



BASES DE DATOS
Segundo Cuatrimestre de 2022
Proyecto N° 2
Implementación de una base de datos

1. Ejercicios

Implemente (ver condiciones de entrega items [I](#) y [II](#)) una base de datos en MySQL respetando el modelo relacional que acompaña este enunciado (ver apéndice [A](#)). El nombre de dicha base de datos deberá ser “*vuelos*” y los esquemas de las tablas deberán respetar los esquemas propuestos por dicho modelo relacional. Se deberán respetar los nombres de las relaciones y los atributos, así como las restricciones de llaves primarias y foráneas. No se aceptarán bases de datos que no respeten estas convenciones (ver sección [2](#), **verificación de la bases de datos**).

Además deberá crear los siguientes usuarios:

- *admin*: Este usuario estará encargado de administrar la base de datos “*vuelos*” por lo tanto deberá tener acceso total sobre todas las tablas, con la opción de crear usuarios y otorgar privilegios sobre las mismas. Para no comprometer la seguridad se restringirá que el acceso de este usuario se realice solo desde la máquina local donde se encuentra el servidor MySQL. El password de este usuario deberá ser *admin*.
- *empleado*: Este usuario estará destinado a las consultas y reservas de vuelos. Para las consultas necesitará acceso de lectura sobre todas las tablas de la b.d. *vuelos*. Para las reservas necesitará además privilegios para ingresar, modificar y borrar datos sobre las tablas *reservas*, *pasajeros* y *reserva_vuelo_clase*. Dado que la empresa cuenta con varios puestos de atención al público distribuidos en diferentes sucursales, este usuario deberá poder conectarse desde cualquier dominio. El password del *empleado* deberá ser *empleado*. **Importante:** Recuerde eliminar el usuario *vacío* (`drop user ''@localhost`) para poder conectarse con el usuario *empleado* desde localhost.
- *cliente*: Este usuario está destinado a permitir un acceso público para satisfacer las consultas sobre vuelos de los clientes de la empresa, ya sean tanto personas como agencias de viajes. El *cliente* tendrá una visión restringida de la base de datos que solamente le permita ver la disponibilidad de vuelos ocultando la estructura de la base de datos. A tal efecto, se deberá crear una *vista* con el nombre *vuelos_disponibles* que contenga la siguiente *información sobre cada instancia de vuelo* (los nombres de los campos deberán ser los indicados entre paréntesis):
 - Número de vuelo (*nro_vuelo*), modelo de avión (*modelo*), fecha (*fecha*), día de salida (*dia_sale*), hora de salida (*hora_sale*), hora de llegada (*hora_llega*) y tiempo estimado de vuelo (*tiempo_estimado*). Puede asumir que no hay vuelos que duren más de 24 hs. para calcular el tiempo estimado de vuelo.
 - Código, nombre, ciudad, estado y país del aeropuerto tanto de salida (*codigo_aero_sale*, *nombre_aero_sale*, *ciudad_sale*, *estado_sale*, *pais_sale*) como de llegada (*codigo_aero_llega*, *nombre_aero_llega*, *ciudad_llega*, *estado_llega*, *pais_llega*).
 - Precio del pasaje (*precio*) y cantidad de asientos disponibles (*asientos_disponibles*) en cada clase (*clase*) brindada por la salida asociada a la instancia de vuelo.Para calcular el número de asientos disponibles en cada clase deberá restar la cantidad asientos reservados para la instancia de vuelo y la clase, a la suma de la cantidad de asientos

que brinda la salida asociada a la instancia de vuelo en esa clase más el porcentaje asociado a la clase. Por ejemplo, si la salida asociada a una instancia de vuelo en clase *turista* brinda 100 asientos, el porcentaje asociado a la clase *turista* es 15 % y la instancia de vuelo tiene 50 asientos reservados en clase *turista* entonces la cantidad de asientos disponibles en clase *turista* en dicha instancia de vuelo es: $(100 + 15) - 50 = 65$

El usuario *cliente* solo tendrá privilegio de lectura sobre la vista *vuelos_disponibles*, deberá poder conectarse desde cualquier dominio y su password deberá ser *cliente*.

2. Fechas y condiciones de entrega

- **fecha límite de entrega:** 20 de Septiembre de 2022 a través del curso **Moodle** de la materia utilizando la [tarea correspondiente](#) habilitada para tal fin, se deberán subir:
 - I. un archivo de texto con el nombre **“vuelos.sql”** con la secuencia de sentencias para la creación de la base de datos, las tablas, la vista, los usuarios con nombre, password y privilegios correspondientes.
 - II. un archivo de texto con el nombre **“datos.sql”** con una carga inicial de datos de prueba adecuados como para poder realizar consultas significativas sobre ellos y probar la vista.
- **verificación de la base de datos:** En la sección del proyecto 2 del curso Moodle, estará disponible para bajar un programa llamado *verificar*. Este programa realiza una verificación sobre la estructura de la base de datos *ya creada*, y muestra un listado con los errores (si los tuviera) que esta presenta con respecto al modelo relacional propuesto en el apéndice [A](#) . Este programa debe ejecutarse con el servidor de MySQL corriendo, una vez creada la base de datos. **No se aceptarán bases de datos que no pasen correctamente la verificación realizada por este programa.**
- **Comisiones:** Los proyectos deben realizarse en comisiones de *dos alumnos* cada una y los datos de sus integrantes deberán ser informados en [este formulario](#) hasta el 6 de septiembre. Las comisiones deberán ser *las mismas* para todas las entregas.
- **Importante:** La entrega en *tiempo y forma* de este proyecto es *condición de cursado* de la materia.

A. Modelo Relacional de la base de datos “vuelos”

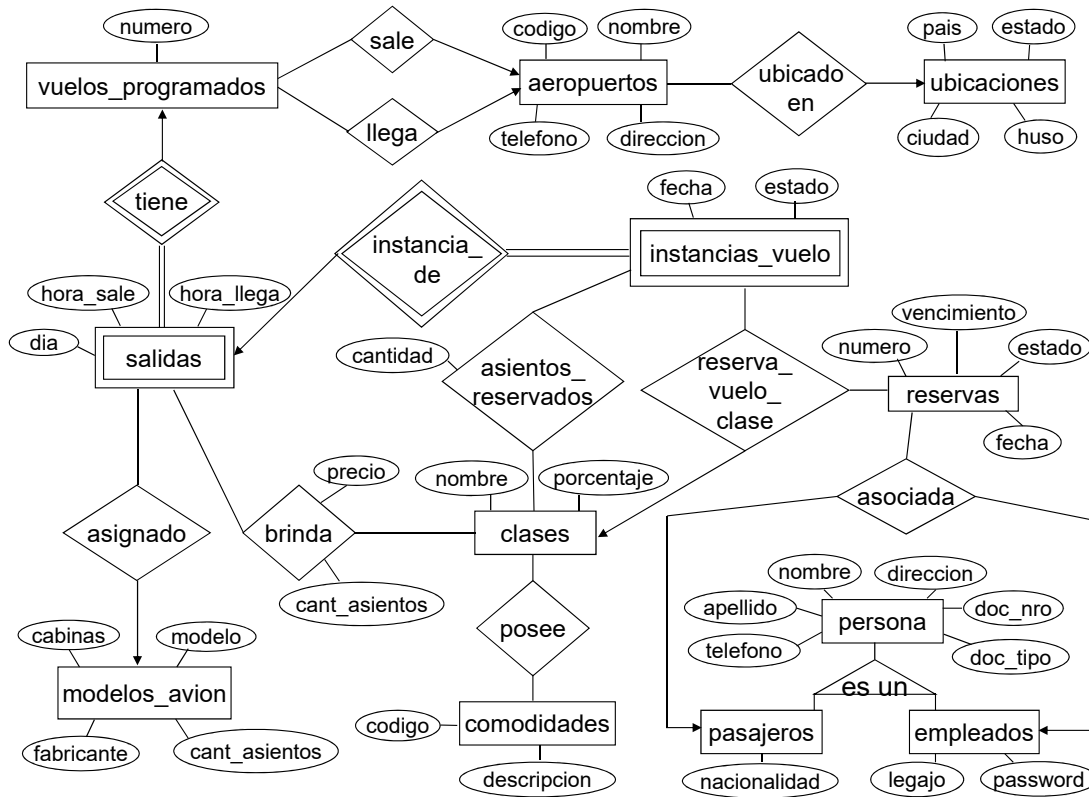


Figura 1: Modelo Entidad-Relación base de datos “vuelos”

- **vuelos_programados**(numero, aeropuerto_salida, aeropuerto_llegada)
numero es una cadena de caracteres; aeropuerto_salida y aeropuerto_llegada corresponden a códigos de aeropuertos
- **salidas**(vuelo, dia, hora_sale, hora_llega, modelo_avion)
vuelo corresponde a un numero de vuelo programado; dia = {'Do', 'Lu', 'Ma', 'Mi', 'Ju', 'Vi', 'Sa'}; modelo_avion corresponde a un modelo de modelos_avion
- **instancias_vuelo**(vuelo, fecha, dia, estado)
vuelo y dia corresponden a una salida; estado es una cadena de caracteres
- **aeropuertos**(codigo, nombre, telefono, direccion, pais, estado, ciudad)
codigo, nombre, telefono y direccion son cadenas de caracteres; pais, estado y ciudad corresponden a una ubicación
- **ubicaciones**(pais, estado, ciudad, huso)
Pais, estado y ciudad son cadenas de caracteres; huso es un entero en el intervalo [-12,12]
- **modelos_avion**(modelo, fabricante, cabinas, cant_asientos)
modelo y fabricante son cadenas de caracteres; cabinas y cant_asientos son enteros positivos
- **clases**(nombre, porcentaje)
nombre es una cadena de caracteres y porcentaje es un numero real en el intervalo [0, 0.99]

- **comodidades**(codigo, descripcion)
codigo es un número natural y descripcion es una cadena de caracteres
- **pasajeros**(doc_tipo, doc_nro, apellido, nombre, direccion, telefono, nacionalidad)
doc_nro es un número natural; el resto de los atributos son cadenas de caracteres
- **empleados**(legajo, password, doc_tipo, doc_nro, apellido, nombre, direccion, telefono)
doc_nro y legajo son números naturales, el resto de los atributos son cadenas de caracteres. El campo password debe ser una cadena de 32 caracteres, para poder almacenarlo de forma segura utilizando la función de hash MD5 provista por MySQL
- **reservas**(numero, fecha, vencimiento, estado, doc_tipo, doc_nro, legajo)
numero es un natural; fecha y vencimiento son fechas; estado es una cadena de caracteres; doc_tipo, doc_numero corresponden a un pasajero; legajo corresponde a un empleado
- **brinda**(vuelo, dia, clase, precio, cant_asientos)
vuelo y dia corresponden a una salida; clase corresponde al nombre de una clase; precio es un real positivo con 2 dígitos decimales y 5 dígitos en la parte entera; cant_asientos es un natural
- **posee**(clase, comodidad)
clase corresponde al nombre de una clase y comodidad corresponde al código de una comodidad
- **reserva_vuelo_clase**(numero, vuelo, fecha_vuelo, clase)
numero corresponde a un número de reserva; vuelo y fecha_vuelo corresponden a una instancia de vuelo; clase corresponde al atributo nombre de clases
- **asientos_reservados**(vuelo, fecha, clase, cantidad)
vuelo y fecha corresponden a una instancia de vuelo, clase corresponde al nombre de una clase y cantidad es un número natural