



PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE  
ESCUELA DE INGENIERÍA  
DEPARTAMENTO DE CIENCIA DE LA COMPUTACIÓN  
SEGUNDO SEMESTRE DE 2022

Bases de Datos - IIC2413.

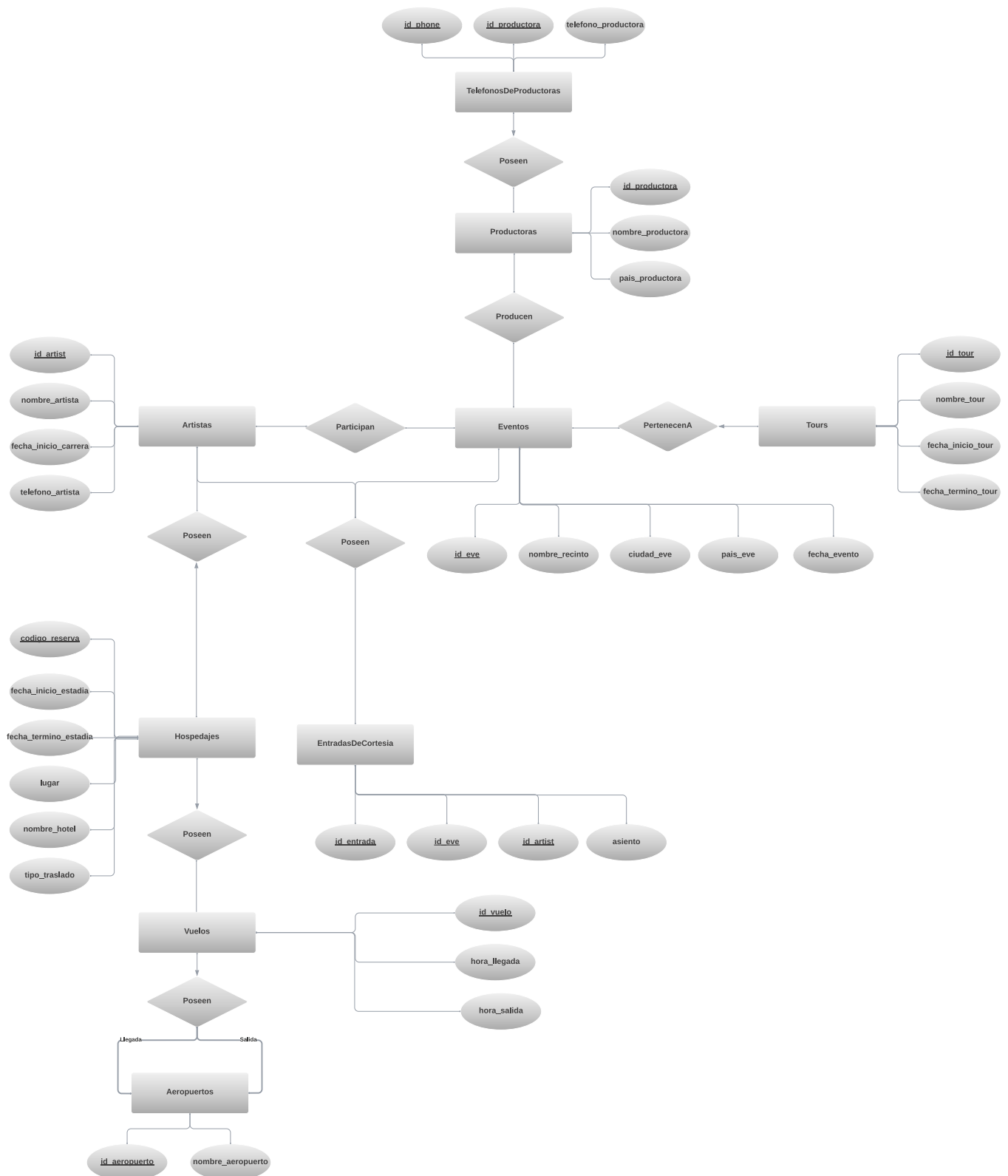
**Entrega 2 - Grupo 67.**

Fátima Valenzuela G. — Oscar Herrera G.

Sección: 1.

Fecha: 9 de Noveimbre, 2022.

# 1. Primera Parte: Diagrama Entidad-Relación.



## 2. Segunda: Esquema preliminar de la base de datos.

### 2.1. Esquema relacional de tablas.

A continuación, se despliega el esquema para cada una de las tablas (relaciones) presente en nuestra base de datos:

- Artistas(id\_artist int, nombre\_artista varchar(50), fecha\_inicio\_carrera date, telefono\_artista varchar(15))
- Eventos(id\_eve int, nombre\_evento varchar(50), nombre\_recinto varchar(50), ciudad\_eve varchar(50), pais\_eve varchar(50), fecha\_evento date NOT NULL)
- Productoras(id\_productora int, nombre\_productora varchar(150), pais\_productora varchar(150))
- TelefonosDeProductoras(id\_phone int, Productora.id\_productora int, telefono\_productora varchar(15))
- Participaciones(Eventos.id\_eve int, Artistas.id\_artist int)
- ProduccionDelEvento(Productora.id\_productora int, Evento.id\_eve int)
- Tours(id\_tour int, nombre\_tour varchar(50) NOT NULL, fecha\_inicio\_tour date NOT NULL, fecha\_termino\_tour date NOT NULL)
- EventosDeTours(Eventos.id\_eve int, Toursid\_tour int)
- EntradasDeCortesia(id\_entrada int, Eventos.id\_eve int, Artistas.id\_artist int, asiento varchar(30))
- Hospedajes(codigo\_reserva varchar(9), Artistas.id\_artist int, fecha\_inicio\_estadia date, fecha\_termino\_estadia date, lugar varchar(50), nombre\_hotel varchar(50), tipo\_traslado varchar(30))
- Vuelos(id\_vuelo int, Hospedajes.codigo\_reserva varchar(9), Aeropuertos.id\_aeropuerto\_salida int, hora\_salida time, Aeropuertos.id\_aeropuerto\_llegado int, hora\_llegada time)
- Aeropuertos(id\_aeropuerto int, nombre\_aeropuerto varchar(50) NOT NULL)

### 2.2. Supuestos y Consideraciones.

A continuación, se enlistan observaciones en los datos que han determinado nuestra forma de modelar:

- Existen eventos distintos con mismo nombre, pero que difieren en fecha. Eliminamos aquellos eventos que no tienen fecha asociada, porque no se pudo distinguir para luego asignarle un id.
- Dado lo anterior, no es posible identificar eventos de entradas de cortesía que poseen mismo nombre. Para esto, asignamos el mismo artista y asiento a entradas de cortesía para ambos eventos.

- Existen eventos con más de una productora y por eso se creó una tabla aparte.
- Existen productoras con más de un teléfono y por eso se creó una tabla aparte.
- Si al separar tipo de traslado en 3 tablas apartes, no podíamos asegurar que un Hospedaje perteneciera a sólo una observación dentro de las 3 tablas, por lo tanto, a pesar de la redundancia, dejamos el atributo en la tabla de Hospedajes para asegurar que un hospedaje sólo tenga 1 traslado asociado.

### 3. Dependencias Funcionales, Forma Normal.

A continuación, las formas funcionales no violan la tercera forma normal:

- La fecha de llegada del vuelo, es la fecha de Hospedaje, que determina la del vuelo.

### 4. Consultas SQL.

#### 4.1. Consulta 1.

un listado del nombre y teléfono de contacto de todos los artistas:

```
SELECT nombre_artista , telefono_artista FROM Artistas;
```

#### 4.2. Consulta 2.

Dado un artista, entregue el número de entradas de cortesía que ha entregado:

```
SELECT nombre_artista , COUNT(nombre_artista) AS cantidad
FROM EntradasDeCortesia AS EN
INNER JOIN Artistas ON EN.id_artist = Artistas.id_artist
WHERE nombre_artista = '$arti'
GROUP BY nombre_artista
ORDER BY cantidad DESC;
```

#### 4.3. Consulta 3.

Dado un artista, entregue los datos de su ultimo tour (el más reciente):

```
SELECT Tours.id_tour , nombre_tour , fecha_inicio_tour , fecha_termino_tour
FROM Tours
INNER JOIN EventosDeTours AS EVT ON EVT.id_tour = Tours.id_tour
INNER JOIN Eventos ON EVT.id_eve = Eventos.id_eve
INNER JOIN Participaciones AS PAR ON EVT.id_eve = PAR.id_eve
```

```

INNER JOIN Artistas ON PAR.id_artist = Artistas.id_artist
WHERE nombre_artista = '$artista'
ORDER BY fecha_evento DESC
LIMIT 1;

```

Un ejemplo que sirve es Paloma Mami en nuestros datos

#### 4.4. Consulta 4.

Dado un tour, liste los paises que serán visitados en dicho tour

```

SELECT pais_eve
FROM EventosDeTours AS ET
INNER JOIN Tours ON ET.id_tour = Tours.id_tour
INNER JOIN Eventos AS EV ON ET.id_eve = EV.id_eve
WHERE nombre_tour = '$tour';

```

Un ejemplo que sirve es Anastasia en nuestros datos

#### 4.5. Consulta 5.

Dado un artista, liste todas las productoras con las que ha trabajado dicho artista

```

SELECT nombre_productora
FROM Artistas AS AR
INNER JOIN Participaciones ON AR.id_artist = Participaciones.id_artist
INNER JOIN ProduccionDelEvento ON Participaciones.id_eve = ProduccionDelEvento.id_eve
INNER JOIN Productoras ON ProduccionDelEvento.id_productora = Productoras.id_productora
WHERE nombre_artista = '$artista';

```

Un ejemplo que sirve es Kudai en nuestros datos

#### 4.6. Consulta 6.

Dado un artista, liste todos los hoteles en los que se ha hospedado y cuantas veces se ha hospedado en cada uno (con codigos de reserva distinto, no cantidad de noches)

```

SELECT nombre_artista, nombre_hotel, Count(nombre_artista) AS cantidad
FROM HOSPEDAJES AS HOS
INNER JOIN Artistas ON HOS.id_artist = Artistas.id_artist
WHERE nombre_artista = '$artista'
GROUP BY nombre_artista, nombre_hotel
ORDER BY cantidad DESC;

```

Un ejemplo que sirve es Snow en nuestros datos

#### 4.7. Consulta 7.

Muestre al artista que ha entregado la mayor cantidad de entradas de cortesía

```
SELECT nombre_artista , COUNT(nombre_artista) AS cantidad
FROM EntradasDeCortesía AS EN
INNER JOIN Artistas ON EN.id_artist = Artistas.id_artist
GROUP BY nombre_artista
ORDER BY cantidad DESC LIMIT 1;
```