

## **Reporte Entrega 2**

Integrantes:

Sebastián Arriagada

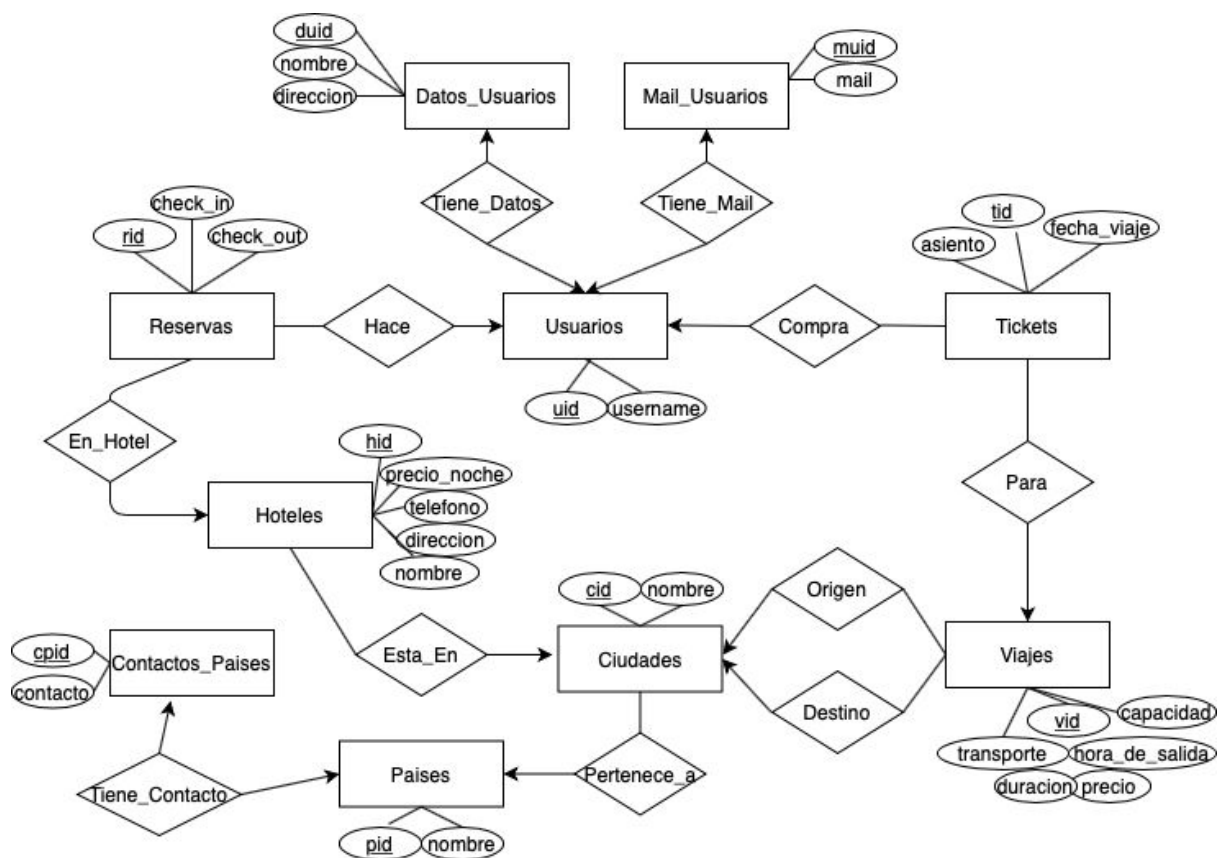
Tomás Castro

### **2.1 Crear modelo**

El modelo consiste en 21 tablas, las cuales 10 de ellas son entidades y 11 corresponden a relaciones. De esta manera, el modelo queda como:

- Usuarios(uid, username) .
- Datos\_Usuarios(duid: int, nombre: varchar(255), direccionusuario: varchar(255))
- Tiene\_Datos(duid: int, uid: int (FK))
- Mail\_Usuarios(muid: int, mail: varchar(255))
- Tiene\_Mail(muid, uid (FK))
- Ciudades(cid: int, nombre: varchar(255))
- Pertenece\_a(cid: int, pid: int (FK))
- Paises(pid: int, nombre: varchar(255))
- Contactos\_Paises(cpid: int, contacto: varchar(255))
- Tiene\_Contacto(pid: int (FK), cpid: int )
- Viajes(vid: int, horasalida: varchar(255), duracion: int, transporte: varchar(255), capacidad: int, precio: int)
- Origen(vid: int, cid: int (FK))
- Destino(vid: int, cid: int (FK))
- Tickets(tid: int, asiento: int, fechaviaje: date)
- Para(tid: int, vid: int(FK))
- Compra\_ticket(tid: int, uid: int (FK), fechacompra: date)
- Hoteles(hid: int, nombre: varchar(255), direccionhotel: varchar(255), telefono: varchar, precionoche: int)
- Esta\_en(hid: int, cid: int (FK))
- Reservas(rid: int, checkin: date, checkout: date)
- Hace(uid: int (FK), rid: int)
- En\_Hotel(rid: int, hid: int (FK))

Del esquema anterior, llegamos a el siguiente diagrama E/R:



## 2.2 Justificar modelo

El presente modelo ha sido diseñado según la normalización 3NF, donde las tablas están en 2NF y ningún atributo no primario de una tabla es transitivamente dependiente de una clave primaria. Es por esto la alta granularidad de nuestro esquema, donde se puede observar que sólo para definir los datos disponibles de los usuarios se han definido 3 tablas: Usuarios, Datos\_usuarios, Mail\_usuarios y 2 relaciones: Tiene\_datos y tiene\_mail. Gracias a esto se reduce la redundancia de nuestra base de datos.

En específico, a continuación se detallan las dependencias y detalles de las tablas:

- La tabla Usuarios sólo contiene la dependencia funcional de su llave primaria: uid determina el username del usuario.
- La tabla Datos\_Usuarios sólo contiene la dependencia funcional de su llave primaria: duid determina el nombre y la dirección de usuario determinada

- La tabla Tiene\_Datos sólo contiene la dependencia funcional de su llave primaria: uid determina el uid, con lo que se relaciona cada uid con los datos que tendrá cada uno.
- La tabla Mail\_Usuarios sólo contiene la dependencia funcional de su llave primaria: muid determina el mail del usuario.
- La tabla Tiene\_Mail sólo contiene la dependencia funcional de su llave primaria: muid determina el uid del usuario, con lo que se relaciona cada uid con el mail que tendrá cada uno. Esto se hace ya que cada mail es único, entonces si se ingresaba en la misma tabla que Datos\_Usuario, tendríamos dependencias funcionales dentro con algo que no sería llave.
- La tabla Ciudades sólo contiene la dependencia funcional de su llave primaria: cid determina el nombre de la ciudad.
- La tabla Pertenece\_a sólo contiene la dependencia funcional de su llave primaria: cid determina el pid. Esta tabla es una relación que nos permite encontrar en que país esta cada ciudad, o que ciudades tiene cada país.
- La tabla Países sólo contiene la dependencia funcional de su llave primaria: pid determina el nombre del país.
- La tabla Contacto\_Paises sólo contiene la dependencia funcional de su llave primaria: cpid determina el contacto del país. Esta tabla se crea ya que los contactos son únicos por país, entonces se tendría otra dependencia funcional.
- La tabla Tiene Contacto sólo contiene la dependencia funcional de su llave primaria: pid determina el cpid. Con esta tabla hacemos la relación de cada país con su contacto.
- La tabla Viajes contiene dependencias funcionales con su llave primaria: pid determina la horasalida, duracion, transporte, capacidad y precio del viaje.
- La tabla Origen sólo contiene la dependencia funcional de su llave primaria: vid determina el cid, que corresponde a la ciudad de origen del viaje.
- La tabla Destino sólo contiene la dependencia funcional de su llave primaria: vid determina el cid, que corresponde a la ciudad de destino del viaje.
- La tabla Tickets contiene la dependencia funcional de su llave primaria: tid determina el asiento y la fechaviaje.
- La tabla Para sólo contiene la dependencia funcional de su llave primaria: tid determina el vid. Esta tabla hace la relación entre el ticket y el viaje, y hace la relación entre el viaje que se realizará y que asientos y en qué fecha viajará.
- La tabla Compra\_ticket sólo contiene la dependencia funcional de su llave primaria: tid determina el uid y la fechacompra. Esta tabla corresponde a la relación entre el un ticket y un usuario, haciendo la relación de quien compro el ticket.
- La tabla Hoteles sólo contiene la dependencia funcional de su llave primaria: pid determina el nombre del hotel, la direccionhotel, el telefono y precionoche.
- La tabla Esta\_en sólo contiene la dependencia funcional de su llave primaria: hid determina el cid, La que corresponde a la relación entre un hotel y la ciudad en la que este se encuentra.
- La tabla Reservas sólo contiene la dependencia funcional de su llave primaria: rid determina el checkin y el checkout.

- La tabla Hace sólo contiene la dependencia funcional de su llave primaria: rid determina el uid. Esta tabla es una relación entre el usuario y las reservas, y cada reserva tiene asignado un único usuario.
- La tabla En\_hotel sólo contiene la dependencia funcional de su llave primaria: rid determina el hid. Esta tabla es una relación que determina en qué hotel se ha hecho la reserva.

## **CONSULTAS:**

Las consultas 1 a 5 fueron implementadas. 6 tuvimos problemas así que no pudimos realizarla.