

Progettazione e sviluppo di una Base di dati relazionale per la gestione di un cinema

Massimo Colucci N86001834

March 2022

1 Descrizione del progetto

1.1 Requisiti informali

Si sviluppi un sistema informativo, composto da una base di dati relazionale e un applicativo Java dotato di GUI (Swing o JavaFX), per la gestione di un **cinema** multisala. Il sistema deve tenere traccia dei **film proiettati** in ciascuna sala, della schedulazione di ogni **spettacolo** e delle caratteristiche di ogni sala (audio dolby, tecnologia Imax). Il sistema dovrà permettere di stimare quali sono le fasce orarie di maggior affluenza, gli spettacoli più **remunerativi** e le sale maggiormente occupate durante gli **orari** di maggior affluenza.

1.2 Analisi del problema

La base di dati che sarà progettata e implementata sarà utile per la gestione di un cinema. Esso sarà di appoggio per l'applicativo Java. Il sistema permetterà di tenere traccia delle proiezioni in ogni sala del cinema per ogni fascia oraria, con i relativi informazioni. Sarà possibile inserire una proiezione specificando una data o un intervallo di giorni in cui sarà proiettato il film selezionato. Il sistema permetterà di selezionare i posti per l'acquisto del biglietto del film desiderato. Inoltre sarà possibile visionare i dati statistiche sui film e sulle sale più remunerativi e affluenti.

1.3 Glossario dei termini

Termine	Descrizione	Sinonimi	Collegamenti
Film	Film presente nel database da proiettare	Film proiettato	Sala
Proiezione	Proiezione del film nella sala	Spettacolo	Film, Sala
Schedulazione	Sistema per la gestione ed esecuzione delle operazioni sulle proiezioni	-	Proiezione
Sala	Luogo dove viene proiettato il film	-	Film, Cinema, Posto
Caratteristiche	Indica la tecnologia che è dotata la sala	Tecnologia	Sala
Posto	Indica la poltrona della sala dove lo spettatore si gode la proiezione del film	Occupate	Sala
Orari	Gli orari di inizio e di fine proiezione di ogni film in programmazione	-	Proiezione, Sala, Film

2 Class Diagram

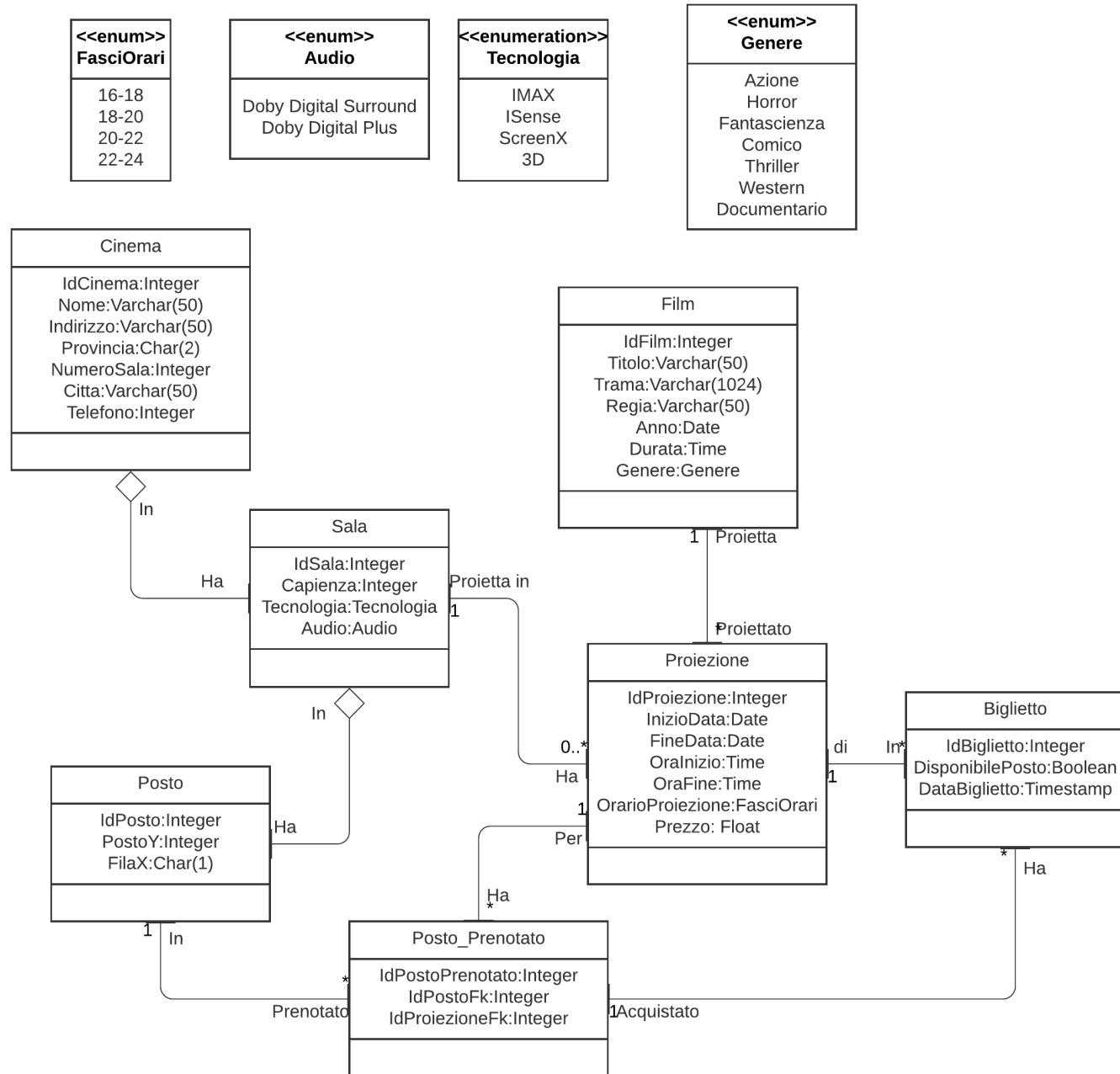


Figure 1: UML

3 Dizionario delle classi

Classe	Descrizione	Attributi
Cinema	Descrittore di un cinema	Nome (Varchar): nome del cinema Indirizzo (Varchar): indirizzo del cinema Provincia (Char(2)): provincia del cinema NumeroSala (Integer): il numero delle sale disponibili del cinema Città (Varchar): la città dove risiede il cinema Telefono (Integer): telefono fisso del cinema
Sala	Descrittore di ogni sala del cinema	Capienza (Integer): numero di posti della sala Tecnologia (Tecnologia): indica la tecnologia che la sala è dotata Audio (Audio): indica la tecnologia dell'audio che la sala è dotata
Posto	Descrittore dei posti appartenenti alla Sala	PostoY (Integer): l'identificativo della colonna nella sala FilaX (Integer): l'identificativo della fila nella sala
Film	Descrittore dei film proiettati	Titolo (Varchar): il titolo del film Trama (Varchar): la descrizione di dei fatti che si vedrà nel film Regia (Varchar): la direzione artistica di una realizzazione cinematografica Anno (Date): anno di uscita del film Durata (Time): la durata del film Genere (Genere): classificazione del film in base al tema che lo caratterizza
Proiezione	Descrittore delle proiezioni	InizioData (Date): prima data di proiezione del film FineData (Date): ultima data di proiezione del film OraInizio (Time): orario di inizio della proiezione del film OraFine (Time): orario di fine della proiezione del film OrarioProiezione (FasciOrari): intervallo di tempo di inizio e fine proiezione del film
Biglietto	Descrittore del biglietto	DisponibilePosto (Boolean): indicatore di disponibilità del posto DataBiglietto (Timestamp): data e ora di acquisto del biglietto
PostoPrenotato	Descrittore dei posti prenotati per le proiezioni selezionati	IdPostoFk (Integer): identifica univocamente il posto prenotato IdProiezioneFk (Integer): identifica univocamente la proiezione desiderata

Table 1: Dizionario delle classi

4 Dizionario delle Associazioni

Nome	Descrizione	Classi coinvolte
Cinema-Sala	Esprime l'appartenenza di sale ad un cinema	Cinema [1] ruolo In : indica il cinema che a cui appartiene una sala. Sala [0..*] ruolo ha : indica le sale che appartengono ad un cinema
Sala-Posto	Esprime l'appartenenza di posti ad una sala	Sala[1] ruolo In : indica la sala che contiene i posti. Posto[0..*] ruolo Ha : indica il posto che appartengono ad una sala.
Film-Proiezione	Esprime l'appartenenza di un film ad una proiezione	Film[1] ruolo proietta : indica il film proiettato. Proiezione[0..*] ruolo proiettato : indica la proiezione del film.
Sala-Proiezione	Esprime l'appartenenza di una sala ad una proiezione	Sala[1] ruolo proietta in : indica la sala dove c'è la proiezione. Proiezione[0..1] ruolo ha : indica le proiezioni per la sala.
Proiezione-Biglietto	Esprime l'appartenenza di una proiezione a dei biglietti	Proiezione[1] ruolo di : indica la proiezione a cui appartiene un biglietto. Biglietto[0..*] ruolo in : indica i biglietti di una proiezione.
PostoPrenotato Proiezione	Indica il posto prenotato per la proiezione	PostoPrenotato [0..*] ruolo Ha : indica il posto prenotato per la proiezione. Proiezione [1] ruolo per : indica la proiezione per il quale il posto è prenotato.
PostoPrenotato Posto	Indica quali posti sono prenotabili	PostoPrenotato[0..*] ruolo prenotato : indica i posti occupati. Posto[1] ruolo In : indica la posizione del posto prenotato.
PostoPrenotato Biglietto	Indica il biglietto a quale posto è stato prenotato	PostoPrenotato[1] ruolo Acquistato : indica il posto acquistato. Biglietto[0..*] ruolo Ha : indica il biglietto del posto prenotato.

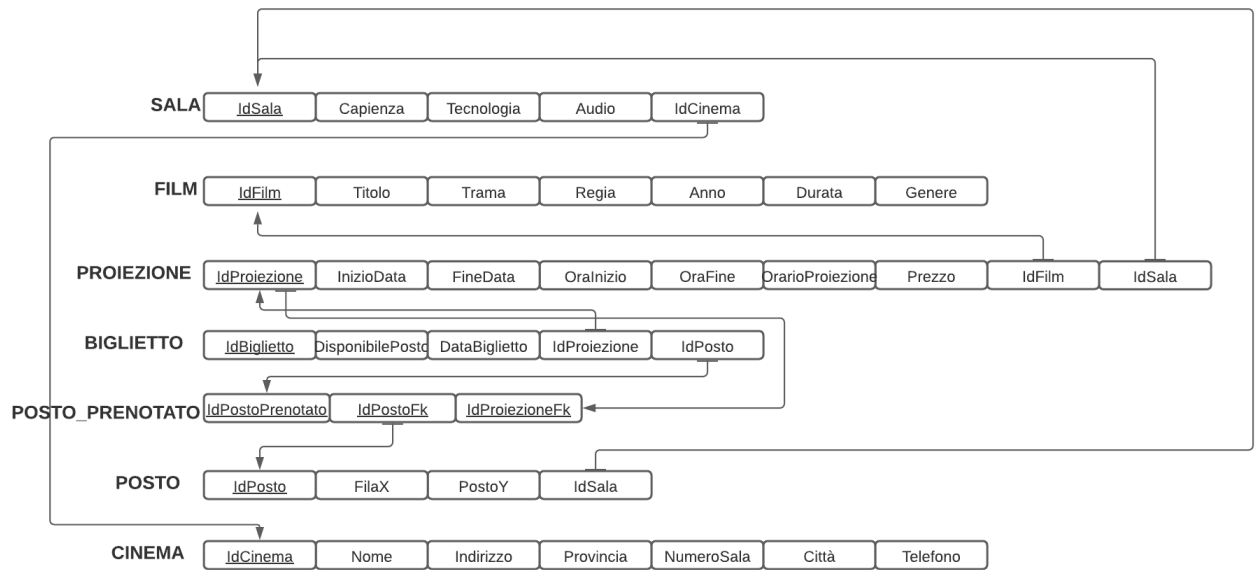
Table 2: Dizionario delle associazioni

5 Dizionario dei Vincoli

Nome Vincolo	Descrizione
chkTelefono	Assicura che un numero che inizia con 0 venga inserito nel Database
Range anno Film	Garantisce l'inserimento di Anno in formato YYYY e verifica che non venga inserito un anno superiore a quella corrente
chkDurata	Verifica che l'inserimento della durata sia compreso dall'intervallo indicato

Table 3: Dizionario dei Vincoli

6 Schema logico



7 Descrizione functions, procedures e triggers

```
1
2
3
4  --TRIGGER E FUNZIONI
5
6
7  --Funzione e Trigger che gestisce la creare di una sala rispettando il numero di
8  sala che il cinema pu contenere
9  DROP TRIGGER if exists controllo_numero_sala ON Sala;
10 DROP FUNCTION if exists limit_sala();
11
12 create function limit_sala() returns trigger as $BODY$
13 DECLARE
14     limite INTEGER;
15     capienza INTEGER;
16 BEGIN
17     select NumeroSala into limite from cinema;
18     select count(*) into capienza from sala;
19
20     if(capienza > limite - 1) then RAISE EXCEPTION 'Non ci sono pi sale disponibili
21     ';
22     ELSE RETURN NEW;
23     END IF;
24 END;
25 $BODY$
26 LANGUAGE PLPGSQL;
27
28 create trigger controllo_numero_sala
29 before insert on SALA
30 EXECUTE PROCEDURE limit_sala();
31
32 --Funzione e Trigger che genera automaticamente le entit POSTO in base alla
33 capienza data dall'entit SALA
34 DROP TRIGGER if exists generate_posto ON Sala;
35 DROP FUNCTION if exists generate_Posto();
36
37 CREATE OR REPLACE FUNCTION generate_Posto()
38 RETURNS TRIGGER AS
39 $BODY$
40 DECLARE
41     capienza_sala INTEGER;
42     filx char[];
43     IdSalafk INTEGER;
44     counX INTEGER := 1;
45     counY INTEGER := 1;
46     curr varchar;
47 BEGIN
48     filx :=ARRAY['A','B','C','D','E','F','G','H','I','J','K','L','M','N','O','P','Q',
49     'R','S','T','U','V','W','X','Y','Z'];
50     IdSalafk := NEW.IdSala;
51     select capienza into capienza_sala from sala where IdSala = (select MAX(idsala)
52     from sala);
53     FOR itemX in 1..capienza_sala
54     LOOP
55         curr := filx[counX];
56         INSERT INTO POSTO(FilaX,PostoY,IdSalaFk)
57         VALUES
```



```

54         (curr,counY,IdSalafk);
55         counY:=counY+1;
56         IF counY = 15 then counY:=1; counX:= counX+1;
57         END IF;
58     END LOOP;
59     RETURN NEW;
60 END;
61 $BODY$
62 LANGUAGE PLPGSQL;
63
64 create trigger generate_posto
65 AFTER insert on SALA
66 FOR EACH ROW
67 EXECUTE PROCEDURE generate_Posto();
68
69 ---Funzione e Trigger che assegna la fascia oraria appropriata per la proiezione.
70 DROP TRIGGER IF EXISTS assignsFasciaOraria ON PROIEZIONE;
71 DROP FUNCTION IF EXISTS assignsFasciaOraria();
72
73 CREATE OR REPLACE FUNCTION assignsFasciaOraria()
74 RETURNS TRIGGER AS
75 $$
76 DECLARE
77 BEGIN
78     IF(NEW.OraInizio >= TIME '16:00:00' AND NEW.OraInizio < TIME '18:00:00') then
79         UPDATE PROIEZIONE
80         SET OrarioProiezione = '16-18':FASCIORARI
81         WHERE IdProiezione = NEW.IdProiezione;
82     ELSIF(NEW.OraInizio >= TIME '18:00:00' AND NEW.OraInizio < TIME '20:00:00') THEN
83         UPDATE PROIEZIONE
84         SET OrarioProiezione = '18-20':FASCIORARI
85         WHERE IdProiezione = NEW.IdProiezione;
86     ELSIF(NEW.OraInizio >= TIME '20:00:00' AND NEW.OraInizio < TIME '22:00:00') THEN
87         UPDATE PROIEZIONE
88         SET OrarioProiezione = '20-22':FASCIORARI
89         WHERE IdProiezione = NEW.IdProiezione;
90     ELSIF(NEW.OraInizio >= TIME '22:00:00' AND NEW.OraInizio < TIME '23:59:00') THEN
91         UPDATE PROIEZIONE
92         SET OrarioProiezione = '22-24':FASCIORARI
93         WHERE IdProiezione = NEW.IdProiezione;
94     END IF;
95     RETURN NEW;
96 END;
97 $$
98 LANGUAGE PLPGSQL;
99
100 CREATE TRIGGER assignsFasciaOraria
101 AFTER INSERT on PROIEZIONE
102 FOR EACH ROW
103 EXECUTE PROCEDURE assignsFasciaOraria();
104
105
106 ---Funzione e Trigger che inserisce automaticamente records ogni nuovo biglietto
107 DROP TRIGGER IF EXISTS insertPostPrenotato ON BIGLIETTO;
108 DROP FUNCTION IF EXISTS insertPostPrenotato();
109
110 CREATE OR REPLACE FUNCTION insertPostPrenotato()
111 RETURNS TRIGGER AS
112 $$
113 DECLARE

```

```

114 BEGIN
115     INSERT INTO POSTO_PRENOTATO(IdPostoFk,IdProiezioneFk,IdBigliettoFk)
116     VALUES
117     (NEW.IdPostoFk,NEW.IdProiezioneFk,NEW.IdBiglietto);
118     RETURN NEW;
119 END;
120 $$
121 LANGUAGE PLPGSQL;
122
123 CREATE TRIGGER insertPostPrenotato
124 AFTER INSERT on BIGLIETTO
125 FOR EACH ROW
126 EXECUTE PROCEDURE insertPostPrenotato();
127
128
129
130
131
132
133
134
135 --Funzione e Trigger che solleva errore se esiste una proiezione nello stesso
    orario e nello stesso giorno
136 DROP TRIGGER if exists checkProiezione ON proiezione;
137 drop function if exists checkProiezione();
138
139 create or replace function checkProiezione()
140 returns trigger as
141 $$
142 DECLARE
143     proiezioni integer;
144 BEGIN
145     select count(*) into proiezioni
146     FROM proiezione
147     WHERE iniziodata<=NEW.iniziodata and
148         finedata>=NEW.finedata and
149         idsalafk=NEW.idsalafk and
150         (orarioproiezione=NEW.orarioproiezione or
151         orainizio=NEW.orainizio and orafine=NEW.orafine);
152
153     IF proiezioni=0
154     THEN
155         RAISE info '%',proiezioni;
156         RAISE info 'Proiezione inserita';
157         return new;
158     ELSE
159         RAISE EXCEPTION 'Sala o proiezione gi esistente';
160     END IF;
161 END;
162 $$
163 LANGUAGE plpgsql;
164
165 CREATE TRIGGER checkProiezione
166 BEFORE INSERT on proiezione
167 FOR EACH ROW
168 EXECUTE PROCEDURE checkProiezione();

```

8 Definizione delle tabelle

```
1
2 CREATE TABLE CINEMA (
3     IdCinema SERIAL PRIMARY KEY,
4     NomeCinema VARCHAR(50) NOT NULL UNIQUE,
5     Indirizzo VARCHAR(250) NOT NULL UNIQUE,
6     Provincia CHAR(2) NOT NULL,
7     NumeroSala INTEGER NOT NULL,
8     Citta VARCHAR(50) NOT NULL,
9     Telefono CHAR(10) NOT NULL UNIQUE,
10    CONSTRAINT chk_telefono CHECK (Telefono not like '%[~0-9]%' ) --Assicura che un
    numero che inizia con 0 venga inserito nel DB
11 );
12
13 CREATE TYPE TECNOLOGIA AS ENUM ('null','IMAX','ISense','ScreenX','3D');
14 CREATE TYPE AUDIO AS ENUM ('null','Dolby Digital Surround','Dolby digital plus');
15
16 CREATE TABLE SALA (
17     IdSala INTEGER PRIMARY KEY,
18     Capienza INTEGER NOT NULL,
19     Tecnologia TECNOLOGIA DEFAULT NULL,
20     Audio AUDIO DEFAULT NULL,
21     IdCinemaFk INTEGER NOT NULL,
22     CONSTRAINT fk_Cinema FOREIGN KEY (IdCinemaFk) REFERENCES CINEMA(IdCinema)
23 );
24
25 CREATE TABLE POSTO(
26     IdPosto SERIAL PRIMARY KEY,
27     FilaX CHAR(1) NOT NULL,
28     PostoY SMALLINT CHECK(PostoY BETWEEN 1 AND 28) NOT NULL,
29     IdSalaFk INTEGER NOT NULL,
30     CONSTRAINT fk_Sala FOREIGN KEY (IdSalaFk) REFERENCES SALA(IdSala) ON DELETE
    CASCADE ON UPDATE CASCADE
31 );
32
33 CREATE TYPE GENERE AS ENUM ('Azione','Horror','Fantascienza','Comico','Thriller','
    Western','Documentario','Drammatico');
34
35 CREATE TABLE FILM (
36     IdFilm SERIAL PRIMARY KEY,
37     Titolo VARCHAR(50) NOT NULL UNIQUE,
38     Trama VARCHAR(100) DEFAULT 'Nessuna trama inserita',
39     Regia VARCHAR(50) DEFAULT 'Nessun regista inserito' NOT NULL,
40     Anno SMALLINT CHECK(Anno BETWEEN 0 AND EXTRACT(YEAR FROM CURRENT_DATE)) NOT NULL,
    --Garantisce l'inserimento di Anno in formato YYYY e verifica che non venga
    inserito un anno superio a quella corrente
41     Durata TIME NOT NULL,
42     Genere GENERE NOT NULL,
43     CONSTRAINT chk_durata CHECK(durata > TIME '01:19:00' AND durata < TIME '03:20:00'
    ) --Verifica che l'inserimento della durata sia compreso dall'intervallo indicato
44 );
45
46
47
48
49
50
51
52 CREATE TYPE FASCIORARI AS ENUM ('16-18','18-20','20-22','22-24');
```

```

53
54 CREATE TABLE PROIEZIONE(
55     IdProiezione SERIAL PRIMARY KEY,
56     InizioData DATE DEFAULT CURRENT_DATE, --Se il campo vuoto, imposta la data
        corrente
57     FineData DATE DEFAULT (CURRENT_DATE + interval '28' DAY), -- se il campo vuoto
        , imposta 28 giorni di disponibilit
58     OraInizio TIME NOT NULL,
59     OraFine TIME NOT NULL, --Trigger che imposta automaticamente l'orario di fine
        proiezione
60     OrarioProiezione FASCIORARI DEFAULT NULL, ---Trigger gestito
61     Prezzo FLOAT NOT NULL,
62     IdFilmFk INTEGER NOT NULL,
63     IdSalaFk INTEGER NOT NULL,
64     CONSTRAINT fk_Film FOREIGN KEY (IdFilmFk) REFERENCES FILM(IdFilm) ON DELETE
        CASCADE ON UPDATE CASCADE,
65     CONSTRAINT fk_SalaP FOREIGN KEY (IdSalaFk) REFERENCES SALA(IdSala) ON DELETE
        CASCADE ON UPDATE CASCADE
66 );
67
68 CREATE TABLE Biglietto(
69     IdBiglietto SERIAL PRIMARY KEY,
70     DisponibilePosto BOOLEAN DEFAULT 'False' NOT NULL,
71     DataBiglietto TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
72     IdProiezioneFk INTEGER NOT NULL,
73     IdPostoFk INTEGER NOT NULL,
74     CONSTRAINT fk_Proiezione FOREIGN KEY (IdProiezioneFk) REFERENCES PROIEZIONE(
        IdProiezione) ON DELETE CASCADE
75 );
76
77 CREATE TABLE POSTO_PRENOTATO(
78     IdPostoFk INTEGER NOT NULL,
79     IdProiezioneFk INTEGER NOT NULL,
80     IdBigliettoFk INTEGER NOT NULL,
81     CONSTRAINT fk_Posto FOREIGN KEY (IdPostoFk) REFERENCES POSTO(IdPosto) ON DELETE
        CASCADE,
82     CONSTRAINT fk_Proiezione FOREIGN KEY (IdProiezioneFk) REFERENCES PROIEZIONE(
        IdProiezione) ON DELETE CASCADE,
83     CONSTRAINT fk_Biglietto FOREIGN KEY (IdBigliettoFk) REFERENCES BIGLIETTO(
        IdBiglietto)
84 );

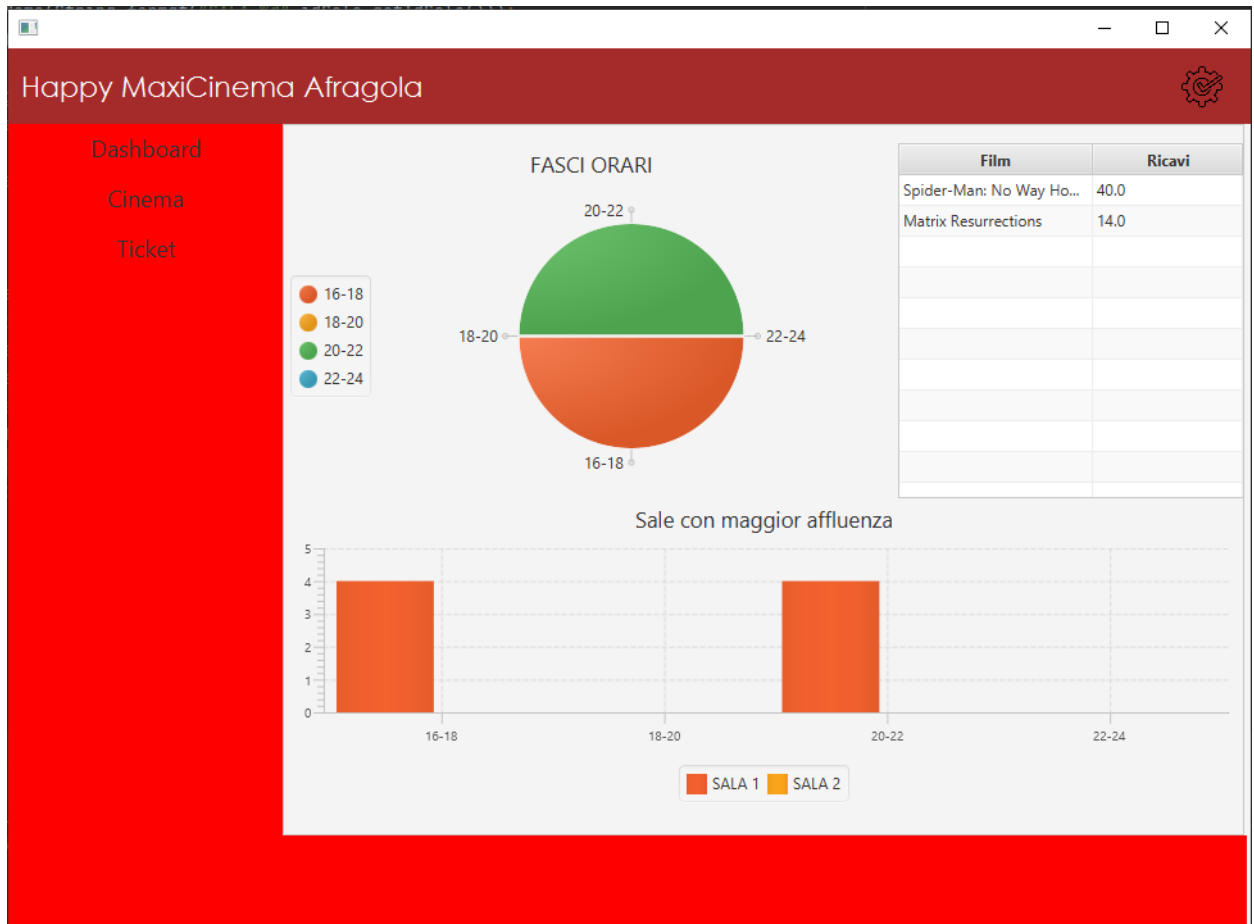
```

9 Manuale Utente

9.1 Pagina di benvenuto



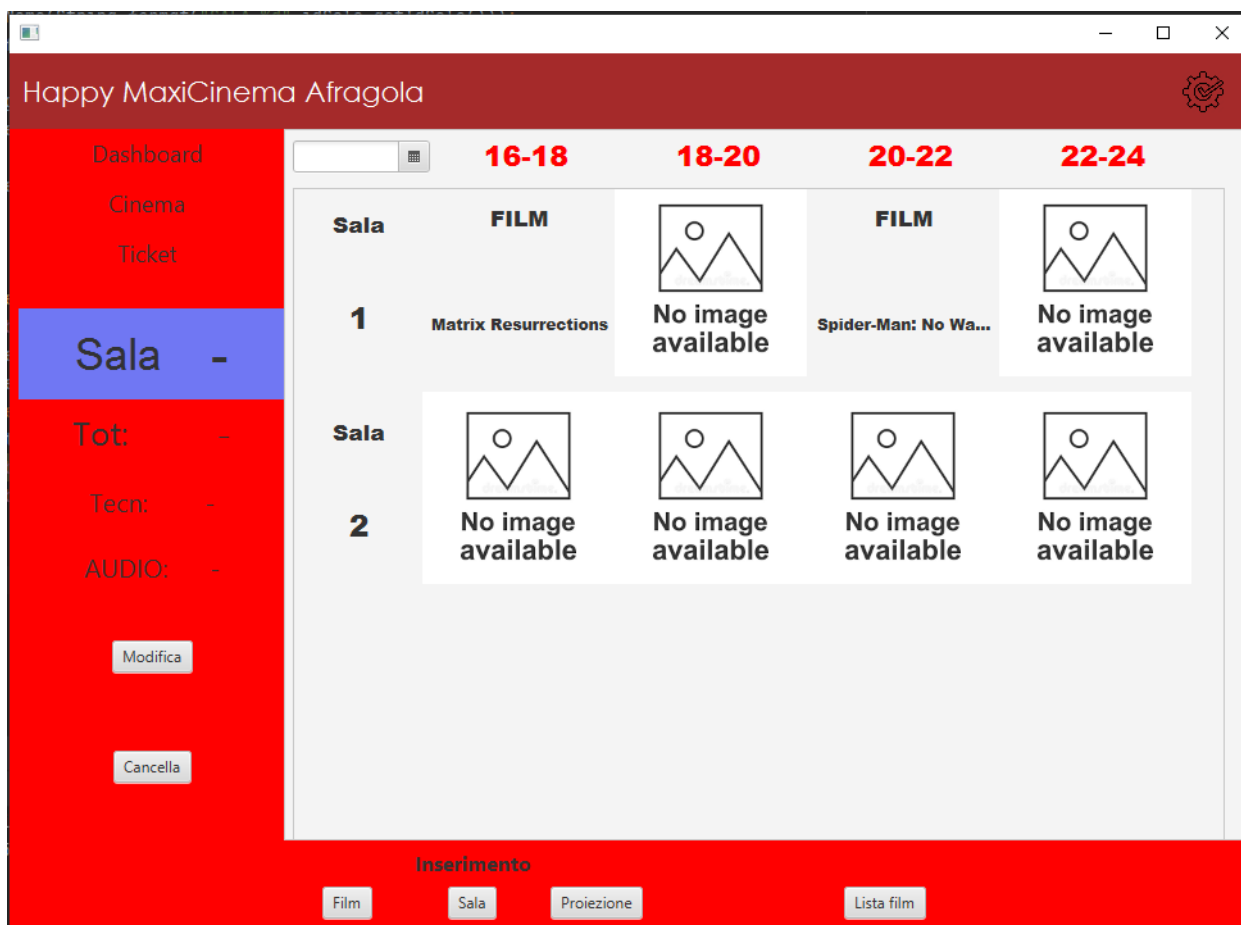
9.2 Dashboard



9.2.1 Descrizione

- **Diagramma a torta**: rappresenta le fasce orarie di maggior affluenza
- **Tabella dei film con ricavi**: elenca gli spettacoli più remunerativi
- **Grafico a barre**: le sale maggiormente occupate durante gli orari di maggior affluenz

9.3 Sezione Cinema



9.3.1 Descrizione

In questa sezione è possibile effettuare le operazioni più importanti sul cinema; in particolare sulle proiezioni per il giorno selezionato. Cliccando sulla sala o sul film, comparirà, sulla sinistra, i dettagli di essi con i pulsanti di modifica o cancellazione dell'elemento selezionato. E' possibile inoltre selezionare le sale con la fascia oraria senza proiezione, per inserire un film in proiezione, stabilendo il suo prezzo. In basso è possibile inserire un film, una sala o una proiezione; e visionare la lista di film che sono disponibili nel sistema.

9.4 Sezione Ticket

The screenshot shows a web application window titled "Happy MaxiCinema Afragola". On the left is a red sidebar with navigation links: "Dashboard", "Cinema", "Ticket", "Data", "Film", "Orario", "Cerca", and "Acquista". The "Ticket" link is highlighted. In the "Data" section, a date input shows "21/03/2022". In the "Film" section, a dropdown menu shows "Spider-Man: No Way H.". In the "Orario" section, a dropdown menu shows "20-22". The main area displays a cinema layout with rows A through H. Rows A through G have 14 seats each, numbered 1 to 14. Row H has only 2 seats, numbered 1 and 2. The seats are represented by numbered circles.

9.5 Descrizione

In questa sezione dell'applicazione è possibile selezionare la data, il film e la fascia oraria della proiezione che si desidera godere. Una volta cliccato su "Cerca", si mostrerà al centro la sala con i posti che è possibile selezionare per effettuare l'acquisto del biglietto, attraverso il pulsante di "Acquista".