

Sistema de Cinema

Felipe Luís Pinheiro - 18/0052667

João Pedro C.N. Mota - 17/0106144

Pedro Catelli - 17/0112624 Pedro Oliveira - 17/0163768

3 de julho de 2019

Resumo

Neste relatório desenvolvemos os requisitos básicos de um sistema de banco de dados para um modelo de vendas de ingresso de um cinema.

Link para o repositório: https://github.com/flpinheiro/banco_de_dados.

1 Introdução

Requisitos gerais:

- Um cinema pode ter muitas salas, sendo necessário, por tanto, registrar informações a respeito de cada uma, como sua capacidade, ou seja, o numero de assentos disponíveis.
- O cinema apresenta muitos filmes. Um filme tem informações, titulo e duração. Assim, sempre que um filme for ser apresentado, deve-se registrá-lo também.
- Um mesmo filme pode ser apresentado em diferentes salas e em horários diferentes. Cada apresentação em uma determinada sala e horário é chamada sessão. Um filme sendo apresentado em uma sessão tem um conjunto máximo de ingressos, determinado pela capacidade da sala.
- Os clientes do cinema podem comprar ou não ingressos para assistir a uma sessão. O funcionário deve intermediar a compra do ingresso. Um ingresso deve conter informação como o tipo de ingresso (Meio ingresso ou ingresso inteiro). Além disso, um cliente só pode comprar ingressos para sessões ainda não encerradas.

2 Diagrama de Entidade Relacionamento

Na figura 1 mostramos a primeira versão conceitual do sistema do

3 Modelo Relacional

Na figura 2 mostramos o modelo relacional utilizado para implementação do programa

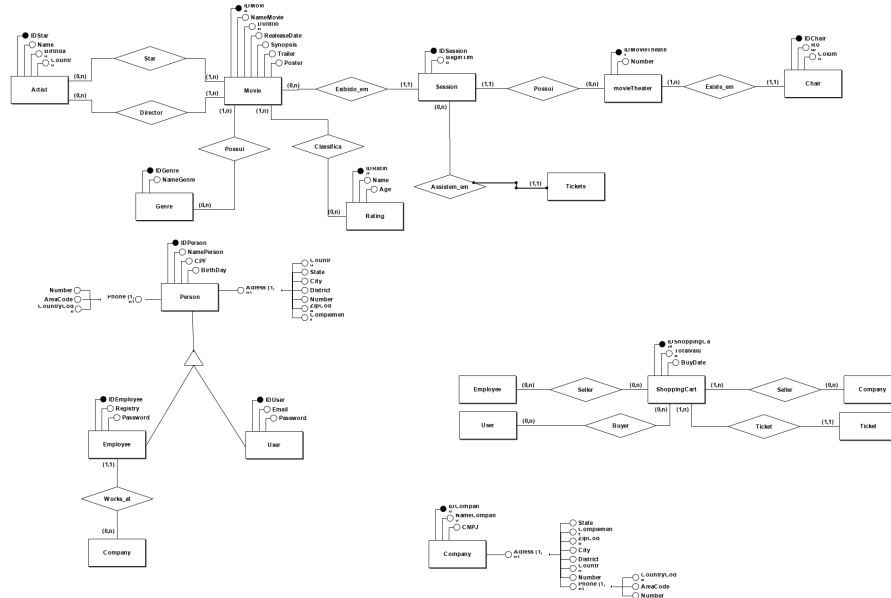


Figura 1: Modelo Entidade Relacionamento

4 Script Sql

Nesta seção mostramos o script sql para geração do banco de dados, que foi gerado utilizando o modelo acima e foi gerado automaticamente pelo MySQL.

```

1      — MySQL Script generated by MySQL Workbench
2      — Thu Jun 27 18:36:45 2019
3      — Model: New Model    Version: 2.0
4      — MySQL Workbench Forward Engineering
5
6      SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS,
UNIQUE_CHECKS=0;
7      SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS,
FOREIGN_KEY_CHECKS=0;
8      SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='
ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,
NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION
';
9
10
11     — Schema UnBCineFlix2
12
13     DROP SCHEMA IF EXISTS `UnBCineFlix2` ;
14
15
16     — Schema UnBCineFlix2
17

```

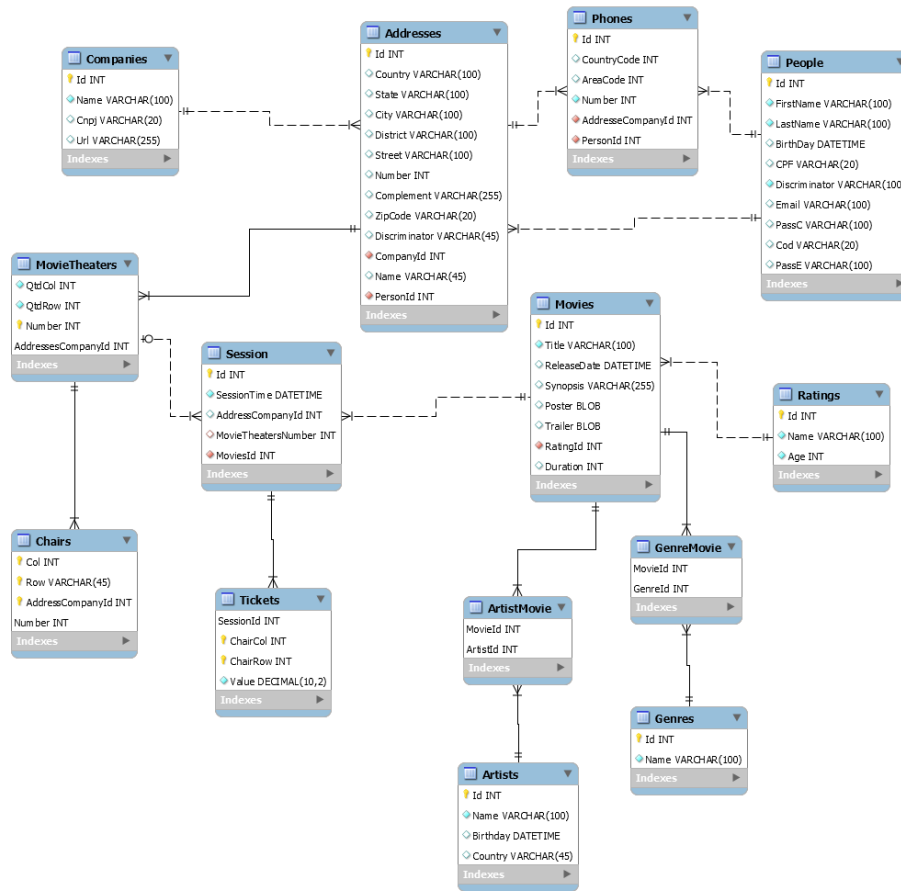


Figura 2: Modelo Relacional

```

18 CREATE SCHEMA IF NOT EXISTS `UnBCineFlix2` DEFAULT
   CHARACTER SET utf8 ;
19 USE `UnBCineFlix2` ;
20
21
22
23
24 CREATE TABLE IF NOT EXISTS `UnBCineFlix2`.``
   Addresses` (
25     `Id` INT NOT NULL AUTO_INCREMENT,
26     `Country` VARCHAR(100) NULL,
27     `State` VARCHAR(100) NULL,
28     `City` VARCHAR(100) NULL,
29     `District` VARCHAR(100) NULL,
30     `Street` VARCHAR(100) NULL,
31     `Number` INT NULL,
32     `Complement` VARCHAR(255) NULL,
33     `ZipCode` VARCHAR(20) NULL,
34     `Discriminator` VARCHAR(45) NULL,
35     `CompanyId` INT NOT NULL,

```

```

36         `Name` VARCHAR(45) NULL,
37         `PersonId` INT NOT NULL,
38         PRIMARY KEY (`Id`),
39         INDEX `fk_Addresses_People1_idx` (`PersonId`
` ASC) VISIBLE,
40         INDEX `fk_Addresses_Companies1_idx` (`
CompanyId` ASC) VISIBLE,
41         CONSTRAINT `fk_Addresses_People1`
42         FOREIGN KEY (`PersonId`)
43         REFERENCES `UnBCineFlix2`.`People`
(`Id`)
44         ON DELETE NO ACTION
45         ON UPDATE NO ACTION,
46         CONSTRAINT `fk_Addresses_Companies1`
47         FOREIGN KEY (`CompanyId`)
48         REFERENCES `UnBCineFlix2`.`
Companies` (`Id`)
49         ON DELETE NO ACTION
50         ON UPDATE NO ACTION)
51     ENGINE = InnoDB;
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
CREATE TABLE IF NOT EXISTS `UnBCineFlix2`.`
ArtistMovie` (
    `MovieId` INT NOT NULL,
    `ArtistId` INT NOT NULL,
    PRIMARY KEY (`MovieId`, `ArtistId`),
    INDEX `fk_Movie_has_Artist_Artist1_idx` (`
ArtistId` ASC) VISIBLE,
    INDEX `fk_Movie_has_Artist_Movie1_idx` (`
MovieId` ASC) VISIBLE,
    CONSTRAINT `fk_Movie_has_Artist_Movie1`
    FOREIGN KEY (`MovieId`)
    REFERENCES `UnBCineFlix2`.`Movies`
(`Id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
    CONSTRAINT `fk_Movie_has_Artist_Artist1`
    FOREIGN KEY (`ArtistId`)
    REFERENCES `UnBCineFlix2`.`Artists`
(`Id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

--
-- Table `UnBCineFlix2`.`Artists`
--
CREATE TABLE IF NOT EXISTS `UnBCineFlix2`.`Artists`
(
    `Id` INT NOT NULL,
    `Name` VARCHAR(100) NOT NULL,
    `Birthday` DATETIME NULL,
    `Country` VARCHAR(45) NULL,

```

```

84         PRIMARY KEY (`Id`))
85     ENGINE = InnoDB;
86
87     ---
88
89     --- Table `UnBCineFlix2`.`Chairs`
90     ---
91
92     CREATE TABLE IF NOT EXISTS `UnBCineFlix2`.`Chairs`
93     (
94         `Col` INT NOT NULL,
95         `Row` VARCHAR(45) NOT NULL,
96         `AddressCompanyId` INT NOT NULL,
97         `Number` INT NOT NULL,
98         PRIMARY KEY (`Col`, `Row`, `
99         AddressCompanyId`, `Number`),
100         INDEX `fk_Chairs_MovieTheaters1_idx` (`
101         AddressCompanyId` ASC, `Number` ASC) VISIBLE,
102         CONSTRAINT `fk_Chairs_MovieTheaters1`
103         FOREIGN KEY (`Number`)
104         REFERENCES `UnBCineFlix2`.`
105         MovieTheaters` (`Number`)
106         ON DELETE NO ACTION
107         ON UPDATE NO ACTION)
108     ENGINE = InnoDB;
109
110     ---
111
112     --- Table `UnBCineFlix2`.`Companies`
113     ---
114
115     CREATE TABLE IF NOT EXISTS `UnBCineFlix2`.`
116     Companies` (
117         `Id` INT NOT NULL AUTO_INCREMENT,
118         `Name` VARCHAR(100) NOT NULL,
119         `Cnpj` VARCHAR(20) NULL,
120         `Url` VARCHAR(255) NULL,
121         PRIMARY KEY (`Id`))
122     ENGINE = InnoDB;
123
124     ---
125
126     --- Table `UnBCineFlix2`.`GenreMovie`
127     ---
128
129     CREATE TABLE IF NOT EXISTS `UnBCineFlix2`.`
130     GenreMovie` (
131         `MovieId` INT NOT NULL,
132         `GenreId` INT ZEROFILL NOT NULL,
133         PRIMARY KEY (`MovieId`, `GenreId`),
134         INDEX `fk_Movie_has_Genre_Genre1_idx` (`
135         GenreId` ASC) VISIBLE,
136         INDEX `fk_Movie_has_Genre_Movie1_idx` (`
137         MovieId` ASC) VISIBLE,
138         CONSTRAINT `fk_Movie_has_Genre_Movie1`
139         FOREIGN KEY (`MovieId`)
140         REFERENCES `UnBCineFlix2`.`Movies`
141         (`Id`)
142         ON DELETE NO ACTION

```

```

131         ON UPDATE NO ACTION,
132         CONSTRAINT `fk_Movie_has_Genre_Genre1`
133         FOREIGN KEY (`GenreId`)
134         REFERENCES `UnBCineFlix2`.`Genres`
135         (`Id`)
136         ON DELETE NO ACTION
137         ON UPDATE NO ACTION)
138     ENGINE = InnoDB;
139
140     ---
141     --- Table `UnBCineFlix2`.`Genres`
142     ---
143
144     CREATE TABLE IF NOT EXISTS `UnBCineFlix2`.`Genres`
145     (
146         `Id` INT ZEROFILL NOT NULL,
147         `Name` VARCHAR(100) NOT NULL,
148         PRIMARY KEY (`Id`))
149     ENGINE = InnoDB;
150
151     ---
152     --- Table `UnBCineFlix2`.`MovieTheaters`
153     ---
154
155     CREATE TABLE IF NOT EXISTS `UnBCineFlix2`.`
156     MovieTheaters` (
157         `QtdCol` INT NOT NULL,
158         `QtdRow` INT NOT NULL,
159         `Number` INT NOT NULL,
160         `AddressesCompanyId` INT NOT NULL,
161         PRIMARY KEY (`Number`, `AddressesCompanyId`
162         `),
163         INDEX `fk_MovieTheaters_Addresses1_idx` (`
164         AddressesCompanyId` ASC) VISIBLE,
165         CONSTRAINT `fk_MovieTheaters_Addresses1`
166         FOREIGN KEY (`AddressesCompanyId`)
167         REFERENCES `UnBCineFlix2`.`
168         Addresses` (`Id`)
169         ON DELETE NO ACTION
170         ON UPDATE NO ACTION)
171     ENGINE = InnoDB;
172
173     ---
174     --- Table `UnBCineFlix2`.`Movies`
175     ---
176
177     CREATE TABLE IF NOT EXISTS `UnBCineFlix2`.`Movies`
178     (
179         `Id` INT NOT NULL AUTO_INCREMENT,
180         `Title` VARCHAR(100) NOT NULL,
181         `ReleaseDate` DATETIME NULL,
182         `Synopsis` VARCHAR(255) NULL,
183         `Poster` BLOB NULL,
184         `Trailer` BLOB NULL,
185         `RatingId` INT NOT NULL,
186         `Duration` INT NULL,

```

```

180 PRIMARY KEY (`Id`),
181 INDEX `fk_Movie_Rating1_idx` (`RatingId`
ASC) VISIBLE,
182 CONSTRAINT `fk_Movie_Rating1`
183 FOREIGN KEY (`RatingId`)
184 REFERENCES `UnBCineFlix2`.`Ratings`
(`Id`)
185 ON DELETE NO ACTION
186 ON UPDATE NO ACTION)
187 ENGINE = InnoDB;
188
189
190
-----
191 --- Table `UnBCineFlix2`.`People`
192 ---
-----
193 CREATE TABLE IF NOT EXISTS `UnBCineFlix2`.`People`
(
194 `Id` INT NOT NULL AUTO_INCREMENT,
195 `FirstName` VARCHAR(100) NOT NULL,
196 `LastName` VARCHAR(100) NOT NULL,
197 `BirthDay` DATETIME NULL,
198 `CPF` VARCHAR(20) NULL,
199 `Discriminator` VARCHAR(100) NOT NULL,
200 `Email` VARCHAR(100) NULL,
201 `PassC` VARCHAR(100) NULL,
202 `Cod` VARCHAR(20) NULL,
203 `PassE` VARCHAR(100) NULL,
204 PRIMARY KEY (`Id`))
205 ENGINE = InnoDB;
206
207
208
-----
209 --- Table `UnBCineFlix2`.`Phones`
210 ---
-----
211 CREATE TABLE IF NOT EXISTS `UnBCineFlix2`.`Phones`
(
212 `Id` INT NOT NULL AUTO_INCREMENT,
213 `CountryCode` INT NULL,
214 `AreaCode` INT NULL,
215 `Number` INT NOT NULL,
216 `AdresseCompanyId` INT NOT NULL,
217 `PersonId` INT NOT NULL,
218 PRIMARY KEY (`Id`),
219 INDEX `fk_Phones_Addresses1_idx` (`
AdresseCompanyId` ASC) VISIBLE,
220 INDEX `fk_Phones_People1_idx` (`PersonId`
ASC) VISIBLE,
221 CONSTRAINT `fk_Phones_Addresses1`
222 FOREIGN KEY (`AdresseCompanyId`)
223 REFERENCES `UnBCineFlix2`.`
Addresses` (`Id`)
224 ON DELETE NO ACTION
225 ON UPDATE NO ACTION,
226 CONSTRAINT `fk_Phones_People1`
227 FOREIGN KEY (`PersonId`)
228 REFERENCES `UnBCineFlix2`.`People`
(`Id`)
229 ON DELETE NO ACTION

```

```

230                                     ON UPDATE NO ACTION)
231     ENGINE = InnoDB;
232
233
234     ---

```

```

235     --- Table `UnBCineFlix2`.`Ratings`
236     ---

```

```

237     CREATE TABLE IF NOT EXISTS `UnBCineFlix2`.`Ratings`
238     (
239         `Id` INT NOT NULL AUTO_INCREMENT,
240         `Name` VARCHAR(100) NOT NULL,
241         `Age` INT NOT NULL,
242         PRIMARY KEY (`Id`))
243     ENGINE = InnoDB;
244
245     ---

```

```

246     --- Table `UnBCineFlix2`.`Session`
247     ---

```

```

248     CREATE TABLE IF NOT EXISTS `UnBCineFlix2`.`Session`
249     (
250         `Id` INT NOT NULL AUTO_INCREMENT,
251         `SessionTime` DATETIME NOT NULL,
252         `AddressCompanyId` INT NULL,
253         `MovieTheatersNumber` INT NULL,
254         `MoviesId` INT NOT NULL,
255         PRIMARY KEY (`Id`),
256         INDEX `fk_Session_MovieTheaters1_idx` (`
257         AddressCompanyId` ASC, `MovieTheatersNumber` ASC) VISIBLE,
258         INDEX `fk_Session_Movies1_idx` (`MoviesId`
259         ASC) VISIBLE,
260         CONSTRAINT `fk_Session_MovieTheaters1`
261         FOREIGN KEY (`MovieTheatersNumber`)
262         REFERENCES `UnBCineFlix2`.`
263         MovieTheaters` (`Number`)
264         ON DELETE NO ACTION
265         ON UPDATE NO ACTION,
266         CONSTRAINT `fk_Session_Movies1`
267         FOREIGN KEY (`MoviesId`)
268         REFERENCES `UnBCineFlix2`.`Movies`
269         (`Id`)
270         ON DELETE NO ACTION
271         ON UPDATE NO ACTION)
272     ENGINE = InnoDB;
273
274     ---

```

```

271     --- Table `UnBCineFlix2`.`Tickets`
272     ---

```

```

273     CREATE TABLE IF NOT EXISTS `UnBCineFlix2`.`Tickets`
274     (
275         `SessionId` INT NOT NULL,
276         `ChairCol` INT NOT NULL,
277         `ChairRow` INT NOT NULL,
278         `Value` DECIMAL(10,2) NOT NULL,

```



```

278 PRIMARY KEY ( `SessionId`, `ChairCol`, `
ChairRow` ),
279 INDEX `fk_Tickets_Session1_idx` ( `SessionId
` ASC) VISIBLE,
280 CONSTRAINT `fk_Tickets_Session1`
281 FOREIGN KEY ( `SessionId` )
282 REFERENCES `UnBCineFlix2` ( `Session`
( `Id` )
283 ON DELETE NO ACTION
284 ON UPDATE NO ACTION)
285 ENGINE = InnoDB;
286
287
288 SET SQL_MODE=@OLD_SQL_MODE;
289 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
290 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;

```

5 Álgebra relacional

6 Avaliação das formas normais