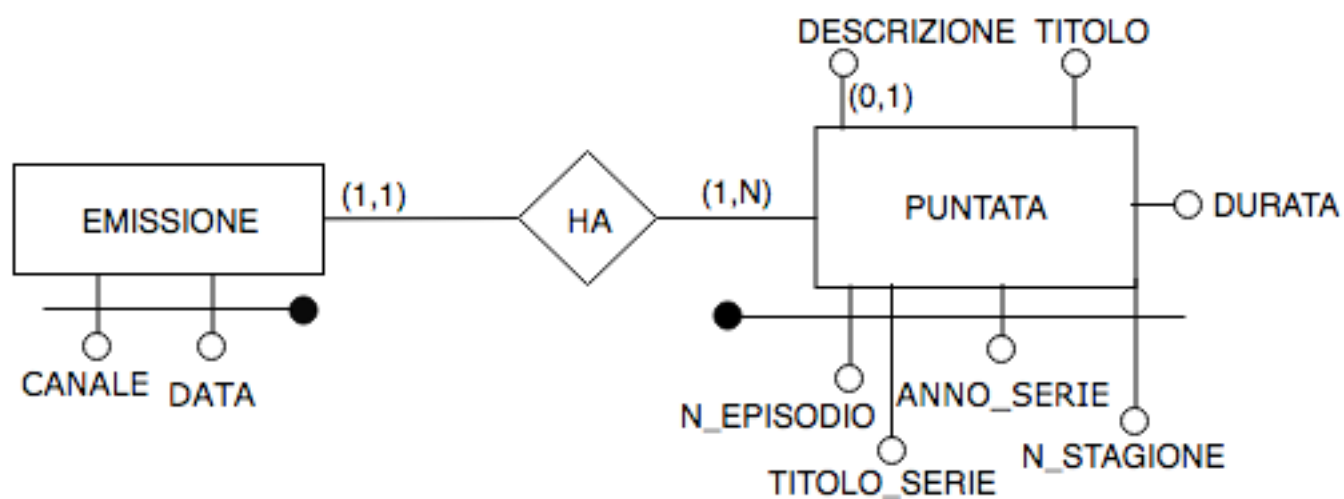
FK: N_Stagione REFERENCES Stagione



Casa_Produttrice (Nome, Titolo Serie, Anno Serie)

FK: Titolo_Serie, Anno_Serie REFERENCES SerieTV

- Le entità **Emissione** e **Puntata** sono legate da un'associazione N:1, per questo motivo è possibile accorpare l'associazione "Ha" all'entità **Emissione**. Il campo data è costituito da giorno, mese, anno e ora (formato timestamp). In un canale e in una certa data ci può essere una sola puntata. Una puntata può essere invece associata ad una o più emissioni (una puntata può essere quindi trasmessa su canali differenti).



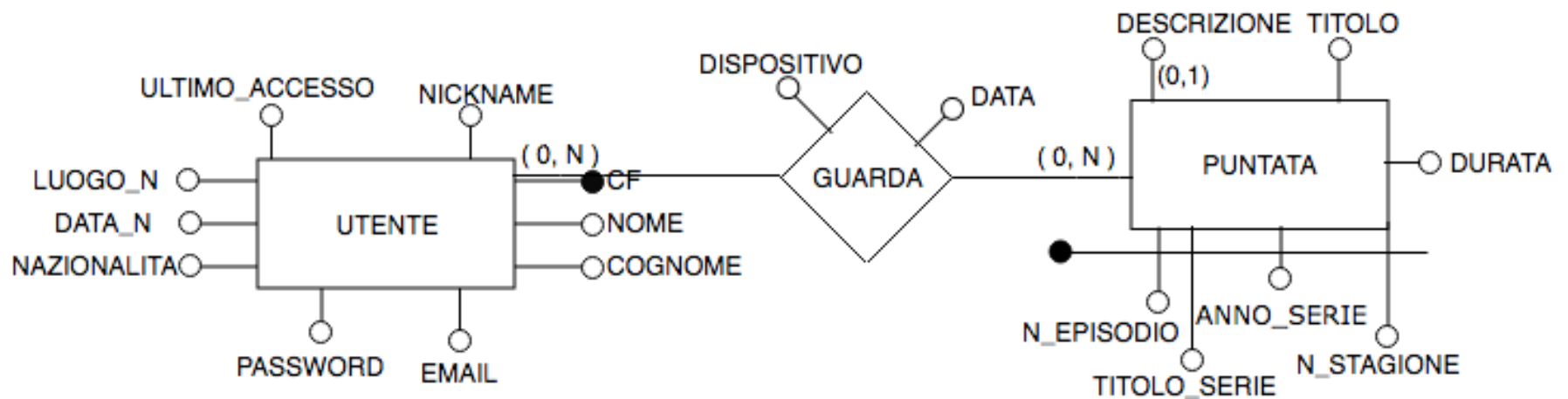
Emissione (Canale, Data, N_Episodio, N_Stagione, Titolo_Serie, Anno_Serie)

FK: N_Episodio REFERENCES Puntata

FK: N_Stagione REFERENCES Stagione

FK: Titolo_Serie, Anno_Serie REFERENCES SerieTV

- Le entità **Utente** e **Puntata** sono legate da un'associazione N:N, per questo motivo vengono tradotte con una relazione per ogni entità e una per l'associazione **Guarda**.



Guarda (CF, N_Episodio, N_Stagione, Titolo_Serie, Anno_Serie, Dispositivo, Data)

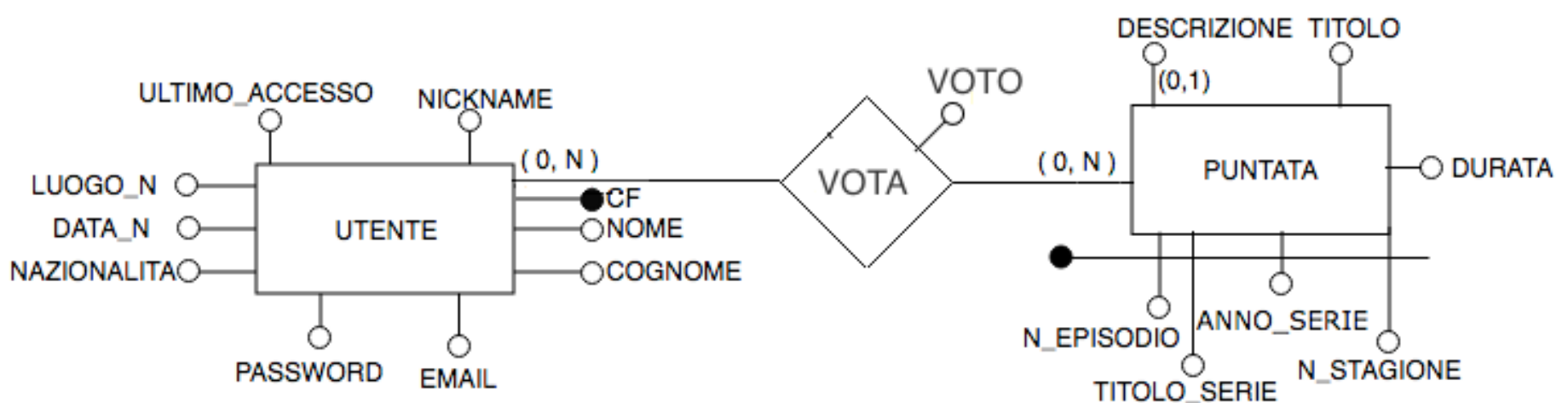
FK: CF REFERENCES Utente

FK: N_Episodio REFERENCES Puntata

FK: N_Stagione REFERENCES Stagione

FK: Titolo_Serie, Anno_Serie REFERENCES SerieTV

- Le entità **Utente** e **Puntata** sono legate da un'associazione N:N, per questo motivo vengono tradotte con una relazione per ogni entità e una per l'associazione **Vota**.



Vota (CF, N_Episodio, N_Stagione, Titolo_Serie, Anno_Serie, Voto)

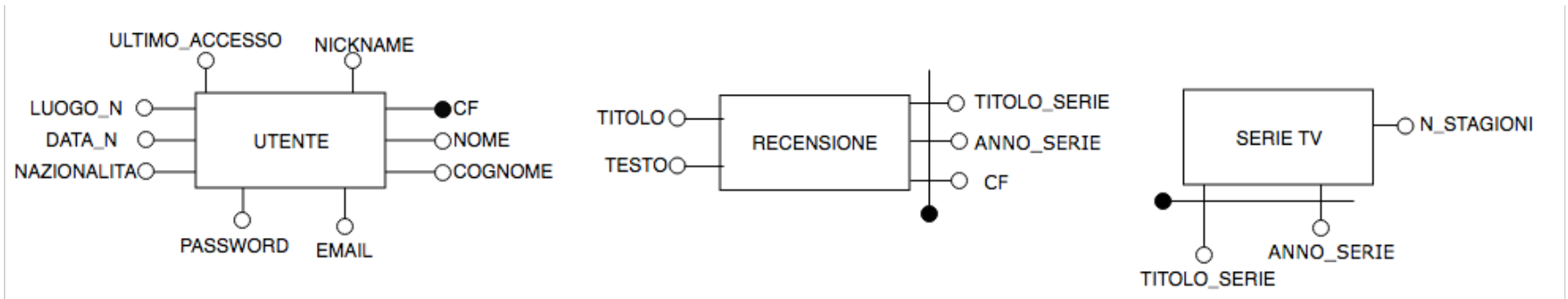
FK: CF REFERENCES Utente

FK: N_Episodio REFERENCES Puntata

FK: N_Stagione REFERENCES Stagione

FK: Titolo_Serie, Anno_Serie REFERENCES SerieTV

- Accorpo l'associazione "Scrive" e "Riguardo" in "**Recensione**".

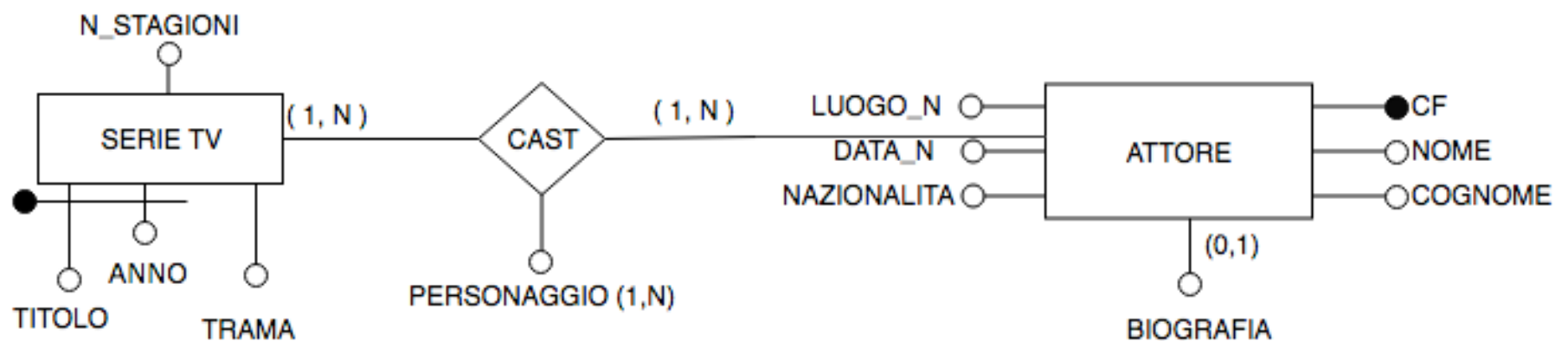


Recensione (CF, Titolo_Serie, Anno_Serie, Titolo, Testo)

FK: CF REFERENCES Utente

FK: Titolo_Serie, Anno_Serie REFERENCES serietv

- Le entità **Attore** e **SerieTV** sono legate da un'associazione N:N, per questo motivo vengono tradotte con una relazione per ogni entità e una per l'associazione **Cast**.

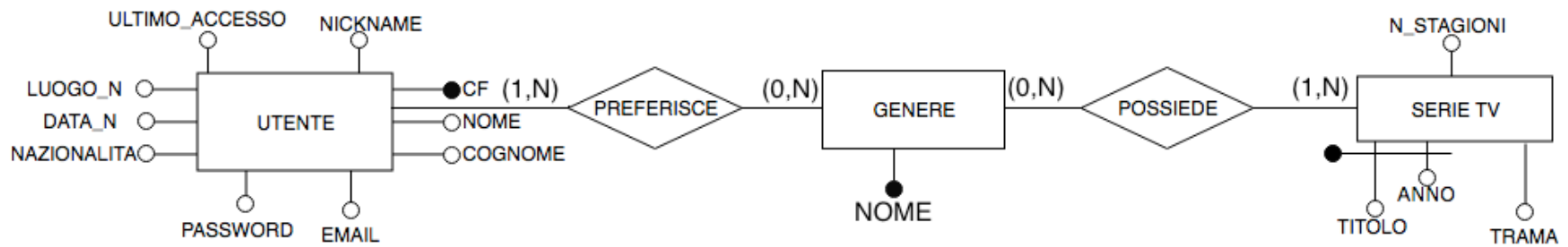


Cast (Titolo_Serie, Anno_Serie, CF Attore, Personaggio)

FK: Titolo_Serie, Anno_Serie REFERENCES SerieTV

FK: CF_Attore REFERENCES Attore

- Le entità **Utente** e **Genere**, così come le entità **Genere** e **SerieTV**, sono legate da



un'associazione N:N, per questo motivo vengono tradotte con una relazione per ogni entità e le associazioni **Preferisce** e **Possiede** vengono trasformate in entità.

Genere (Nome)

Utente_Genere (CF, Genere)

FK: CF REFERENCES Utente

FK: Genere REFERENCES Genere

Serie_Genere (Titolo, Anno, Genere)

FK: Titolo, Anno REFERENCES SerieTV

FK: Genere REFERENCES Genere

5 - VERIFICA DELLA NORMALIZZAZIONE

Dopo aver effettuato la verifica della normalizzazione non abbiamo ritenuto necessaria alcuna ulteriore modifica.

VINCOLI AGGIUNTIVI

Alcuni vincoli non possono essere gestiti direttamente dal database e per questo motivo si renderanno necessari appositi **check e trigger**. I vincoli da analizzare sono i seguenti:

1. L'attributo "n_stagioni" relativo al numero di stagioni per serie tv deve essere uguale al numero di tuple inserite in "Stagione" per quella serie tv.
2. L'attributo "n_episodi" relativo al numero di episodi per stagione deve essere uguale al numero di tuple inserite in "Puntata" per quella stagione.
3. Un utente può votare una puntata solo se l'ha vista (ovvero se è presente in "guarda").
4. La password dell'utente durante l'inserimento deve essere criptata.
5. Il nickname non può avere meno di 4 caratteri.
6. La password deve contenere almeno 6 caratteri.
7. La data di nascita dell'utente deve essere compresa tra la data corrente e una "limite" per poter essere inserita.
8. L'anno di una stagione deve essere maggiore o uguale dell'anno della serie tv a cui si riferisce.
9. I voti che l'utente può inserire devono essere compresi tra 1 e 10.
10. Controllo sul formato dell'email tramite espressione regolare.
11. Una puntata deve durare più di 0 minuti.
12. Per poter recensire una serie tv, l'utente deve aver visto più di un episodio di quella serie.
13. L'utente una puntata la può guardare solo in una data il cui anno è maggiore rispetto all'anno di uscita della stagione di quella serie TV.
14. La data di messa in onda di una puntata su un canale non può avvenire in anni precedenti a quelli dell'anno di uscita della stagione di quella serie TV.
15. L'utente può vedere una puntata su uno di questi dispositivi: TV, Computer, Tablet o Phone.
16. L'ultimo accesso all'applicazione dell'utente non può essere successivo alla data corrente.

QUERY DI CREAZIONE

-- Creazione tabella utente

```
CREATE TABLE utente
(
    cf                CHAR(16) PRIMARY KEY,
    nickname          VARCHAR(20) NOT NULL UNIQUE,
    nome              VARCHAR(30) NOT NULL,
    cognome           VARCHAR(30) NOT NULL,
    data_n            DATE NOT NULL CHECK (data_n BETWEEN DATE '1930-01-01' AND
current_date),
    nazionalita       VARCHAR(60) NOT NULL,
    luogo_n           VARCHAR(60) NOT NULL,
    email             VARCHAR(50) NOT NULL UNIQUE,
    ultimo_accesso    DATE DEFAULT NULL CHECK (ultimo_accesso <= current_date),
    password          VARCHAR(60) NOT NULL,
    CHECK (Length(nickname) >= 4),
    CHECK (Length(password) >= 6),
    CHECK(email ~* '^[A-Za-z0-9._%-]+@[A-Za-z0-9.-]+[.][A-Za-z]+$')
);
```

-- Creazione tabella attore

```
CREATE TABLE attore
(
    cf                CHAR(16) PRIMARY KEY,
    nome              VARCHAR(30) NOT NULL,
    cognome           VARCHAR(30) NOT NULL,
    data_n            DATE NOT NULL CHECK (data_n < CURRENT_DATE),
    nazionalita       VARCHAR(60) NOT NULL,
    luogo_n           VARCHAR(60) NOT NULL,
    biografia         VARCHAR(500)
);
```

-- Creazione tabella Serie TV

```
CREATE TABLE serietv
(
    titolo_serie     VARCHAR(60),
    anno_serie       INTEGER,
    n_stagioni       INT NOT NULL DEFAULT 0,
    trama            VARCHAR(500) NOT NULL,
    PRIMARY KEY (titolo_serie, anno_serie)
);
```

-- Creazione tabella sulle Stagioni

```
CREATE TABLE stagione
(
    n_stagione       INT,
    titolo_serie     VARCHAR(60),
    anno_serie       INTEGER,
    anno_stagione    INTEGER NOT NULL,
    n_episodi        INT NOT NULL DEFAULT 0,
    PRIMARY KEY (n_stagione, titolo_serie, anno_serie),
    FOREIGN KEY (titolo_serie, anno_serie) REFERENCES serietv ON DELETE CASCADE
ON UPDATE CASCADE,
    CHECK (anno_stagione >= anno_serie)
);
```

-- Creazione tabella Puntate

```
CREATE TABLE puntata
(
    n_episodio      INT,
    n_stagione      INT,
    titolo_serie    VARCHAR(60),
    anno_serie      INTEGER,
    titolo_puntata  VARCHAR(60) NOT NULL,
    durata          INTEGER NOT NULL CHECK (durata >= 0),
    descrizione     VARCHAR(500) DEFAULT NULL,
    PRIMARY KEY (n_episodio, n_stagione, titolo_serie, anno_serie),
    FOREIGN KEY (n_stagione, titolo_serie, anno_serie) REFERENCES stagione ON
    DELETE CASCADE ON UPDATE CASCADE
);
```

-- Creazione tabella casa produttrice

```
CREATE TABLE casa_produttrice
(
    titolo_serie    VARCHAR(60),
    anno_serie      INTEGER,
    nome            VARCHAR(30),
    PRIMARY KEY (nome, titolo_serie, anno_serie),
    FOREIGN KEY (titolo_serie, anno_serie) REFERENCES serietv ON DELETE CASCADE
    ON UPDATE CASCADE
);
```

-- Creazione tabella Emissione

```
CREATE TABLE emissione
(
    canale          VARCHAR(15),
    data            TIMESTAMP,
    n_episodio      INT NOT NULL,
    n_stagione      INT NOT NULL,
    titolo_serie    VARCHAR(60) NOT NULL,
    anno_serie      INTEGER NOT NULL,
    PRIMARY KEY (canale, data),
    FOREIGN KEY (n_episodio, n_stagione, titolo_serie, anno_serie) REFERENCES
    puntata ON DELETE CASCADE ON UPDATE CASCADE
);
```

-- Creazione tabella guarda

```
CREATE TABLE guarda
(
    cf              CHAR(16),
    n_episodio      INT,
    n_stagione      INT,
    titolo_serie    VARCHAR(60),
    anno_serie      INTEGER,
    dispositivo     VARCHAR(20) NOT NULL,
    data            DATE NOT NULL DEFAULT CURRENT_DATE,
    PRIMARY KEY (cf, n_episodio, n_stagione, titolo_serie, anno_serie),
    FOREIGN KEY (n_episodio, n_stagione, titolo_serie, anno_serie) REFERENCES
    puntata ON DELETE CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (cf) REFERENCES utente ON DELETE CASCADE ON UPDATE CASCADE,
    CHECK (dispositivo='Phone' OR dispositivo='Tablet' OR dispositivo=
    'Computer' OR dispositivo='TV')
);
```

-- Creazione tabella voti

```
CREATE TABLE vota
(
    cf            CHAR(16),
    voto          INTEGER CHECK (voto BETWEEN 1 AND 10) NOT NULL,
    n_episodio    INT,
    n_stagione     INT,
    titolo_serie  VARCHAR(60),
    anno_serie     INTEGER,
    PRIMARY KEY(cf, n_episodio, n_stagione, titolo_serie, anno_serie),
    FOREIGN KEY (cf) REFERENCES utente ON DELETE CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (n_episodio, n_stagione, titolo_serie, anno_serie) REFERENCES
    puntata ON DELETE CASCADE ON UPDATE CASCADE
);
```

-- Creazione tabella recensioni

```
CREATE TABLE recensione
(
    cf            CHAR(16),
    titolo_serie  VARCHAR(60),
    anno_serie     INTEGER,
    titolo        VARCHAR(30) NOT NULL,
    testo         VARCHAR(500) NOT NULL,
    PRIMARY KEY(cf, titolo_serie, anno_serie),
    FOREIGN KEY (cf) REFERENCES utente ON DELETE CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (titolo_serie, anno_serie) REFERENCES serietv ON DELETE CASCADE
    ON UPDATE CASCADE
);
```

-- Creazione tabella cast

```
CREATE TABLE casts
(
    titolo_serie  VARCHAR(60),
    anno_serie     INTEGER,
    cf_attore     CHAR(16),
    personaggio   VARCHAR(30) NOT NULL,
    PRIMARY KEY(titolo_serie, anno_serie, cf_attore, personaggio),
    FOREIGN KEY (cf_attore) REFERENCES attore(cf) ON DELETE CASCADE ON UPDATE
    CASCADE,
    FOREIGN KEY (titolo_serie, anno_serie) REFERENCES serietv ON DELETE CASCADE
    ON UPDATE CASCADE
);
```

-- Creazione tabella genere

```
CREATE TABLE genere
(
    nome VARCHAR(30) PRIMARY KEY NOT NULL UNIQUE
);
```

-- Creazione tabella "Utente preferisce Genere"

```
CREATE TABLE utente_genere
(
    cf            CHAR(16),
    genere        VARCHAR(30),
    PRIMARY KEY(cf, genere),
    FOREIGN KEY (cf) REFERENCES utente ON DELETE CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (genere) REFERENCES genere(nome) ON DELETE CASCADE ON UPDATE
    CASCADE
);
```

```
-- Creazione tabella "SerieTV possiede Genere"
CREATE TABLE serie_genere
(
    titolo VARCHAR(60),
    anno    INTEGER,
    genere  VARCHAR(30),
    PRIMARY KEY(titolo, anno, genere),
    FOREIGN KEY (titolo, anno) REFERENCES serietv ON DELETE CASCADE ON UPDATE
    CASCADE,
    FOREIGN KEY (genere) REFERENCES genere(nome) ON DELETE CASCADE ON UPDATE
    CASCADE
);
```

QUERY DI INSERIMENTO

```
-- Inserimento utenti.
-- Carico la funzione pgcrypto per criptare la password in md5 nel database.
CREATE extension pgcrypto;
INSERT INTO utente
VALUES      ( 'MATDES86B12I854K',
              'mattia.desiderio',
              'Mattia',
              'Desiderio',
              '1996-04-05',
              'Italia',
              'Modena',
              '230925@studenti.unimore.it',
              NULL,
              Crypt('miapassword', Gen_salt('md5')) );

INSERT INTO utente VALUES ('GIOGIB84H62C9930', 'gibs', 'Giorgia', 'Gibellini', '1997-07-05', 'I
talia', 'Modena', 'gibba@hotmail.it', NULL, crypt('123qwe', gen_salt('md5')));

-- Inserimento serie tv - stagione - puntata: The Flash
INSERT INTO serietv VALUES ('The Flash',
2014,0,'«Mi chiamo Barry Allen e sono l''uomo più veloce del mondo. Da piccolo vidi mia ma
dre morire per mano di qualcosa di impossibile, mio padre fu accusato del suo omicidio. Po
i un incidente trasformò me in qualcosa di impossibile... Io sono Flash!» ');

INSERT INTO stagione VALUES (1, 'The Flash', 2014, 2015, 0);
INSERT INTO stagione VALUES (2, 'The Flash', 2014, 2016, 0);
INSERT INTO stagione VALUES (3, 'The Flash', 2014, 2017, 0);
INSERT INTO stagione VALUES (4, 'The Flash', 2014, 2018, 0);

INSERT INTO puntata VALUES (1,1, 'The Flash', 2014, 'Una città di eroi', 43, NULL);
INSERT INTO puntata VALUES (2,1, 'The Flash', 2014, 'L''uomo più veloce del mondo', 43, NULL);
INSERT INTO puntata VALUES (3,1, 'The Flash', 2014, 'The Mist', 43, NULL);
INSERT INTO puntata VALUES (4,1, 'The Flash', 2014, 'Capitan Cold', 43, NULL);
INSERT INTO puntata VALUES (5,1, 'The Flash', 2014, 'Plastique', 43, NULL);
INSERT INTO puntata VALUES (6,1, 'The Flash', 2014, 'Girder', 43, NULL);
INSERT INTO puntata VALUES (7,1, 'The Flash', 2014, 'Blackout', 43, NULL);
INSERT INTO puntata VALUES (8,1, 'The Flash', 2014, 'Flash vs. Arrow', 43, NULL);
INSERT INTO puntata VALUES (9,1, 'The Flash', 2014, 'Firestorm', 43, NULL);
INSERT INTO puntata VALUES (10,1, 'The Flash', 2014, 'Heat Wave', 43, NULL);
INSERT INTO puntata VALUES (11,1, 'The Flash', 2014, 'Il pifferaio', 43, NULL);
INSERT INTO puntata VALUES (12,1, 'The Flash', 2014, 'Peek-a-Boo', 43, NULL);
INSERT INTO puntata VALUES (13,1, 'The Flash', 2014, 'L''uomo nucleare', 43, NULL);
INSERT INTO puntata VALUES (14,1, 'The Flash', 2014, 'Gorilla Grodd', 43, NULL);
INSERT INTO puntata VALUES (15,1, 'The Flash', 2014, 'Il mago del tempo', 43, NULL);
INSERT INTO puntata VALUES (16,1, 'The Flash', 2014, 'Lisa Snart', 43, NULL);
INSERT INTO puntata VALUES (17,1, 'The Flash', 2014, 'Trickster', 43, NULL);
INSERT INTO puntata VALUES (18,1, 'The Flash', 2014, 'Bug-Eyed Bandit', 43, NULL);
INSERT INTO puntata VALUES (19,1, 'The Flash', 2014, 'Everyman', 43, NULL);
INSERT INTO puntata VALUES (20,1, 'The Flash', 2014, 'La trappola', 43, NULL);
INSERT INTO puntata VALUES (21,1, 'The Flash', 2014, 'Grodd vive', 43, NULL);
INSERT INTO puntata VALUES (22,1, 'The Flash', 2014, 'I nemici', 43, NULL);
INSERT INTO puntata VALUES (23,1, 'The Flash', 2014, 'Wormhole', 43, NULL);
```

```

INSERT INTO puntata VALUES (1,2,'The Flash',2014,'L'uomo che ha salvato Central City',43,NULL);
INSERT INTO puntata VALUES (2,2,'The Flash',2014,'Flash dei due mondi',43,NULL);
INSERT INTO puntata VALUES (3,2,'The Flash',2014,'Famiglia di nemici',43,NULL);
INSERT INTO puntata VALUES (4,2,'The Flash',2014,'La furia di Firestorm',43,NULL);
INSERT INTO puntata VALUES (5,2,'The Flash',2014,'Dottor Light',43,NULL);
INSERT INTO puntata VALUES (6,2,'The Flash',2014,'Zoom',43,NULL);
INSERT INTO puntata VALUES (7,2,'The Flash',2014,'Il ritorno di Grodd',43,NULL);
INSERT INTO puntata VALUES (8,2,'The Flash',2014,'Leggende di oggi',43,NULL);
INSERT INTO puntata VALUES (9,2,'The Flash',2014,'Ricatto di Natale',43,NULL);
INSERT INTO puntata VALUES (10,2,'The Flash',2014,'Energia potenziale',43,NULL);
INSERT INTO puntata VALUES (11,2,'The Flash',2014,'Il ritorno dell'Anti-Flash',43,NULL);
INSERT INTO puntata VALUES (12,2,'The Flash',2014,'Tar-Pit',43,NULL);
INSERT INTO puntata VALUES (13,2,'The Flash',2014,'Benvenuti su Terra-Due',43,NULL);
INSERT INTO puntata VALUES (14,2,'The Flash',2014,'Fuga da Terra-Due',43,NULL);
INSERT INTO puntata VALUES (15,2,'The Flash',2014,'King Shark',43,NULL);
INSERT INTO puntata VALUES (16,2,'The Flash',2014,'Trajectory',43,NULL);
INSERT INTO puntata VALUES (17,2,'The Flash',2014,'Flash Back',43,NULL);
INSERT INTO puntata VALUES (18,2,'The Flash',2014,'Versus Zoom',43,NULL);
INSERT INTO puntata VALUES (19,2,'The Flash',2014,'Ritorno alle origini',43,NULL);
INSERT INTO puntata VALUES (20,2,'The Flash',2014,'Rupture',43,NULL);
INSERT INTO puntata VALUES (21,2,'The Flash',2014,'Il dinosauro in fuga',43,NULL);
INSERT INTO puntata VALUES (22,2,'The Flash',2014,'Invincibile',43,NULL);
INSERT INTO puntata VALUES (23,2,'The Flash',2014,'Sfida finale',43,NULL);

INSERT INTO puntata VALUES (1,3,'The Flash',2014,'Flashpoint',43,NULL);
INSERT INTO puntata VALUES (2,3,'The Flash',2014,'Dottor Alchemy',43,NULL);
INSERT INTO puntata VALUES (3,3,'The Flash',2014,'Magenta',43,NULL);
INSERT INTO puntata VALUES (4,3,'The Flash',2014,'I nuovi nemici',43,NULL);
INSERT INTO puntata VALUES (5,3,'The Flash',2014,'Il mostro',43,NULL);
INSERT INTO puntata VALUES (6,3,'The Flash',2014,'L'ombra',43,NULL);
INSERT INTO puntata VALUES (7,3,'The Flash',2014,'Killer Frost',43,NULL);
INSERT INTO puntata VALUES (8,3,'The Flash',2014,'Invasione!',43,NULL);
INSERT INTO puntata VALUES (9,3,'The Flash',2014,'Il presente',43,NULL);
INSERT INTO puntata VALUES (10,3,'The Flash',2014,'Plunder',43,NULL);
INSERT INTO puntata VALUES (11,3,'The Flash',2014,'Vivo o morto',43,NULL);
INSERT INTO puntata VALUES (12,3,'The Flash',2014,'Intoccabile',43,NULL);
INSERT INTO puntata VALUES (13,3,'The Flash',2014,'Attacco a Gorilla City',43,NULL);
INSERT INTO puntata VALUES (14,3,'The Flash',2014,'Attacco a Central City',43,NULL);
INSERT INTO puntata VALUES (15,3,'The Flash',2014,'La collera di Savitar',43,NULL);
INSERT INTO puntata VALUES (16,3,'The Flash',2014,'Dentro la Forza della Velocità',43,NULL);
INSERT INTO puntata VALUES (17,3,'The Flash',2014,'Duetto',43,NULL);
INSERT INTO puntata VALUES (18,3,'The Flash',2014,'Abra Kadabra',43,NULL);
INSERT INTO puntata VALUES (19,3,'The Flash',2014,'Flash eterno',43,NULL);
INSERT INTO puntata VALUES (20,3,'The Flash',2014,'So chi sei',43,NULL);
INSERT INTO puntata VALUES (21,3,'The Flash',2014,'Causa ed effetto',43,NULL);
INSERT INTO puntata VALUES (22,3,'The Flash',2014,'Infantino Street',43,NULL);
INSERT INTO puntata VALUES (23,3,'The Flash',2014,'La linea del traguardo',43,NULL);

INSERT INTO puntata VALUES (1,4,'The Flash',2014,'La rinascita di Flash',43,NULL);
INSERT INTO puntata VALUES (2,4,'The Flash',2014,'Segnali misti',43,NULL);
INSERT INTO puntata VALUES (3,4,'The Flash',2014,'La fortuna è donna',43,NULL);
INSERT INTO puntata VALUES (4,4,'The Flash',2014,'Plastic Man',43,NULL);
INSERT INTO puntata VALUES (5,4,'The Flash',2014,'Hashtag femminismo',43,NULL);
INSERT INTO puntata VALUES (6,4,'The Flash',2014,'Harry, ti presento Harry...',43,NULL);
INSERT INTO puntata VALUES (7,4,'The Flash',2014,'Il Pensatore',43,NULL);
INSERT INTO puntata VALUES (8,4,'The Flash',2014,'Crisi su Terra-X - III Parte',43,NULL);
INSERT INTO puntata VALUES (9,4,'The Flash',2014,'Non correre',43,NULL);
INSERT INTO puntata VALUES (10,4,'The Flash',2014,'Processo a Flash',43,NULL);
INSERT INTO puntata VALUES (11,4,'The Flash',2014,'L'uomo allungabile',43,NULL);
INSERT INTO puntata VALUES (12,4,'The Flash',2014,'Tesoro, mi si è ristretto il Team Flash',
43,NULL);
INSERT INTO puntata VALUES (13,4,'The Flash',2014,'La vera natura',43,NULL);
INSERT INTO puntata VALUES (14,4,'The Flash',2014,'Soggetto 9',43,NULL);
INSERT INTO puntata VALUES (15,4,'The Flash',2014,'Enter Flashtime',43,NULL);
INSERT INTO puntata VALUES (16,4,'The Flash',2014,'Corri, Iris, Corri',43,NULL);
INSERT INTO puntata VALUES (17,4,'The Flash',2014,'Null and Annoyed',43,NULL);
INSERT INTO puntata VALUES (18,4,'The Flash',2014,'Lose Yourself',43,NULL);

```



```

INSERT INTO puntata VALUES (19,4,'The Flash',2014,'Fury Rogue',
43,NULL);INSERT INTO puntata VALUES (20,4,'The Flash',2014,'Therefore She is',
43,NULL);INSERT INTO puntata VALUES (21,4,'The Flash',2014,'Harry e gli Harrison',
43,NULL);INSERT INTO puntata VALUES (22,4,'The Flash',2014,'Pensa velocemente',
43,NULL);INSERT INTO puntata VALUES (23,4,'The Flash',2014,'Noi siamo Flash',43,NULL);

-- Inserimento serie tv - stagione - puntata: Stranger Things
INSERT INTO serietv VALUES ('Stranger Things',
2016,0,'Succedono strane cose a Hawkins nell''Indiana, dove l''improvvisa scomparsa di un ragazzino porta al
la luce una coetanea dai poteri ultraterreni.057281');

INSERT INTO stagione VALUES (1,'Stranger Things',2016,2016,0);
INSERT INTO stagione VALUES (2,'Stranger Things',2016,2017,0);
INSERT INTO stagione VALUES (3,'Stranger Things',2016,2019,0);
INSERT INTO puntata VALUES (1,1,'Stranger Things',2016,'Capitolo primo: La scomparsa di Will Byers',
49,'Di ritorno da casa di un amico, il giovane Will vede qualcosa di terrificante. Nel vicino laboratorio go
vernativo si cela un oscuro segreto.');
```

INSERT INTO puntata VALUES (2,1,'Stranger Things',2016,'Capitolo due: La stramba di Maple Street',
55,'Lucas, Mike e Dustin provano a parlare con la ragazza ritrovata nel bosco. Hopper interroga Joyce su un'
'inquietante telefonata.');

INSERT INTO puntata VALUES (3,1,'Stranger Things',2016,'Capitolo tre: Luci natalizie',
52,'Nancy, sempre più preoccupata, cerca Barb e scopre che cosa ha fatto Jonathan. Joyce è convinta che Will
stia cercando di comunicare con lei.');

INSERT INTO puntata VALUES (4,1,'Stranger Things',2016,'Capitolo quattro: Il corpo',
50,'Rifiutando la morte del figlio, Joyce cerca di entrare in contatto con lui. I ragazzi sistemano Undici,
mentre Nancy e Jonathan siglano un''improbabile alleanza.');

INSERT INTO puntata VALUES (5,1,'Stranger Things',2016,'Capitolo cinque: La pulce e l''acrobata',
53,'Hopper entra nel laboratorio mentre Nancy e Jonathan affrontano la forza che ha preso Will. I ragazzi ch
iedono al signor Clarke come viaggiare in un''altra dimensione.');

INSERT INTO puntata VALUES (6,1,'Stranger Things',2016,'Capitolo sei: Il mostro',
47,'Jonathan, in preda all''agitazione, cerca Nancy al buio ignorando la presenza di Steve. Hopper e Joyce s
coprono la verità sugli esperimenti del laboratorio.');

INSERT INTO puntata VALUES (7,1,'Stranger Things',2016,'Capitolo sette: La vasca da bagno',
42,'Undici cerca di raggiungere Will, mentre Lucas avverte che "i cattivi stanno arrivando". Nancy e Jonatha
n mostrano alla polizia le riprese di Jonathan.');

INSERT INTO puntata VALUES (8,1,'Stranger Things',2016,'Capitolo otto: Il sottosopra',
55,'Il dottor Brenner trattiene Hopper e Joyce per interrogarli mentre i ragazzi aspettano con Undici in pal
estra. Nancy e Jonathan si preparano allo scontro a casa di Will.');

INSERT INTO puntata VALUES (1,2,'Stranger Things',2016,'Capitolo uno: Mad Max',
48,'Mentre la città si prepara per Halloween, un rivale da record porta scompiglio in sala giochi e uno scet
tico Hopper ispeziona un campo di zucche marce.');

INSERT INTO puntata VALUES (2,2,'Stranger Things',2016,'Capitolo due: Dolcetto o scherzetto, matto',
56,'Dopo la visione terribile di Will la notte di Halloween, Mike si chiede dove sia Undici. Nancy deve fare
i conti con la verità su Barb.');

INSERT INTO puntata VALUES (3,2,'Stranger Things',2016,'Capitolo tre: Il girino',
51,'Dustin adotta uno strano animaletto e Undici diventa sempre più impaziente. Animato da buone intenzioni,
Bob sprona Will ad affrontare le sue paure.');

INSERT INTO puntata VALUES (4,2,'Stranger Things',2016,'Capitolo quattro: Will il saggio',
46,'Mentre è malato, Will si apre con Joyce con conseguenze inquietanti. Hopper cerca di arrivare alla verit
à e Undici fa una scoperta sorprendente.');

INSERT INTO puntata VALUES (5,2,'Stranger Things',2016,'Capitolo cinque: Dig Dug',
58,'Nancy e Jonathan scambiano teorie del complotto con un nuovo alleato, mentre Undici cerca una persona de
l passato. "Bob il cervellone" affronta un difficile problema.');

INSERT INTO puntata VALUES (6,2,'Stranger Things',2016,'Capitolo sei: La spia',
51,'Il rapporto di Will con una forza oscura si rafforza, ma nessuno sa come fermarla. Intanto, Dustin e Ste
ve stringono un legame improbabile.');

INSERT INTO puntata VALUES (7,2,'Stranger Things',2016,'Capitolo sette: La sorella perduta',
45,'Alcune visioni paranormali avvicinano Undici a una banda di violenti reietti e a una ragazza rabbiosa da
l passato oscuro.');

INSERT INTO puntata VALUES (8,2,'Stranger Things',2016,'Capitolo otto: Il Mind Flayer',
47,'Un improbabile eroe si fa avanti quando i laboratori di Hawkins vengono chiusi, intrappolando Will e mol
te altre persone all''interno.');

INSERT INTO puntata VALUES (9,2,'Stranger Things',2016,'Capitolo nove: La porta',
62,'Undici si organizza per portare a termine quanto ha cominciato, mentre i sopravvissuti passano all''azio
ne contro la forza mostruosa che ha catturato Will.');

```

-- Inserimento serie tv - stagione - puntata: Tredici
INSERT INTO serietv VALUES ('Tredici',
2017,0,'Dopo lo strano suicidio di un''adolescente, un suo compagno di classe riceve alcune audiocassette ch
e rivelano il mistero dietro al tragico gesto.');
```

```

INSERT INTO stagione VALUES (1,'Tredici',2017,2017,0);
INSERT INTO stagione VALUES (2,'Tredici',2017,2018,0);
```

```

INSERT INTO puntata VALUES (1,1,'Tredici',2017,'Cassetta 1, parte A',
54,'Mentre la scuola è in lutto per la morte di Hannah Baker, il suo amico Clay riceve alcune audiocassette
con dei messaggi registrati da Hannah prima del suicidio.');
```

```

INSERT INTO puntata VALUES (2,1,'Tredici',2017,'Cassetta 1, parte B',
52,'Hannah diventa amica di Jessica e Alex, due altri studenti appena arrivati. Justin non si presenta a scu
ola e la madre di Hannah fa una scoperta sconcertante.');
```

```

INSERT INTO puntata VALUES (3,1,'Tredici',2017,'Cassetta 2, parte A',
57,'La gelosia e i pettegolezzi rovinano la vita ad Hannah. Mentre Alex affronta i propri rimpianti, i ragaz
zi spingono Clay a ubriacarsi.');
```

```

INSERT INTO puntata VALUES (4,1,'Tredici',2017,'Cassetta 2, parte B',
57,'Hannah pensa che qualcuno la stia perseguitando e prepara una trappola. La madre di Hannah affronta il p
reside e Clay reagisce contro chi ha fatto del male ad Hannah.');
```

```

INSERT INTO puntata VALUES (5,1,'Tredici',2017,'Cassetta 3, parte A',
59,'Hannah e Clay si trovano al ballo scolastico, ma un volgare pettegolezzo rovina l''atmosfera. Clay porta
Courtney sulla tomba di Hannah.');
```

```

INSERT INTO puntata VALUES (6,1,'Tredici',2017,'Cassetta 3, parte B',
52,'L''appuntamento di Hannah per San Valentino non va come previsto. Alex viene coinvolto in una rissa a sc
uola e deve affrontare il consiglio disciplinare.');
```

```

INSERT INTO puntata VALUES (7,1,'Tredici',2017,'Cassetta 4, parte A',
54,'Durante un progetto di classe Hannah viene boicottata ancora una volta. Gli incubi riguardanti Hannah co
minciano a tormentare Clay anche di giorno.');
```

```

INSERT INTO puntata VALUES (8,1,'Tredici',2017,'Cassetta 4, parte B',
54,'Hannah viene ispirata da una lettura di poesie e decide di aprire il suo cuore. Tony parla con Clay dell
a notte in cui Hannah è morta.');
```

```

INSERT INTO puntata VALUES (9,1,'Tredici',2017,'Cassetta 5, parte A',
59,'Durante una festa d'estate Hannah assiste a un evento traumatico. Clay cerca di parlare con Justin, men
tre Marcus lo avverte che il peggio deve ancora arrivare.');
```

```

INSERT INTO puntata VALUES (10,1,'Tredici',2017,'Cassetta 5, parte B',
50,'Hannah è sconvolta e trova un passaggio per tornare a casa. Il comportamento di Jessica è sempre più ins
tabile e Clay scopre la verità nascosta dietro a una disgrazia.');
```

```

INSERT INTO puntata VALUES (11,1,'Tredici',2017,'Cassetta 6, parte A',
54,'Clay e Hannah diventano più intimi. Mentre Clay trascorre una notte straziante ad ascoltare la sua audio
cassetta con Tony, la tensione esplode a casa di Bryce.');
```

```

INSERT INTO puntata VALUES (12,1,'Tredici',2017,'Cassetta 6, parte B',
62,'Dopo aver litigato con i suoi genitori, Hannah finisce a una festa. Gli studenti ricevono dei mandati di
comparizione e Justin è combattuto tra alleanze contrapposte.');
```

```

INSERT INTO puntata VALUES (13,1,'Tredici',2017,'Cassetta 7, parte A',
60,'Hannah chiede aiuto al signor Porter, lo psicologo scolastico. Clay ascolta il nuovo nastro per Tony e v
aluta i prossimi passi da fare.');
```

```

INSERT INTO puntata VALUES (1,2,'Tredici',2017,'La prima Polaroid',
60,'Cinque mesi dopo la morte di Hannah, la causa contro la Liberty approda in aula e Tyler è il primo a tes
timoniare. Clay trova una foto inquietante nel suo armadietto.');
```

```

INSERT INTO puntata VALUES (2,2,'Tredici',2017,'Due ragazze e un bacio',
60,'Courtney prende una decisione coraggiosa. Il ricordo di Hannah rovina il rapporto tra Clay e Skye. L''uf
ficio dell''annuario viene devastato.');
```

```

INSERT INTO puntata VALUES (3,2,'Tredici',2017,'L''ubriacona',
57,'Spaventati dalle minacce ricevute, Alex e Clay spingono Jessica a testimoniare contro Bryce. Clay chiede
aiuto a Tony per trovare un potenziale testimone.');
```

```

INSERT INTO puntata VALUES (4,2,'Tredici',2017,'La seconda Polaroid',
57,'Clay nasconde un segreto ai genitori. Preoccupato per la sua reputazione, Marcus mente in aula. Jessica
e Alex saltano insieme la scuola.');
```

```

INSERT INTO puntata VALUES (5,2,'Tredici',2017,'La macchina tracciarighe',
57,'Tyler e Cyrus accettano la loro condizione di outsider. Alex cerca disperatamente di ricordare. Ryan e l
a signora Baker analizzano le poesie di Hannah in cerca d''indizi.');
```

```

INSERT INTO puntata VALUES (6,2,'Tredici',2017,'Il sorriso alla fine del molo',
58,'Clay cerca di capire dove sono state scattate le Polaroid. Justin si presenta a scuola. La rivelazione d
i Zach su Hannah coglie tutti di sorpresa.');
```

```

INSERT INTO puntata VALUES (7,2,'Tredici',2017,'La terza Polaroid',
56,'Alex esplode contro Bryce. Marcus viene ricattato. Clay è chiamato a testimoniare sulla notte passata co
n Hannah. Jessica ha un flashback.');
```

```

INSERT INTO puntata VALUES (8,2,'Tredici',2017,'La bambina',
57,'Quando i genitori di Hannah sono sotto investigazione, Jessica contatta la signora Baker. Justin lotta p
er restare pulito. Clay si riavvicina a Skye.');
```

```

INSERT INTO puntata VALUES (9,2,'Tredici',2017,'La pagina mancante',
57,'Tyler esce con la sorella di Cyrus. Chloe affronta Bryce per le cassette. Jess si chiede se sia pronta p
er uscire con un ragazzo. Il professor Porter lotta con i rimorsi.');
```



```

INSERT INTO puntata VALUES (10,2,'Tredici',2017,'Un bel sorriso, stronze',
55,'Tony si apre sul suo passato. Sheri escogita un piano per entrare nel club. Il professor Porter cerca di
aiutare Justin. Zach e Bryce si scontrano.');
```

```

INSERT INTO puntata VALUES (11,2,'Tredici',2017,'Bryce e Chloe',
58,'Dopo la testimonianza di Bryce, a scuola esplode il caos. Jessica racconta a Chloe del club. I ricordi f
iniscono per travolgere Alex.');
```

```

INSERT INTO puntata VALUES (12,2,'Tredici',2017,'La scatola di Polaroid',
56,'Le minacce contro Clay e gli altri s'intensificano. Tyler affronta provvedimenti disciplinari. Justin te
stimonia contro Bryce, mettendo a rischio il suo futuro.');
```

```

INSERT INTO puntata VALUES (13,2,'Tredici',2017,'Addio',
70,'Un mese dopo, le persone legate ad Hannah le rendono omaggio, confortandosi a vicenda. Intanto, una brut
ale aggressione spinge uno studente al limite.');
```

```

-- Inserimento serie tv - stagione - puntata: Wayward Pines
INSERT INTO serietv VALUES ('Wayward Pines',
2015,0,'Mistery ambientato in una perfetta città americana. Adesso immaginate di non poter lasciare questo p
osto. Mai. Nessun contatto con l'esterno, nessuna via di fuga. Questa è Wayward Pines.');
```

```

INSERT INTO stagione VALUES (1,'Wayward Pines',2015,2015,0);
INSERT INTO stagione VALUES (2,'Wayward Pines',2015,2016,0);
```

```

INSERT INTO puntata VALUES (1,1,'Wayward Pines',2015,'Dove il paradiso è di casa',45,NULL);
INSERT INTO puntata VALUES (2,1,'Wayward Pines',2015,'Un piano di fuga',45,NULL);
INSERT INTO puntata VALUES (3,1,'Wayward Pines',2015,'La nostra città, la nostra legge',45,NULL);
INSERT INTO puntata VALUES (4,1,'Wayward Pines',2015,'Una nuova vita',45,NULL);
INSERT INTO puntata VALUES (5,1,'Wayward Pines',2015,'La verità',45,NULL);
INSERT INTO puntata VALUES (6,1,'Wayward Pines',2015,'Scelte',45,NULL);
INSERT INTO puntata VALUES (7,1,'Wayward Pines',2015,'Tradimento',45,NULL);
INSERT INTO puntata VALUES (8,1,'Wayward Pines',2015,'Il posto più bello sulla Terra',45,NULL);
INSERT INTO puntata VALUES (9,1,'Wayward Pines',2015,'Esecuzione',45,NULL);
INSERT INTO puntata VALUES (10,1,'Wayward Pines',2015,'Un nuovo inizio',45,NULL);
```

```

INSERT INTO puntata VALUES (1,2,'Wayward Pines',2015,'Linee nemiche',45,NULL);
INSERT INTO puntata VALUES (2,2,'Wayward Pines',2015,'Contro ogni regola',45,NULL);
INSERT INTO puntata VALUES (3,2,'Wayward Pines',2015,'C'era una volta a Wayward Pines',45,NULL);
INSERT INTO puntata VALUES (4,2,'Wayward Pines',2015,'Strategia di fuga',45,NULL);
INSERT INTO puntata VALUES (5,2,'Wayward Pines',2015,'Utopia',45,NULL);
INSERT INTO puntata VALUES (6,2,'Wayward Pines',2015,'L'imboscata',45,NULL);
INSERT INTO puntata VALUES (7,2,'Wayward Pines',2015,'Ogni anno, per duemila anni',45,NULL);
INSERT INTO puntata VALUES (8,2,'Wayward Pines',2015,'Il giorno del giudizio',45,NULL);
INSERT INTO puntata VALUES (9,2,'Wayward Pines',2015,'I prescelti',45,NULL);
INSERT INTO puntata VALUES (10,2,'Wayward Pines',2015,'La favola della buonanotte',45,NULL);
```

```

-- Inserimento case produttrici
INSERT INTO casa_produttrice VALUES ('Tredici',2017,'July Moon Productions');
INSERT INTO casa_produttrice VALUES ('Tredici',2017,'Kicked to the Curb Productions');
INSERT INTO casa_produttrice VALUES ('Tredici',2017,'Anonymous Content');
INSERT INTO casa_produttrice VALUES ('Tredici',2017,'Paramount Television');
INSERT INTO casa_produttrice VALUES ('Wayward Pines',2015,'FX Productions');
INSERT INTO casa_produttrice VALUES ('Stranger Things',2016,'Camp Hero Productions');
INSERT INTO casa_produttrice VALUES ('Stranger Things',2016,'21 Laps Entertainment');
INSERT INTO casa_produttrice VALUES ('Stranger Things',2016,'Monkey Massacre');
INSERT INTO casa_produttrice VALUES ('The Flash',2014,'Warner Bros. Television');
INSERT INTO casa_produttrice VALUES ('The Flash',2014,'DC Comics');
```

```

-- Inserimento emissione per ogni puntata (parziale)
INSERT INTO emissione VALUES ('Premium Action','2018-02-06 21:15:00',1,4,'The Flash',2014);
INSERT INTO emissione VALUES ('Premium Action','2018-02-13 21:15:00',2,4,'The Flash',2014);
INSERT INTO emissione VALUES ('Premium Action','2018-02-20 21:15:00',3,4,'The Flash',2014);
INSERT INTO emissione VALUES ('Premium Action','2018-02-27 21:15:00',4,4,'The Flash',2014);
INSERT INTO emissione VALUES ('Premium Action','2018-03-06 21:15:00',5,4,'The Flash',2014);
INSERT INTO emissione VALUES ('Premium Action','2018-03-13 21:15:00',6,4,'The Flash',2014);
INSERT INTO emissione VALUES ('Premium Action','2018-03-20 21:15:00',7,4,'The Flash',2014);
INSERT INTO emissione VALUES ('Premium Action','2018-03-27 21:15:00',8,4,'The Flash',2014);
INSERT INTO emissione VALUES ('Premium Action','2018-04-03 21:15:00',9,4,'The Flash',2014);
INSERT INTO emissione VALUES ('Premium Action','2018-04-10 21:15:00',10,4,'The Flash',2014);
INSERT INTO emissione VALUES ('Premium Action','2018-04-17 21:15:00',11,4,'The Flash',2014);
INSERT INTO emissione VALUES ('Premium Action','2018-04-24 21:15:00',12,4,'The Flash',2014);
INSERT INTO emissione VALUES ('Premium Action','2018-06-05 21:15:00',13,4,'The Flash',2014);
INSERT INTO emissione VALUES ('Premium Action','2018-06-12 21:15:00',14,4,'The Flash',2014);
INSERT INTO emissione VALUES ('Premium Action','2018-06-19 21:15:00',15,4,'The Flash',2014);
```

```

INSERT INTO emissione VALUES ('Premium Action', '2018-06-26 21:15:00', 16, 4, 'The Flash', 2014);
INSERT INTO emissione VALUES ('Premium Action', '2018-07-03 21:15:00', 17, 4, 'The Flash', 2014);
INSERT INTO emissione VALUES ('Premium Action', '2018-07-10 21:15:00', 18, 4, 'The Flash', 2014);
INSERT INTO emissione VALUES ('Premium Action', '2018-07-17 21:15:00', 19, 4, 'The Flash', 2014);
INSERT INTO emissione VALUES ('Premium Action', '2018-07-24 21:15:00', 20, 4, 'The Flash', 2014);
INSERT INTO emissione VALUES ('Premium Action', '2018-07-31 21:15:00', 21, 4, 'The Flash', 2014);
INSERT INTO emissione VALUES ('Premium Action', '2018-08-07 21:15:00', 22, 4, 'The Flash', 2014);
INSERT INTO emissione VALUES ('Premium Action', '2018-08-14 21:15:00', 23, 4, 'The Flash', 2014);

INSERT INTO emissione VALUES ('Fox', '2015-05-14 20:30:00', 1, 1, 'Wayward Pines', 2015);
INSERT INTO emissione VALUES ('Fox', '2015-05-21 20:30:00', 2, 1, 'Wayward Pines', 2015);
INSERT INTO emissione VALUES ('Fox', '2015-05-28 20:30:00', 3, 1, 'Wayward Pines', 2015);
INSERT INTO emissione VALUES ('Fox', '2015-06-05 20:30:00', 4, 1, 'Wayward Pines', 2015);
INSERT INTO emissione VALUES ('Fox', '2015-06-11 20:30:00', 5, 1, 'Wayward Pines', 2015);
INSERT INTO emissione VALUES ('Fox', '2015-06-25 20:30:00', 6, 1, 'Wayward Pines', 2015);
INSERT INTO emissione VALUES ('Fox', '2015-07-02 20:30:00', 7, 1, 'Wayward Pines', 2015);
INSERT INTO emissione VALUES ('Fox', '2015-07-09 20:30:00', 8, 1, 'Wayward Pines', 2015);
INSERT INTO emissione VALUES ('Fox', '2015-07-16 20:30:00', 9, 1, 'Wayward Pines', 2015);
INSERT INTO emissione VALUES ('Fox', '2015-07-23 20:30:00', 10, 1, 'Wayward Pines', 2015);
INSERT INTO emissione VALUES ('Fox', '2016-08-29 20:30:00', 1, 2, 'Wayward Pines', 2015);
INSERT INTO emissione VALUES ('Fox', '2016-08-29 21:30:00', 2, 2, 'Wayward Pines', 2015);
INSERT INTO emissione VALUES ('Fox', '2016-09-05 20:30:00', 3, 2, 'Wayward Pines', 2015);
INSERT INTO emissione VALUES ('Fox', '2016-09-05 21:30:00', 4, 2, 'Wayward Pines', 2015);
INSERT INTO emissione VALUES ('Fox', '2016-09-12 20:30:00', 5, 2, 'Wayward Pines', 2015);
INSERT INTO emissione VALUES ('Fox', '2016-09-19 20:30:00', 6, 2, 'Wayward Pines', 2015);
INSERT INTO emissione VALUES ('Fox', '2016-09-26 20:30:00', 7, 2, 'Wayward Pines', 2015);
INSERT INTO emissione VALUES ('Fox', '2016-10-03 20:30:00', 8, 2, 'Wayward Pines', 2015);
INSERT INTO emissione VALUES ('Fox', '2016-10-10 20:30:00', 9, 2, 'Wayward Pines', 2015);
INSERT INTO emissione VALUES ('Fox', '2016-10-17 20:30:00', 10, 2, 'Wayward Pines', 2015);
INSERT INTO emissione VALUES ('Netflix', '2017-10-27 00:00:00', 1, 2, 'Stranger Things', 2016);
INSERT INTO emissione VALUES ('Netflix', '2018-05-18 00:00:00', 1, 2, 'Tredici', 2017);

-- Inserimento generi
INSERT INTO genere VALUES ('Animazione');
INSERT INTO genere VALUES ('Avventura');
INSERT INTO genere VALUES ('Azione');
INSERT INTO genere VALUES ('Biografico');
INSERT INTO genere VALUES ('Comico');
INSERT INTO genere VALUES ('Commedia');
INSERT INTO genere VALUES ('Documentari');
INSERT INTO genere VALUES ('Drammatico');
INSERT INTO genere VALUES ('Fantascienza');
INSERT INTO genere VALUES ('Fantasy');
INSERT INTO genere VALUES ('Horror');
INSERT INTO genere VALUES ('Thriller');
INSERT INTO genere VALUES ('Western');
INSERT INTO genere VALUES ('Supereroi');
INSERT INTO genere VALUES ('Giallo');
INSERT INTO genere VALUES ('Adolescenziale');

-- Assegnamento generi alle serie tv
INSERT INTO serie_genere VALUES ('The Flash', 2014, 'Supereroi');
INSERT INTO serie_genere VALUES ('The Flash', 2014, 'Azione');
INSERT INTO serie_genere VALUES ('The Flash', 2014, 'Drammatico');
INSERT INTO serie_genere VALUES ('The Flash', 2014, 'Avventura');
INSERT INTO serie_genere VALUES ('Stranger Things', 2016, 'Fantascienza');
INSERT INTO serie_genere VALUES ('Stranger Things', 2016, 'Drammatico');
INSERT INTO serie_genere VALUES ('Stranger Things', 2016, 'Avventura');
INSERT INTO serie_genere VALUES ('Stranger Things', 2016, 'Thriller');
INSERT INTO serie_genere VALUES ('Tredici', 2017, 'Giallo');
INSERT INTO serie_genere VALUES ('Tredici', 2017, 'Drammatico');
INSERT INTO serie_genere VALUES ('Tredici', 2017, 'Thriller');
INSERT INTO serie_genere VALUES ('Tredici', 2017, 'Adolescenziale');
INSERT INTO serie_genere VALUES ('Wayward Pines', 2015, 'Thriller');
INSERT INTO serie_genere VALUES ('Wayward Pines', 2015, 'Fantascienza');
INSERT INTO serie_genere VALUES ('Wayward Pines', 2015, 'Giallo');

-- Lista preferenze generi per gli utenti
INSERT INTO utente_genere VALUES ('MATDES86B12I854K', 'Supereroi');
INSERT INTO utente_genere VALUES ('MATDES86B12I854K', 'Azione');
INSERT INTO utente_genere VALUES ('MATDES86B12I854K', 'Avventura');
INSERT INTO utente_genere VALUES ('GIOGIB84H62C9930', 'Fantasy');
INSERT INTO utente_genere VALUES ('GIOGIB84H62C9930', 'Drammatico');

```

```

-- Utente guarda una puntata
INSERT INTO guarda VALUES ('MATDES86B12I854K', 1, 1, 'The Flash', 2014, 'TV', '2015-05-04');
INSERT INTO guarda VALUES ('MATDES86B12I854K', 2, 1, 'The Flash', 2014, 'TV', '2015-05-16');
INSERT INTO guarda VALUES ('MATDES86B12I854K', 3, 1, 'The Flash', 2014, 'TV', '2015-05-17');
INSERT INTO guarda VALUES ('MATDES86B12I854K', 1, 1, 'Stranger Things', 2016, 'Computer', '2016-10-17');
INSERT INTO guarda VALUES ('MATDES86B12I854K', 1, 1, 'Wayward Pines', 2015, 'Tablet', '2018-01-13');
INSERT INTO guarda VALUES ('MATDES86B12I854K', 2, 1, 'Wayward Pines', 2015, 'Tablet', '2018-01-13');
INSERT INTO guarda VALUES ('GIOGIB84H62C9930', 1, 1, 'Wayward Pines', 2015, 'Phone', '2017-09-30');
INSERT INTO guarda VALUES ('GIOGIB84H62C9930', 1, 1, 'Tredici', 2017, 'TV', '2018-03-05');
INSERT INTO guarda VALUES ('GIOGIB84H62C9930', 2, 1, 'Tredici', 2017, 'TV', '2018-03-06');
INSERT INTO guarda VALUES ('GIOGIB84H62C9930', 3, 1, 'Tredici', 2017, 'TV', '2018-03-07');
INSERT INTO guarda VALUES ('GIOGIB84H62C9930', 1, 1, 'The Flash', 2014, 'Computer', '2016-02-04');

-- Utente vota una puntata
INSERT INTO vota VALUES ('MATDES86B12I854K', 10, 1, 1, 'The Flash', 2014);
INSERT INTO vota VALUES ('MATDES86B12I854K', 8, 3, 1, 'The Flash', 2014);
INSERT INTO vota VALUES ('MATDES86B12I854K', 8, 1, 1, 'Stranger Things', 2016);
INSERT INTO vota VALUES ('MATDES86B12I854K', 7, 2, 1, 'Wayward Pines', 2015);
INSERT INTO vota VALUES ('GIOGIB84H62C9930', 6, 1, 1, 'Wayward Pines', 2015);
INSERT INTO vota VALUES ('GIOGIB84H62C9930', 8, 1, 1, 'Tredici', 2017);
INSERT INTO vota VALUES ('GIOGIB84H62C9930', 7, 3, 1, 'Tredici', 2017);
INSERT INTO vota VALUES ('GIOGIB84H62C9930', 9, 1, 1, 'The Flash', 2014);

-- Utente recensisce una stagione
INSERT INTO recensione VALUES ('MATDES86B12I854K', 'The Flash',
2014, 'Molto bella!', 'Questa serie TV mi sta piacendo molto...');

-- Inserimento attori
INSERT INTO attore VALUES ('GSTGNT90A14Z404L', 'Grant', 'Gustin', '1990-01-14', 'Stati Uniti', 'Norfolk', 'Grant è
nato il 14 gennaio 1990 a Norfolk in Virginia da Tina Haney, un''infermiera pediatrica, e Thomas (Tom) Gust
in, un insegnante. Ha un fratello maggiore di nome Tyler e una sorella minore di nome Gracie.');
```

INSERT INTO attore VALUES ('PTTCDC18H06Z404A', 'Candice', 'Patton', '1988-06-24', 'Stati Uniti', 'Jackson', 'Nasce
a Jackson, nel Mississippi, ma cresce a Plano, in Texas. Ha frequentato la Southern Methodist University a
Dallas.');

INSERT INTO attore VALUES ('BRWMLB04B50Z131R', 'Millie Bobby', 'Brown', '2004-02-19', 'Spagna', 'Malaga', 'Terzoge
nita di quattro figli, Millie è nata a Malaga (Spagna) da genitori britannici che all''epoca gestivano un ri
storante a Marbella.');

INSERT INTO attore VALUES ('LNGKHR96D69Z700Y', 'Katherine', 'Langford', '1996-04-29', 'Australia', 'Perth', 'Figli
a di Stephen e Elizabeth, Katherine ha iniziato a studiare recitazione dopo il diploma alla Perth Modern Sch
ool, in cui era una nuotatrice classificata a livello nazionale.');

INSERT INTO attore VALUES ('DLLMTT18H06Z404U', 'Matt', 'Dillon', '1964-02-18', 'Stati Uniti', 'New Rochelle', 'Dil
lon nasce a New Rochelle (New York) il 18 febbraio 1964 figlio di immigrati irlandesi.');

INSERT INTO attore VALUES ('JNSTBY66P06Z404P', 'Toby', 'Jones', '1966-09-06', 'Regno Unito', 'Londra', 'Figlio del
l''attore Freddie Jones e dell''attrice Jeanne Heslewood, ha studiato alla Abingdon School di Oxfordshire');

```

-- Inserimento casts
INSERT INTO casts VALUES ('The Flash', 2014, 'GSTGNT90A14Z404L', 'Barry Allen');
INSERT INTO casts VALUES ('The Flash', 2014, 'PTTCDC18H06Z404A', 'Iris West');
INSERT INTO casts VALUES ('Stranger Things', 2016, 'BRWMLB04B50Z131R', 'Undici');
INSERT INTO casts VALUES ('Stranger Things', 2016, 'BRWMLB04B50Z131R', 'Jane Ives');
INSERT INTO casts VALUES ('Stranger Things', 2016, 'BRWMLB04B50Z131R', 'Jane Hopper');
INSERT INTO casts VALUES ('Tredici', 2017, 'LNGKHR96D69Z700Y', 'Hannah Baker');
INSERT INTO casts VALUES ('Wayward Pines', 2015, 'DLLMTT18H06Z404U', 'Ethan Burke');
INSERT INTO casts VALUES ('Wayward Pines', 2015, 'JNSTBY66P06Z404P', 'David Pilcher');
```

QUERY DI INTERROGAZIONE

-- Vedere la durata totale in ore/minuti della seconda stagione della serie TV 'Tredici'

```
SELECT Sum(durata) / 60 AS Ore,  
       Sum(durata)%60   Minuti  
FROM   puntata  
WHERE  titolo_serie = 'Tredici'  
AND    anno_serie  = 2017  
AND    n_stagione  = 2
```

-- Vedere l'elenco delle prossime puntata che saranno trasmesse nei prossimi giorni

```
SELECT data,  
       canale,  
       titolo_serie,  
       n_stagione,  
       n_episodio  
FROM   emissione  
WHERE  data >= CURRENT_TIMESTAMP  
ORDER BY data
```

-- Vedere la durata totale complessiva in giorni/ore/minuti per ciascuna serie tv.

```
SELECT titolo_serie,  
       Sum(durata) / ( 60 * 24 ) AS giorni,  
       ( Sum(durata) / 60 )%24   AS ore,  
       Sum(durata)%60           minuti,  
       Sum(durata)              AS totale_min  
FROM   puntata  
GROUP BY titolo_serie  
ORDER BY totale_min DESC
```

-- Visualizza i generi della serie TV "The Flash"

```
SELECT genere  
FROM   serie_genere  
WHERE  titolo = 'The Flash';
```

-- Coppie di utenti nate nella stessa città

```
SELECT x.nickname,  
       y.nickname  
FROM   utente x,  
       utente y  
WHERE  x.luogo_n = y.luogo_n  
AND    x.nickname < y.nickname
```

-- Vedere le puntate che sono senza trama

```
SELECT *  
FROM   puntata  
WHERE  descrizione IS NULL
```

-- Ottenere il cast della serie tv 'Wayward Pines'.

```
SELECT DISTINCT attore.nome,  
               attore.cognome  
FROM   attore,  
       casts  
WHERE  casts.cf_attore = attore.cf  
AND    titolo_serie = 'Wayward Pines'
```

-- Vedere tutte le serie tv in cui c'è 'Millie Bobby Brown'

```
SELECT DISTINCT titolo_serie  
FROM   casts,  
       attore  
WHERE  casts.cf_attore = attore.cf  
AND    nome = 'Millie Bobby'  
AND    cognome = 'Brown'
```

-- Vedere le casi produttrici della serie TV 'Tredici'

```
SELECT nome  
FROM   casa_produttrice  
WHERE  titolo_serie = 'Tredici';
```

```
-- Calcolare la media dei voti per la serie TV 'Wayward Pines'
SELECT Avg(voto)
FROM   vota
WHERE  titolo_serie = 'Wayward Pines'
      AND anno_serie = '2015';
```

```
-- Ottenere una classifica di gradimento delle serie tv in base ai voti degli utenti
SELECT titolo_serie,
       Avg(voto)
FROM   vota
GROUP BY titolo_serie
ORDER BY 2 DESC
```

```
-- Calcolare il numero totale di puntate viste per ciascun utente in ordine decrescente
SELECT nome,
       cognome,
       Count(*) AS EpisodiVisti
FROM   guarda,
       utente
WHERE  guarda.cf = utente.cf
GROUP BY nome,
       cognome
ORDER BY episodivisti DESC
```

```
-- Vedere la lista di tutte le serie TV con genere thriller
SELECT titolo
FROM   serie_genere
WHERE  genere = 'Thriller'
```

```
-- Informazioni sui giorni/ore/minuti TOTALI di tutte le puntate viste da 'Mattia Desiderio'
SELECT Sum(durata) / ( 60 * 24 ) AS giorni,
       ( Sum(durata) / 60 )%24   AS ore,
       Sum(durata)%60           minuti
FROM   puntata p,
       guarda g,
       utente u
WHERE  g.cf = u.cf
      AND g.n_episodio = p.n_episodio
      AND g.n_stagione = p.n_stagione
      AND g.titolo_serie = p.titolo_serie
      AND g.anno_serie = p.anno_serie
      AND nome = 'Mattia'
      AND cognome = 'Desiderio'
```

```
Selezionare le serie TV che hanno più di 2 stagioni, ma che non hanno più di 10 episodi per stagione
SELECT titolo_serie,
       anno_serie
FROM   serietv
WHERE  n_stagioni > 2
EXCEPT
SELECT serietv.titolo_serie,
       serietv.anno_serie
FROM   serietv,
       stagione
WHERE  serietv.titolo_serie = stagione.titolo_serie
      AND serietv.anno_serie = stagione.anno_serie
      AND n_episodi > 20
```

QUERY DI MODIFICA

Modificare la password dell'utente con codice fiscale 'MATDES86B12I854K'

```
UPDATE utente
SET password = Crypt('nuovapassword', Gen_salt('md5'))
WHERE cf = 'MATDES86B12I854K';
```

Aggiornare la biografia dell'attore Matt Dillon

```
UPDATE attore
SET biografia = 'Nuova biografia'
WHERE nome = 'Matt'
AND cognome = 'Dillon';
```

Aggiornare a 8 il voto dell'utente con codice fiscale 'GIOGIB84H62C9930' per la serie TV Tredici (3x1)

```
UPDATE votoa
SET voto = 8
WHERE cf = 'GIOGIB84H62C9930'
AND n_episodio = 3
AND n_stagione = 1
AND titolo_serie = 'Tredici';
```

L'utente con CF 'MATDES86B12I854K' ha rivisto l'episodio 2x1 di 'The Flash' nella data odierna. Aggiorna la data di visione alla data corrente.

```
UPDATE guarda
SET data = CURRENT_DATE
WHERE cf = 'MATDES86B12I854K'
AND n_episodio = 2
AND n_stagione = 1
AND titolo_serie = 'The Flash';
```


QUERY DI ELIMINAZIONE

-- Eliminazione di tutte le recensioni che contengono un insulto (es: "schifo")

```
DELETE FROM recensione
WHERE testo LIKE '%schifo%';
```

-- Eliminazione degli utenti che non hanno visto nessuna puntata

```
DELETE FROM utente
WHERE cf NOT IN (SELECT cf
                 FROM guarda);
```

Eliminazione di tutti i voti effettuati dell'utente con codice fiscale 'MATDES86B12I854K'

```
DELETE FROM vota
WHERE cf = 'MATDES86B12I854K';
```

Eliminare il genere 'Drammatico' da quelli preferiti dell'utente con CF 'GIOGIB84H62C9930'

```
DELETE FROM utente_genere
WHERE cf = 'GIOGIB84H62C9930'
      AND genere = 'Drammatico';
```

Annulla la visione dell'episodio 3x1 della serie tv Tredici per l'utente con CF 'GIOGIB84H62C9930'

```
DELETE FROM guarda
WHERE cf = 'GIOGIB84H62C9930'
      AND n_episodio = 3
      AND n_stagione = 1
      AND titolo_serie = 'Tredici';
```

TRIGGERS

Trigger auto incremento numero di stagioni. Quando una stagione viene inserita, viene incrementato il numero di stagioni di quella serie.

```
CREATE FUNCTION new_stagione() returns TRIGGER AS $$
BEGIN
    UPDATE serietv
    SET     n_stagioni=n_stagioni+1
           /* Alternativa: n_stagioni=new.n_stagioni */
    WHERE  titolo_serie=new.titolo_serie
    AND    anno_serie=new.anno_serie;

    return new;
END;
$$ language 'plpgsql';

CREATE TRIGGER nuova_stagione
AFTER INSERT ON stagione
FOR each row EXECUTE PROCEDURE new_stagione();
```

Trigger auto incremento numero di episodi. Quando un episodio viene inserito, viene incrementato il numero di episodi di quella stagione.

```
CREATE FUNCTION new_puntata() returns TRIGGER AS $$
BEGIN
    UPDATE stagione
    SET     n_episodi=n_episodi+1
    WHERE  titolo_serie=new.titolo_serie
    AND    anno_serie=new.anno_serie
    AND    n_stagione=new.n_stagione;

    return new;
end;
$$ language 'plpgsql';

CREATE TRIGGER nuova_puntata
AFTER INSERT ON puntata
FOR each row EXECUTE PROCEDURE new_puntata();
```

Trigger: un utente può votare una puntata solo se l'ha vista. Il trigger controlla se nella tabella guarda è presente almeno una tupla relativa all'utente e serie in questione.

```
CREATE FUNCTION check_voto() returns TRIGGER AS $$
DECLARE
    contatore int;
BEGIN
    SELECT count(*)
    INTO     contatore
    FROM     guarda
    WHERE    cf=new.cf
    AND      n_episodio=new.n_episodio
    AND      n_stagione=new.n_stagione
    AND      titolo_serie=new.titolo_serie
    AND      anno_serie=new.anno_serie;

    if (contatore = 0) THEN
        raise exception 'Non puoi votare una puntata che non hai visto!';
    END IF;
    return new;
end;
$$ language 'plpgsql';

CREATE TRIGGER controllo_voto
BEFORE INSERT ON vota
FOR each row EXECUTE PROCEDURE check_voto();
```


-- Trigger: per poter recensire una serie tv, l'utente deve aver visto più di un episodio di quella serie.

```
CREATE FUNCTION check_recensione() returns TRIGGER AS $$
DECLARE
    contatore int;
begin
    SELECT count(*)
    INTO    contatore
    FROM    guarda
    WHERE   cf=new.cf
    AND     titolo_serie=new.titolo_serie
    AND     anno_serie=new.anno_serie;

    if (contatore <= 1) THEN
        raise exception 'Per poter recensire devi aver visto più di un episodio di quella serie tv.';
    end IF;
return new;
end;
$$ language 'plpgsql';

CREATE TRIGGER CONTROLLO_RECENSIONE
BEFORE INSERT ON recensione
FOR EACH ROW EXECUTE PROCEDURE check_recensione();
```

Trigger: controllo data di visione della puntata (deve essere maggiore dell'anno di uscita della stagione di quella puntata)

```
CREATE FUNCTION check_data_guarda() returns TRIGGER AS $$
BEGIN
    IF (date_part('year', new.data) <
        (
            SELECT anno_stagione
            FROM    stagione
            WHERE   n_stagione=new.n_stagione
            AND     titolo_serie=new.titolo_serie
            AND     anno_serie=new.anno_serie))
    THEN
        raise exception
        'L'anno della data di visione della puntata è inferiore all'anno di uscita della stagione di quella puntata';
    end IF;
RETURN new;
END;
$$ language 'plpgsql';

CREATE TRIGGER controllo_dataguarda
BEFORE INSERT ON guarda
FOR each row EXECUTE PROCEDURE check_data_guarda();
```

Trigger: controllo data dell'emissione (deve essere maggiore dell'anno di uscita della stagione di quella puntata)

```
CREATE FUNCTION check_data_emissione() returns TRIGGER AS $$
BEGIN
    IF (date_part('year', new.data) <
        (
            SELECT anno_stagione
            FROM    stagione
            WHERE   n_stagione=new.n_stagione
            AND     titolo_serie=new.titolo_serie
            AND     anno_serie=new.anno_serie))
    THEN
        raise exception
        'L'anno della data dell'emissione è inferiore all'anno di uscita della stagione di quella puntata';
    end IF;
RETURN new;
END;
$$ language 'plpgsql';

CREATE TRIGGER controllo_dataguarda
BEFORE INSERT ON emissione
FOR each row EXECUTE PROCEDURE check_data_emissione();
```

.....

```
localhost - serietv v
1 -- Vedere la durata totale complessiva in giorni/ore/minuti per ciascuna serie tv.
2 SELECT titolo_serie, sum(durata)/(60*24) AS giorni,
3                (sum(durata)/60)%24 AS ore,
4                sum(durata)%60 minuti,
5                sum(durata) AS totale_min
6 FROM puntata
7 GROUP BY titolo_serie
8 ORDER BY totale_min DESC
```

	titolo_serie	giorni	ore	minuti	totale_min
1	The Flash	2	17	56	3956
2	Tredici	1	0	42	1482
3	Wayward Pines	0	15	0	900
4	Stranger Things	0	14	27	867

localhost - serietv ▾

```
1 -- Visualizza i generi della serie TV "The Flash"
2 SELECT genere
3 FROM serie_genere
4 WHERE titolo='The Flash';|
```

▶

	genere
1	Supereroi
2	Azione
3	Drammatico
4	Avventura

localhost - serietv ▾

```
1 -- Ottenere il cast della serie tv 'Wayward Pines'.
2 SELECT DISTINCT attore.nome, attore.cognome
3 FROM attore, casts
4 WHERE casts.cf_attore=attore.cf
5 AND titolo_serie='Wayward Pines'
```

▶

	nome	cognome
1	Matt	Dillon
2	Toby	Jones

localhost - serietv ▾

```
1 -- Calcolare la media dei voti per la serie TV 'Wayward Pines'
2 SELECT AVG(voto) FROM vota
3 WHERE titolo_serie='Wayward Pines' AND anno_serie='2015';|
```

▶

	avg
1	6.5

localhost - serietv ▾			
<pre> 1 -- Calcolare il numero totale di puntate viste per ciascun utente in ordine decrescente 2 SELECT nome,cognome, COUNT(*) AS EpisodiVisti 3 FROM guarda,utente 4 WHERE guarda.cf=utente.cf 5 GROUP BY nome,cognome 6 ORDER BY EpisodiVisti DESC </pre>			
▶			
	nome	cognome	episodivisti
1	Mattia	Desiderio	6
2	Giorgia	Gibellini	5

localhost - serietv Favorites

```
1 -- Classifica utenti visione serie TV
2 SELECT nome, cognome, sum(durata)/(60*24) AS giorni, (sum(durata)/60)%24 AS ore, sum(durata)%60 minuti
3 FROM puntata p,guarda g, utente u
4 WHERE g.cf = u.cf AND g.n_episodio=p.n_episodio
5 AND g.n_stagione=p.n_stagione AND g.titolo_serie=p.titolo_serie
6 AND g.anno_serie=p.anno_serie
7 GROUP BY nome, cognome
8 ORDER BY 3,4,5 DESC
```

	nome	cognome	giorni	ore	minuti
1	Mattia	Desiderio	0	4	28
2	Giorgia	Gibellini	0	4	11

```
localhost - serietv v Fa
1 -- Informazioni sui giorni/ore/minuti TOTALI di tutte le puntate viste da 'Mattia Desiderio'
2 SELECT sum(durata)/(60*24) AS giorni, (sum(durata)/60)%24 AS ore, sum(durata)%60 minuti
3 FROM puntata p,guarda g, utente u
4 WHERE g.cf = u.cf AND g.n_episodio=p.n_episodio
5 AND g.n_stagione=p.n_stagione AND g.titolo_serie=p.titolo_serie
6 AND g.anno_serie=p.anno_serie AND nome='Mattia' AND cognome='Desiderio'
```

▶

	giorni	ore	minuti
1	0	4	28

localhost - serietv ▾			
<pre> 1 -- Ottenere una classifica di gradimento delle serie tv in base ai voti degli utenti 2 SELECT titolo_serie, AVG(voto) 3 FROM vota 4 GROUP BY titolo_serie 5 ORDER BY 2 DESC </pre>			
▶			
	titolo_serie	avg	
1	The Flash	9	
2	Stranger Things	8	
3	Tredici	7.5	
4	Wayward Pines	6.5	

STUDIO DEI DATI DERIVATI

Nel nostro progetto ci sono due dati derivati, ovvero il numero di stagioni per ciascuna serie TV e il numero di episodi di ciascuna stagione.

1) Come prima cosa andiamo a studiare il dato derivato "**NUM_STAGIONI**". Si suppone che vi siano 300 serie tv e 900 stagioni.

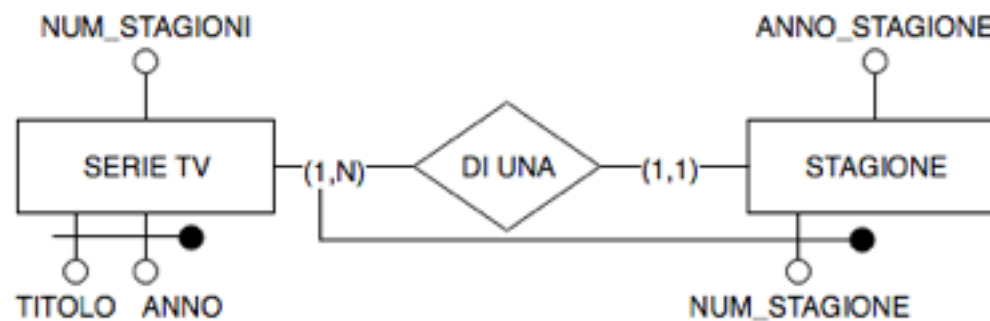


Tabella dei volumi

CONCETTO	TIPO	VOLUME
SERIE TV	E	300
STAGIONE	E	900
DI UNA	R	900

Tabella delle operazioni

OPERAZIONE	TIPO	FREQUENZA
OP.1 Inserimento Stagione	I	15/G
OP.2 Visione Serie TV	I	200/G

Scenario con dato derivato:

OP.1: Dobbiamo inserire una stagione. Successivamente dovremo "muoverci" verso il dato derivato (num_stagioni) per andarlo ad aggiornare. Ad ogni stagione è associata un'unica serie tv.

OP.2: Vogliamo vedere le informazioni relative alla serie TV (siccome abbiamo il dato derivato, ci basterà effettuare un'unica lettura sulla tabella delle serie TV).

Operazione 1

CONCETTO	ACCESSO	TIPO
STAGIONE	1	S
DI UNA	1	S
SERIE TV	1	L
SERIE TV	1	S

Operazione 2

CONCETTO	ACCESSO	TIPO
SERIE TV	1	L

OP.1: $7 \times 15 = 105$ accessi al giorno

OP.2: $1 \times 200 = 200$ accessi al giorno

TOTALE: $105 + 200 = 305/G$

Scenario senza dato derivato: (non abbiamo più il numero di stagioni in serie tv)

OP.1: Dobbiamo inserire una stagione. Non abbiamo più il dato derivato da aggiornare.

OP.2: Queste volta oltre a fare una lettura su serie tv, dovremo andarci a calcolare il dato derivato, spostandoci verso stagione.

Operazione 1 (no dato derivato)

CONCETTO	ACCESSO	TIPO
STAGIONE	1	S
DI UNA	1	S

Operazione 2 (no dato derivato)

CONCETTO	ACCESSO	TIPO
SERIE TV	1	L
DI UNA	$900/300=3$	L
STAGIONE	3	L

OP.1: $4 \times 15 = 60$ accessi al giorno

OP.2: $7 \times 200 = 1400$ accessi al giorno

TOTALE: $60 + 1400 = \underline{1460/G}$

Ci conviene avere ed usare il dato derivato "num_stagioni".

2) Ora studiamo il dato derivato "**NUM_EPISODI**". Si suppone che vi siano 900 stagioni e 10.000 episodi.

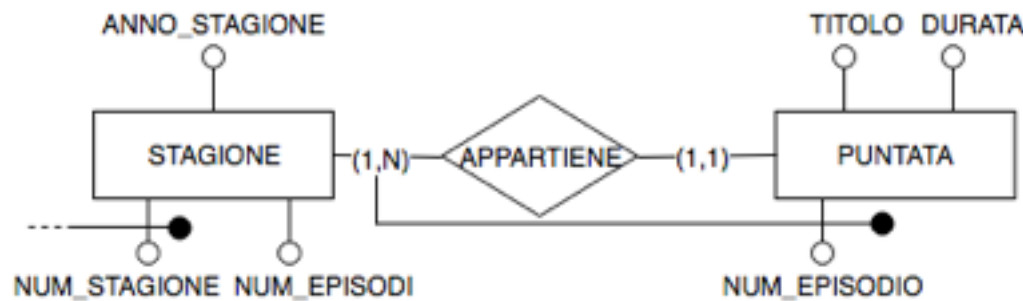


Tabella dei volumi

CONCETTO	TIPO	VOLUME
STAGIONE	E	900
PUNTATA	E	10.000
APPARTIENE	R	10.000

Tabella delle operazioni

OPERAZIONE	TIPO	FREQUENZA
OP.1 Inserimento Puntata	I	250/G
OP.2 Visione Stagione	I	600/G

Scenario con dato derivato:

OP.1: Dobbiamo inserire una puntata. Successivamente dovremo "muoverci" verso il dato derivato (num_episodi) per andarlo ad aggiornare. Ad ogni puntata è associata un'unica stagione.

OP.2: Vogliamo vedere le informazioni relative alla stagione (siccome abbiamo il dato derivato, ci basterà effettuare un'unica lettura sulla tabella delle stagioni).

Operazione 1

CONCETTO	ACCESSO	TIPO
PUNTATA	1	S
APPARTIENE	1	S
STAGIONE	1	L
STAGIONE	1	S

Operazione 2

CONCETTO	ACCESSO	TIPO
STAGIONE	1	L

OP.1: $7 \times 250 = 1750$ accessi al giorno

OP.2: $1 \times 600 = 600$ accessi al giorno

TOTALE: $1750 + 600 = \underline{2350/G}$

Scenario senza dato derivato: (non abbiamo più il numero di episodi in stagione)

OP.1: Dobbiamo inserire una puntata. Non abbiamo più il dato derivato da aggiornare.

OP.2: Queste volta oltre a fare una lettura su stagione, dovremo andarci a calcolare il dato derivato spostandoci verso puntata.

Operazione 1 (senza dato derivato)

CONCETTO	ACCESSO	TIPO
PUNTATA	1	S
APPARTIENE	1	S

Operazione 2 (senza dato derivato)

CONCETTO	ACCESSO	TIPO
STAGIONE	1	L
APPARTIENE	$1000/900=11$	L
PUNTATA	11	L

OP.1: $4 \times 250 = 1000$ accessi al giorno

OP.2: $23 \times 600 = 13800$ accessi al giorno

TOTALE: $1000 + 13800 = \underline{14800/G}$

Ci conviene avere ed usare il dato derivato "num_episodi".