



TV SHOW TIME

GRUPPO 3

MATTIA DESIDERIO, GIORGIA GIBELLINI

COS'E' TV SHOW TIME

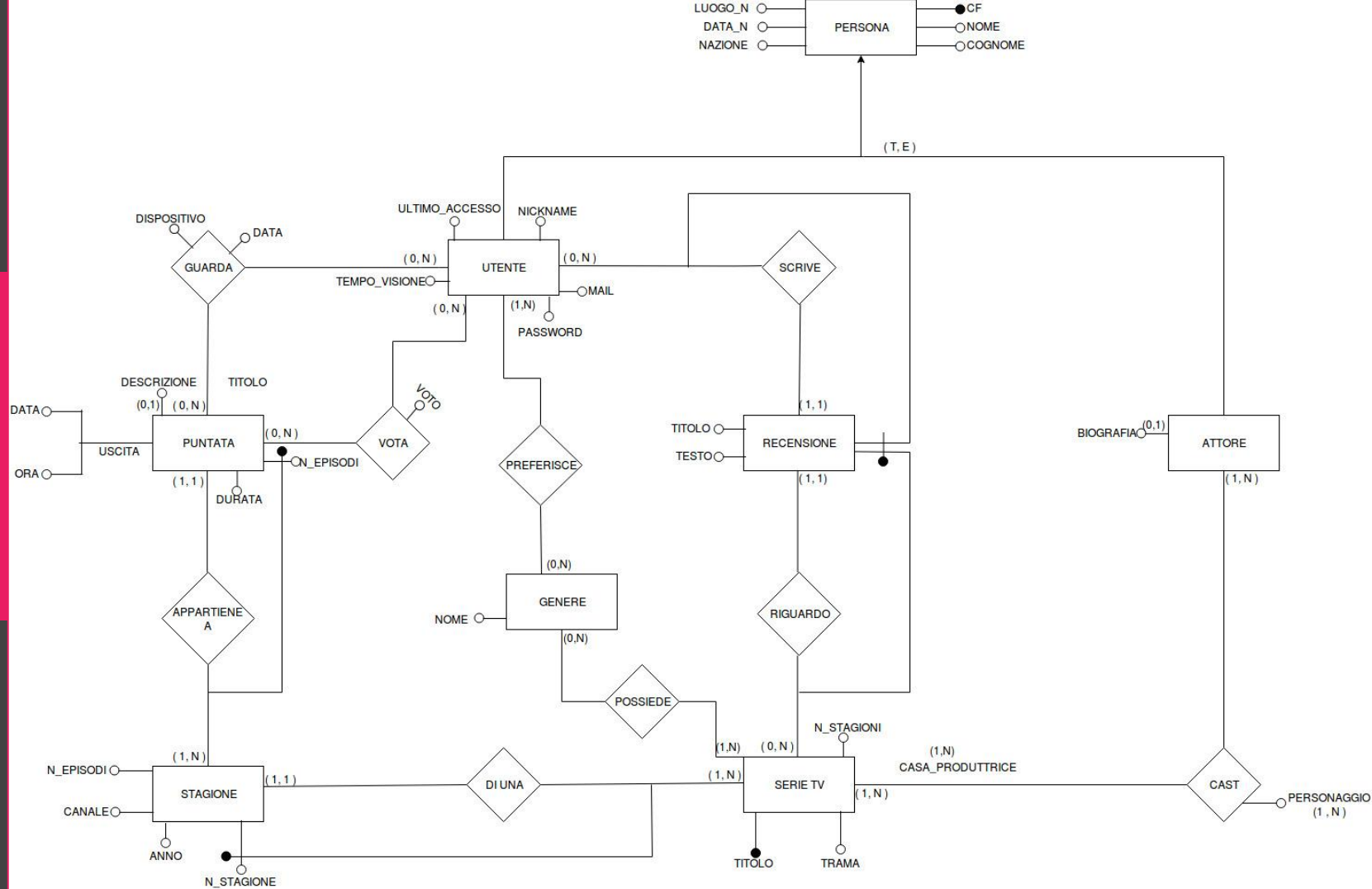
VOTA
EPISODI

RECENSISCI
SERIE TV

VISUALIZZA
CAST

PIATTAFORMA DI
GESTIONE DELLA
VISIONE DI SERIE
TV





PROSSIMI EPISODI IN ONDA

localhost - TV Time ▾

```
1 SELECT data, ora, titolo_serietv, num_stagione, num_episodio, titolo_puntata
2 FROM puntata
3 WHERE data >=current_date
4 ORDER BY data
```

	data	ora	titolo_serietv	num_stagione	num_episodio	titolo_puntata
1	2018-05-23	20:00:00	Legion	1	8	La caverna delle ombre
2	2018-05-30	20:00:00	Legion	1	9	Chapter 17
3	2018-06-05	21:15:00	The Flash	4	13	Colori veri
4	2018-06-06	20:00:00	Legion	1	10	Chapter 18
5	2018-06-12	21:15:00	The Flash	4	14	Soggetto 9
6	2018-06-19	21:15:00	The Flash	4	15	Enter Flashtime
7	2018-06-26	21:15:00	The Flash	4	16	Corri, Iris, Corri
8	2018-07-03	21:15:00	The Flash	4	17	Null and Annoyed
9	2018-07-10	21:15:00	The Flash	4	18	Lose Yourself
10	2018-07-17	21:15:00	The Flash	4	19	Fury Rogue
11	2018-07-24	21:15:00	The Flash	4	20	Therefore She is
12	2018-07-31	21:15:00	The Flash	4	21	Harry e gli Harrison
13	2018-08-07	21:15:00	The Flash	4	22	Pensa velocemente
14	2018-08-14	21:15:00	The Flash	4	23	Noi siamo Flash

DURATA TOTALE DI UNA SERIE TV

```
1 SELECT titolo_serietv, sum(durata)/(60*24) AS giorni,  
2                      (sum(durata)/60)%24 AS ore,  
3                      sum(durata)%60 minuti,  
4                      sum(durata) AS totale_min  
5 FROM puntata  
6 GROUP BY titolo_serietv  
7 ORDER BY totale_min DESC|
```

	titolo_serietv	giorni	ore	minuti	totale_min
1	The Flash	2	17	56	3956
2	Tredici	1	0	42	1482
3	Stranger Things	0	14	27	867

PASSWORD UTENTE CRIPTATA

```
CREATE TABLE utente
(
  cf char(16) PRIMARY KEY,
  nickname varchar(30) NOT NULL UNIQUE,
  nome varchar(30) NOT NULL,
  cognome varchar(30) NOT NULL,
  datan date NOT NULL CHECK (datan BETWEEN date '1930-01-01' AND current_date),
  nazione varchar(60) NOT NULL,
  luogonascita varchar(60) NOT NULL,
  email varchar(50) NOT NULL UNIQUE,
  ultimoacc date DEFAULT NULL,
  password varchar(60) NOT NULL,
  oretotvisione integer DEFAULT 0,
  CHECK (LENGTH(nickname) >= 4),
  CHECK (LENGTH(password) >= 6),
  CHECK(email ~* '^[A-Za-z0-9._%-]+@[A-Za-z0-9.-]+[.][A-Za-z]+$')
);
```

```
-- Gestione password
CREATE EXTENSION pgcrypto;
INSERT INTO utente VALUES (
  'GBRCVG86B12I854K',
  'mattia.desiderio',
  'Mattia',
  'Desiderio',
  '1996-04-05',
  'Italia',
  'Modena',
  '230925@studenti.unimore.it',
  NULL,
  crypt('miapassword', gen_salt('md5')),
  0
);
```

TRIGGER DI AGGIORNAMENTO

```
-- Trigger +1 numstagioni
CREATE FUNCTION new_stagione() RETURNS trigger AS $$
BEGIN
    UPDATE serietv
    SET numstagioni=numstagioni+1 /* Alternativa: numstagioni=new.num_stagione */
    WHERE titolo_serietv=new.titolo_serietv AND anno_serietv=new.anno_serietv;

    RETURN NEW;
END;
$$ LANGUAGE 'plpgsql';

CREATE TRIGGER NUOVA_STAGIONE
AFTER INSERT ON stagione
FOR EACH ROW EXECUTE PROCEDURE new_stagione();

-- Trigger +1 numepisodi
CREATE FUNCTION new_puntata() RETURNS trigger AS $$
BEGIN
    UPDATE stagione
    SET numepisodi=numepisodi+1
    WHERE titolo_serietv=new.titolo_serietv AND anno_serietv=new.anno_serietv AND num_stagione=new.num_stagione;

    RETURN NEW;
END;
$$ LANGUAGE 'plpgsql';

CREATE TRIGGER NUOVA_PUNTATA
AFTER INSERT ON puntata
FOR EACH ROW EXECUTE PROCEDURE new_puntata();
```