



Consejo de Educación
Técnico - Profesional
(Universidad del Trabajo del Uruguay)



Instituto Tecnológico Superior
UTU

Tercera entrega

B.D II



TECHROAD

We Create, We Design, We Develop

Solicitante: I.T.S. – Instituto Tecnológico Superior Arias - Balparda

Nombre de Fantasía del Proyecto: TECHROAD

Grupo de Clase: 3 ID

Turno: Vespertino

Materia: Base de Datos

Nombre de los Integrantes del Grupo: Bravo Fabrizio, De Mello Ignacio, Muñiz Manuel (representante), Robaina Agustín.

Fecha de entrega: 13/11/2020

Instituto Tecnológico Superior F. Arias – L. Balparda
Gral. Flores 3591 esq. Bvar. José Batlle y Ordoñez - Montevideo

Objetivo

El objetivo de este documento es especificar con total claridad todos las entidades, atributos y relaciones que se ubicarán dentro del modelo entidad relación para luego realizar un pasaje a tablas y normalizarlas hasta tercera forma normal para que no exista ninguna redundancia en los datos. Desde esa base podremos crear la Base de datos que se utilizara en este proyecto la cual trabajara en conjunto con la aplicación que se nos fue pedida.

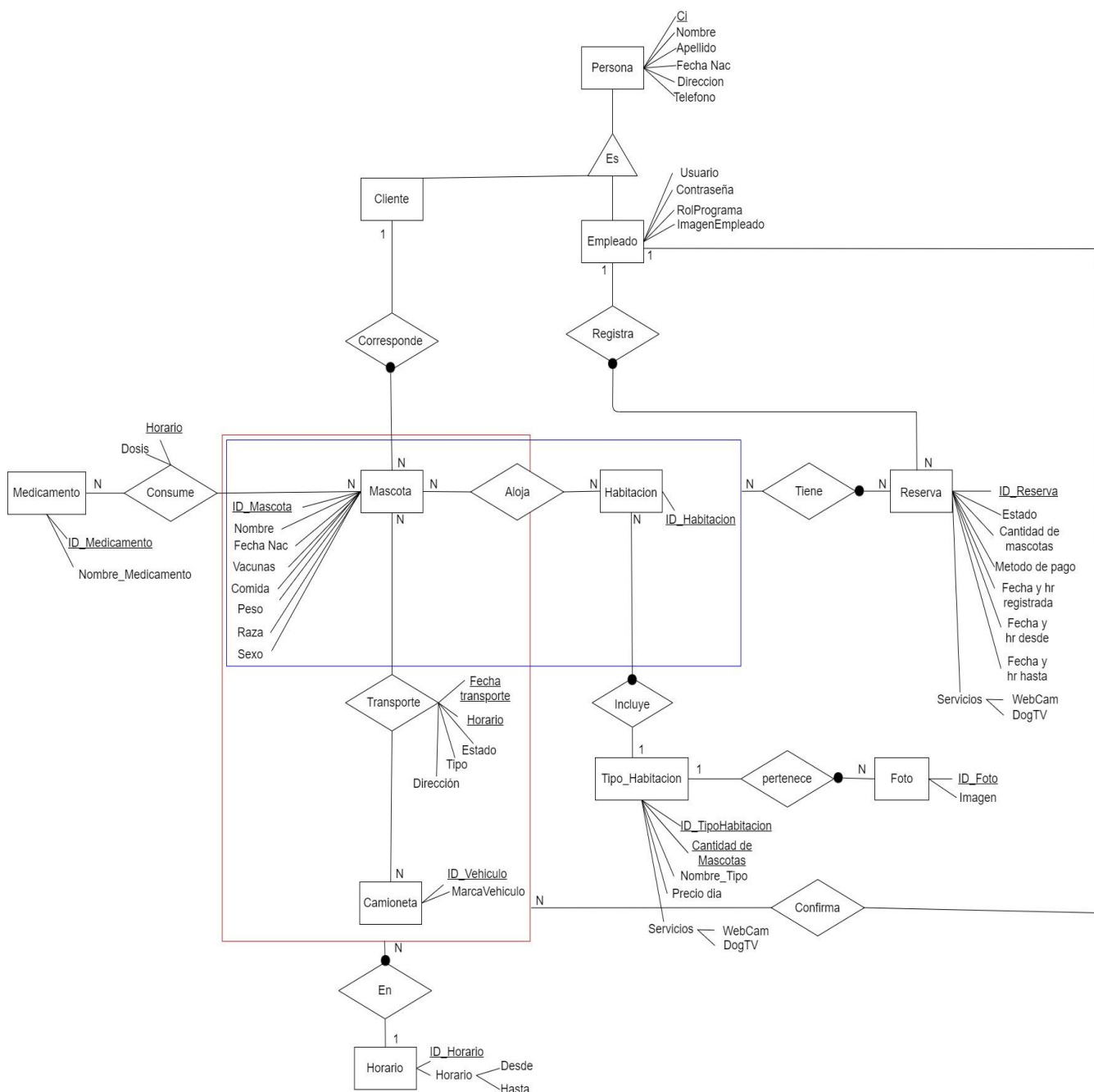
Alcance

A partir de dicho documento tanto los presentes como futuros lectores podrán entender los respectivos atributos que se ubicarán dentro de la base de datos y como los diferentes datos dentro de la misma se relacionan entre ellos.

Índice

1. M.E.R DE LA SITUACIÓN PLANTEADA.....	4
1.1. RNE.....	5
2. ESQUEMA RELACIONAL (E.R.).....	7
2.1. PASAJE A TABLAS	7
2.2. NORMALIZACIÓN (3FN)	9
3. JUSTIFICACIÓN DE CAMBIOS EN EL MER	10
4. DICCIONARIO DE DATOS.....	11
5. SENTENCIAS DDL Y DML.....	19
6. USUARIOS.....	27
6.1. CREACIÓN DE LOS USUARIOS	27
7. PERMISOS	28
7.1. PERMISOS DE LOS DIFERENTES ROLES	28
7.2. MATRIZ DE CONTROL DE ACCESO	30
7.3. SENTENCIAS UTILIZADAS PARA ASIGNAR PERMISOS.....	31
8. CONSULTAS S.Q.L. SOLICITADAS POR EL DOCENTE.....	33
9. RESPALDO DE DATOS.....	35
HOJA TESTIGO	36

1. M.E.R de la situación planteada



1.1. RNE

- Cliente \cup Empleado \neq Persona
- Fecha y hr registrada \leq Fecha y hr desde $<$ Fecha y hr hasta
- Empleado.RolPrograma = {"G", "O", "R"}
- "G" = Gerente
- "O" = Oficinista
- "R" = Recepcionista de Garaje
- Reserva.Estado = {"P", "C", "CA", "T"}
- "P" = Pendiente
- "C" = Confirmada
- "CA" = Cancelada
- "T" = Terminada
- Mascota.sexo = {"H", "M"}
- "H" = Hembra
- "M" = Macho
- Mascota.Vacunas= {"T", "F"}
- "T" = Todas
- "F" = Faltantes
- Reserva.Cantidad de Mascotas ≤ 3
- Tipo_Habitacion.Cantidad de Mascotas ≤ 3
- Tipo_Habitacion.Nombre_Tipo = {"S", "J", "M", "E"}
- "S" = Suite
- "J" = Junior Suite
- "M" = Mini Suite
- "E" = Estándar
- Realiza.MetodoDePago = {"T", "C"}
- "T" = Tarjeta de crédito
- "C" = Contado
- Transporte.Horario = {"M", "Ta", "N"}
- "M" = Mañana
- "Ta" = Tarde
- "N" = Noche
- Transporte.Estado = {"P", "C", "F"}
- "P" = Pendiente
- "C" = Confirmado
- "F" = Finalizado
- Transporte.Tipo = {"LL", "T"}
- "LL" = Lleva
- "T" = Trae
- Horario.Horario = {"11am-13pm", "14pm-16:30pm", "17pm-19pm"}
- En la habitación Mini Suite el peso de la mascota debe ser igual o menor a 12 KG.
- Solo existen 2 camionetas.

-
- Cada camioneta puede llevar un máximo de 3 perros por viaje.
 - Cantidad de días de alojamiento = (Fch y hr hasta – Fch y hr desde)
 - Monto a pagar = Cantidad de días de alojamiento * Tipo_Habitacion.Precio Dia
 - Fecha de transporte y Fecha y hr hasta deben estar entre los horarios de la entidad Horario
 - Empleado.RolPrograma es utilizado para validar el Rol del empleado que está utilizando la aplicación y para que pueda acceder a ciertas ventanas en específico dependiendo de su Rol.
 - Solo los Empleados con rol de Gerente y Oficinista podrán registrar Reservas.
 - Solo los Empleados con rol de Gerente y Recepcionista de Garaje podrán confirmar transportes.

2. Esquema Relacional (E.R.)

2.1. Pasaje a tablas

Persona(CI, Nombre, Apellido, Fecha Nac, Dirección, Telefono)

Cliente(CICliente)

CICliente → Persona

Empleado(CIEmpleado, Usuario, Contraseña, RolPrograma Enum("G", "O", "R"), ImagenEmpleado)

CIEmpleado → Persona

Mascota(ID_Mascota, Nombre, FechaNac, Vacunas Enum("T", "F"), Comida, Peso, Raza, Sexo Enum("H", "M"), CICliente)

CICliente → Cliente

Medicamento(ID_Medicamento, Nombre_Medicamento)

Consume(Horario, ID_Mascota, ID_Medicamento, Dosis)

ID_Mascota → Mascota

ID_Medicamento → Medicamento

Camioneta(ID_Vehículo, MarcaVehículo)

Transporte(ID_Mascota, ID_Vehículo, Fecha de transporte, Horario Enum ("M", "Ta", "N"), Estado Enum("P", "C", "F"), Tipo Enum("LL", "T"), Dirección, ID_Horario)

ID_Horario → Horario

ID_Mascota → Mascota

ID_Vehículo → Camioneta

Horario(ID_Horario, Horario {Desde, Hasta})

Confirma(ID_Mascota, ID_Vehículo, Fecha de transporte, Horario Enum ("M", "Ta", "N"), CIEmpleado)

ID_Mascota, ID_Vehículo, Fecha de transporte, Horario Enum("M", "Ta", "N") → Transporte

CIEmpleado → Empleado

Habitación(ID_Habitación, ID_TipoHabitación)

ID_TipoHabitación → Tipo_Habitación

Tipo_Habitación(ID_TipoHabitación, Cantidad de mascotas, Nombre_Tipo Enum("S", "J", "M", "E"), Precio día, Servicios {WebCam, DogTV})

ID_TipoHabitación → Tipo_Habitación

Foto(ID Foto, Imagen, ID_TipoHabitación)

ID_TipoHabitación → Tipo_Habitación

Aloja(ID_Mascota, ID_Habitación)

ID_Mascota → Mascota

ID_Habitación → Habitación

Tiene(ID_Mascota, ID_Habitación, ID_Reserva)

ID_Mascota, ID_Habitación → Aloja

ID_Reserva → Reserva

Reserva(ID_Reserva, Estado Enum (“P”, “C”, ”CA”, ”T”), Cantidad de mascotas, Fecha y hr registrada, Fecha y hr desde, Fecha y hr hasta, Metodo de pago Enum (“C”, “T”), Servicios {WebCam, DogTV}), CIEmpleado)
CIEmpleado→CIEmpleado

2.2. Normalización (3FN)

Persona(CI, Nombre, Apellido, Fecha Nac, Dirección, Telefono)

Cliente(CICliente)

CICliente → Persona

Empleado(CIEmpleado, Usuario, Contraseña, RolPrograma, ImagenEmpleado)

CIEmpleado → Persona

Mascota(ID_Mascota, Nombre, FechaNac, Vacunas, Comida, Peso, Raza, Sexo, CICliente)

CICliente → Cliente

Medicamento(ID_Medicamento, Nombre_Medicamento)

Consume (Horario, ID_Mascota, ID_Medicamento, Dosis)

ID_Mascota → Mascota

ID_Medicamento → Medicamento

Camioneta(ID_Vehículo, MarcaVehículo)

Transporte(ID_Mascota, ID_Vehículo, Fecha de transporte, Horario, Estado, Tipo, Dirección, ID_Horario)

ID_Horario → Horario

ID_Mascota → Mascota

ID_Vehículo → Camioneta

Horario(ID_Horario, Desde, Hasta)

Confirma(ID_Mascota, ID_Vehículo, Fecha de transporte, Horario, CIEmpleado)

ID_Mascota, ID_Vehículo, Fecha de transporte, Horario → Transporte

CIEmpleado → Empleado

Habitación(ID_Habitación, ID_TipoHabitación)

ID_TipoHabitación → Tipo_Habitación

Tipo_Habitación(ID_TipoHabitación, Cantidad de mascotas, Precio día)

Descripción_Tipo(ID_TipoHabitación, Nombre_Tipo Enum, WebCam, DogTV)

Foto (ID Foto, Imagen, ID_TipoHabitación)

ID_TipoHabitación → Tipo_Habitación

Aloja(ID_Mascota, ID_Habitación)

ID_Mascota → Mascota

ID_Habitación → Habitación

Tiene (ID_Mascota, ID_Habitación, ID_Reserva)

ID_Mascota, ID_Habitación → Aloja

ID_Reserva → Reserva

Reserva(ID_Reserva, Estado, Cantidad de mascotas, Fecha y hr registrada, Fecha y hr desde, Fecha y hr hasta, Metodo de pago, WebCam, DogTV, CIEmpleado)

CIEmpleado → Empleado

3. Justificación de cambios en el MER

Hemos añadido una tabla (Foto), atributos en la tabla reserva (Servicios) y atributos en la tabla Tipo_Habitación (Servicios). Su razón es se debe a que se presentaron nuevos requerimientos funcionales por parte del docente a cargo de la asignatura programación que deben aparecer esencialmente en el sistema informático que estamos desarrollando.

4. Diccionario de Datos

Tabla Persona			
Atributo	Tipo	Tamaño	Descripción
CI	int	8	Refiere a la cédula de identidad de la persona
Nombre	varchar	40	Refiere al nombre propio de la persona.
Apellido	varchar	40	Refiere al apellido propio de la persona.
FechaNac	date	-	Refiere a la fecha de nacimiento de la persona
Dirección	varchar	40	Refiere a la dirección de residencia de la persona
Telefono	int	8	Refiere al teléfono que posee la persona

Tabla Cliente			
Atributo	Tipo	Tamaño	Descripción
CiCliente	int	8	Refiere a la cédula de identidad del cliente

Tabla Empleado			
Atributo	Tipo	Tamaño	Descripción
CIEmpleado	int	8	Refiere a la cédula de identidad del empleado
Usuario	varchar	30	Refiere al nombre de usuario del empleado en el programa
Contraseña	varchar	30	Refiere a la contraseña del empleado en el programa
RolPrograma	Enum (‘G’, ‘O’, ‘R’)	-	Refiere al rol que tiene el empleado en el programa
ImagenEmpleado	longblob	-	Refiere a la imagen o foto de perfil de cada empleado dentro de la empresa

Tabla Mascota			
Atributo	Tipo	Tamaño	Descripción
ID_Mascota	int	11	Refiere al número que identifica a la mascota
Nombre	varchar	30	Refiere al nombre de la mascota
FechaNac	date	-	Refiere a la fecha de nacimiento de la mascota
Vacunas	Enum ('T','F')	-	Refiere a si la mascota tiene todas sus mascotas o no
Comida	varchar	30	Refiere a la comida especial que come la mascota
Peso	varchar	11	Refiere al peso de la mascota
Raza	varchar	40	Refiere a la raza de la mascota
Sexo	Enum ('H','M')	-	Refiere al sexo de la mascota
CICliente	int	8	Refiere a la cédula del dueño de la mascota

Tabla Medicamento			
Atributo	Tipo	Tamaño	Descripción
ID_Medicamento	int	11	Refiere al número que identifica al medicamento
Nombre_Medicamento	varchar	30	Refiere al nombre del medicamento

Tabla Tiene			
Atributo	Tipo	Tamaño	Descripción
ID_Reserva	int	11	Refiere al número que identifica a la reserva
ID_Mascota	int	11	Refiere al número que identifica a la mascota
ID_Habitacion	int	11	Refiere al número de habitación en donde la mascota se encuentra alojada

Tabla Consume			
Atributo	Tipo	Tamaño	Descripción
Horario	time	-	Refiere al horario en que la mascota consume el medicamento
ID_Mascota	int	11	Refiere al número que identifica a la mascota que consumió el medicamento
ID_Medicamento	int	11	Refiere al número que identifica al medicamento consumido
Dosis	varchar	40	Refiere a la dosis en la cual se consume el medicamento

Tabla Camioneta			
Atributo	Tipo	Tamaño	Descripción
ID_Vehículo	int	11	Refiere al número que identifica el vehículo
MarcaVehículo	varchar	40	Refiere a la marca del vehículo

Tabla Transporte			
Atributo	Tipo	Tamaño	Descripción
ID_Mascota	int	11	Refiere al número que identifica la mascota transportada
ID_Vehículo	int	11	Refiere al número que identifica al vehículo que hace el transporte
Fecha de transporte	date	-	Refiere a la fecha en la cual se realiza el transporte
Horario	Enum (‘M’, ‘Ta’, ‘N’)	-	Refiere al horario en el cual se realiza el transporte
Estado	Enum (‘P’, ‘C’, ‘F’)	-	Refiere al estado del transporte
Tipo	Enum (‘LL’, ‘T’)	-	Refiere a qué tipo de transporte es el transporte
Dirección	varchar	30	Refiere a la dirección hacia la cual se realiza el transporte
ID_Horario	int	11	Refiere a el número que identifica al horario en el cual se realiza el transporte

Tabla Horario			
Atributo	Tipo	Tamaño	Descripción
ID_Horario	int	11	Refiere al número que identifica al horario
Desde	Time	-	Refiere a la hora en la cual empieza el horario
Hasta	Time	-	Refiere a la hora en la cual termina el horario

Tabla Confirma			
Atributo	Tipo	Tamaño	Descripción
ID_Mascota	int	11	Refiere al número que identifica la mascota transportada
ID_Vehículo	int	11	Refiere al número que identifica al vehículo que hace el transporte
Fecha de transporte	date	-	Refiere a la fecha en la cual se realiza el transporte
Horario	Enum (‘M’, ‘Ta’, ‘N’)	-	Refiere al horario en el cual se realiza el transporte
CIEmpleado	int	8	Refiere a la cédula del empleado que confirma el transporte

Tabla Habitación			
Atributo	Tipo	Tamaño	Descripción
ID_Habitación	int	11	Refiere al número que identifica a la habitación
ID_TipoHabitación	int	11	Refiere al número que identifica al tipo de habitación

Tabla Tipo_Habitación			
Atributo	Tipo	Tamaño	Descripción
ID_TipoHabitación	int	11	Refiere al número que identifica al tipo de habitación
Cantidad de mascotas	int	11	Refiere a la cantidad de mascotas en la habitación
Precio día	double	-	Refiere al precio de la habitación por día, basado en su tipo y cuantas mascotas tiene

Tabla Descripción_Tipo			
Atributo	Tipo	Tamaño	Descripción
ID_TipoHabitación	int	11	Refiere al número que identifica al tipo de habitación
Nombre_Tipo	Enum (‘S’,’J’,’M’,’E’)	-	Refiere al nombre de la habitación
WebCam	boolean	-	Refiere si el tipo habitación incluye un servicio WebCam para las mascotas
DogTV	boolean	-	Refiere si el tipo habitación incluye un servicio DogTV para las mascotas

Tabla Aloja			
Atributo	Tipo	Tamaño	Descripción
ID_Mascota	int	11	Refiere a el número que identifica a la mascota alojada
ID_Habitación	int	11	Refiere a el número que identifica a la habitación en la cuál se aloja la mascota

Tabla Foto			
Atributo	Tipo	Tamaño	Descripción
ID_Foto	int	2	Refiere a el número que identifica a una foto de una habitación
Imagen	longblob	-	Refiere a la imagen que le corresponde a la a un tipo de habitación
ID_TipoHabitación	int	11	Refiere el numero de Tipo de habitación que le corresponde a un 'ID Foto'

Tabla Reserva			
Atributo	Tipo	Tamaño	Descripción
ID_Reserva	int	11	Refiere al número que identifica a la reserva
Estado	Enum (‘P’, ‘C’, ‘CA’, ‘T’)	-	Refiere al estado de la reserva
Cantidad de mascotas	int	11	Refiere a la cantidad de mascotas ingresadas en la reserva
Fecha y hr registrada	DateTime	-	Refiere a la fecha y la hora en la cual se registra la reserva
Fecha y hr desde	DateTime	-	Refiere a la fecha y la hora en la cual empieza la reserva
Fecha y hr hasta	DateTime	-	Refiere a la fecha y la hora en la cual se termina la reserva
Metodo de pago	Enum (‘C’, ‘T’)	-	Refiere al método en el cual se paga la reserva
CIEmpleado	int	8	Refiere a la cédula del empleado que registra la reserva
ID_Mascota	int	11	Refiere al número que identifica a la mascota para la cual se hace la reserva
ID_Habitación	int	11	Refiere al número que identifica a la habitación que se reserva para la mascota
WebCam	boolean	-	Indica si el cliente desea incluir el servicio WebCam cuando realiza una reserva
DogTV	boolean	-	Indica si el cliente desea incluir el servicio DogTV cuando realiza una reserva

5. Sentencias DDL y DML

```
CREATE TABLE camioneta (  
    ID_Vehículo int NOT NULL AUTO_INCREMENT,  
    MarcaVehículo varchar(40) DEFAULT NULL,  
    PRIMARY KEY (ID_Vehículo)  
);
```

```
CREATE TABLE persona (  
    CI int NOT NULL,  
    Nombre varchar(40) DEFAULT NULL,  
    Apellido varchar(40) DEFAULT NULL,  
    FechaNac date DEFAULT NULL,  
    Direccion varchar(40) DEFAULT NULL,  
    Telefono int DEFAULT NULL,  
    PRIMARY KEY (CI)  
);
```

```
CREATE TABLE cliente (  
    CICliente int NOT NULL,  
    PRIMARY KEY (CICliente),  
    KEY CICliente (CICliente),  
    CONSTRAINT cliente_ibfk_1 FOREIGN KEY (CICliente) REFERENCES persona  
(CI)  
);
```

```
CREATE TABLE empleado (  
    CIEmpleado int NOT NULL,  
    Usuario varchar(30) DEFAULT NULL,  
    Contraseña varchar(30) DEFAULT NULL,  
    RolPrograma enum('G','O','R') DEFAULT NULL,  
    ImagenEmpleado longblob,  
    PRIMARY KEY (CIEmpleado),  
    KEY CIEmpleado (CIEmpleado),  
    CONSTRAINT empleado_ibfk_1 FOREIGN KEY (CIEmpleado) REFERENCES  
persona (CI)  
);
```

```
CREATE TABLE tipo_habitación (  
  ID_TipoHabitación int NOT NULL AUTO_INCREMENT,  
  CantidadDeMascotas int NOT NULL,  
  PrecioDia double DEFAULT NULL,  
  PRIMARY KEY (ID_TipoHabitación,CantidadDeMascotas)  
);
```

```
CREATE TABLE descripción_tipo (  
  ID_TipoHabitación int NOT NULL AUTO_INCREMENT,  
  Nombre_Tipo enum('S','J','M','E') DEFAULT NULL,  
  WebCam tinyint DEFAULT NULL,  
  DogTV tinyint DEFAULT NULL,  
  PRIMARY KEY (ID_TipoHabitación),  
  KEY ID_TipoHabitación (ID_TipoHabitación),  
  CONSTRAINT descripción_tipo_ibfk_1 FOREIGN KEY (ID_TipoHabitación)  
  REFERENCES tipo_habitación (ID_TipoHabitación)  
);
```

```
CREATE TABLE habitación (  
  ID_Habitación int NOT NULL AUTO_INCREMENT,  
  ID_TipoHabitación int DEFAULT NULL,  
  PRIMARY KEY (ID_Habitación),  
  KEY ID_TipoHabitación (ID_TipoHabitación),  
  CONSTRAINT habitación_ibfk_1 FOREIGN KEY (ID_TipoHabitación)  
  REFERENCES tipo_habitación (ID_TipoHabitación)  
);
```

```
CREATE TABLE foto (  
  idFoto int NOT NULL AUTO_INCREMENT,  
  IDTipoHabitacion int DEFAULT NULL,  
  Imagen longblob,  
  PRIMARY KEY (idFoto),  
  KEY tipo_idx (IDTipoHabitacion),  
  CONSTRAINT tipo FOREIGN KEY (IDTipoHabitacion) REFERENCES  
  tipo_habitación (ID_TipoHabitación)  
);
```

```
CREATE TABLE horario (  
  ID_Horario int NOT NULL AUTO_INCREMENT,  
  Desde time DEFAULT NULL,  
  Hasta time DEFAULT NULL,  
  PRIMARY KEY (ID_Horario)  
);
```

```
CREATE TABLE reserva (  
  ID_Reserva int NOT NULL AUTO_INCREMENT,  
  Estado enum('P','C','Ca','T') DEFAULT NULL,  
  CantidadDeMascotas int DEFAULT NULL,  
  FechaYHoraRegistrada timestamp DEFAULT CURRENT_TIMESTAMP,  
  FechaYHoraDesde datetime DEFAULT NULL,  
  FechaYHoraHasta datetime DEFAULT NULL,  
  MetodoDePago enum('C','T') DEFAULT NULL,  
  WebCam tinyint DEFAULT NULL,  
  DogTV tinyint DEFAULT NULL,  
  CIEmpleado int DEFAULT NULL,  
  PRIMARY KEY (ID_Reserva),  
  KEY reserva_ibfk_3_idx (CIEmpleado),  
  CONSTRAINT reserva_ibfk_3 FOREIGN KEY (CIEmpleado) REFERENCES  
  empleado (CIEmpleado)  
);
```

```
CREATE TABLE mascota (  
  ID_Mascota int NOT NULL AUTO_INCREMENT,  
  Nombre varchar(30) DEFAULT NULL,  
  FechaNac date DEFAULT NULL,  
  Vacunas enum('T','F') DEFAULT NULL,  
  Comida varchar(30) DEFAULT NULL,  
  Peso int DEFAULT NULL,  
  Raza varchar(40) DEFAULT NULL,  
  Sexo enum('M','H') DEFAULT NULL,  
  CICliente int DEFAULT NULL,  
  PRIMARY KEY (ID_Mascota),  
  KEY ID_Mascota (ID_Mascota),  
  KEY CICliente (CICliente),  
  CONSTRAINT mascota_ibfk_1 FOREIGN KEY (CICliente) REFERENCES cliente  
  (CICliente)  
);
```

```
CREATE TABLE transporte (  
  ID_Mascota int NOT NULL,  
  ID_Vehículo int NOT NULL,  
  FechaDEtransporte date NOT NULL,  
  Horario enum('M','Ta','N') NOT NULL,  
  Estado enum('P','C','F') DEFAULT NULL,  
  Tipo enum('LL','T') DEFAULT NULL,  
  Dirección varchar(30) DEFAULT NULL,  
  ID_Horario int DEFAULT NULL,  
  PRIMARY KEY (ID_Mascota,ID_Vehículo,FechaDEtransporte,Horario),  
  KEY ID_Mascota (ID_Mascota),  
  KEY ID_Vehículo (ID_Vehículo),  
  KEY ID_Horario (ID_Horario),  
  KEY FechaDEtransporte (FechaDEtransporte),  
  KEY Horario (Horario),  
  CONSTRAINT transporte_ibfk_1 FOREIGN KEY (ID_Mascota) REFERENCES  
mascota (ID_Mascota),  
  CONSTRAINT transporte_ibfk_2 FOREIGN KEY (ID_Vehículo) REFERENCES  
camioneta (ID_Vehículo),  
  CONSTRAINT transporte_ibfk_3 FOREIGN KEY (ID_Horario) REFERENCES  
horario (ID_Horario)  
);
```

```
CREATE TABLE confirma (  
  ID_Mascota int NOT NULL,  
  ID_Vehículo int NOT NULL,  
  FechaDEtransporte date NOT NULL,  
  Horario enum('M','Ta','N') NOT NULL,  
  CIEmpleado int DEFAULT NULL,  
  PRIMARY KEY (ID_Mascota,ID_Vehículo,FechaDEtransporte,Horario),  
  KEY ID_Mascota (ID_Mascota),  
  KEY ID_Vehículo (ID_Vehículo),  
  KEY FechaDEtransporte (FechaDEtransporte),  
  KEY Horario (Horario),  
  KEY confirma_ibfk_5_idx (CIEmpleado),  
  CONSTRAINT confirma_ibfk_1 FOREIGN KEY (ID_Mascota) REFERENCES  
transporte (ID_Mascota),  
  CONSTRAINT confirma_ibfk_2 FOREIGN KEY (`ID_Vehículo`) REFERENCES  
transporte (`ID_Vehículo`),  
  CONSTRAINT confirma_ibfk_3 FOREIGN KEY (FechaDEtransporte)  
REFERENCES transporte (FechaDEtransporte),  
  CONSTRAINT confirma_ibfk_4 FOREIGN KEY (Horario) REFERENCES  
transporte (Horario),
```

```
CONSTRAINT confirma_ibfk_5 FOREIGN KEY (CIEmpleado) REFERENCES  
empleado (CIEmpleado)  
);
```

```
CREATE TABLE aloja (  
  ID_Mascota int NOT NULL,  
  ID_Habitación int NOT NULL,  
  PRIMARY KEY (ID_Mascota, ID_Habitación),  
  KEY ID_Mascota (ID_Mascota),  
  KEY ID_Habitación (ID_Habitación),  
  CONSTRAINT aloja_ibfk_1 FOREIGN KEY (ID_Habitación) REFERENCES  
habitación (ID_Habitación),  
  CONSTRAINT aloja_ibfk_2 FOREIGN KEY (ID_Mascota) REFERENCES  
mascota (ID_Mascota)  
);
```

```
CREATE TABLE tiene (  
  IDReserva int NOT NULL,  
  IDMascota int NOT NULL,  
  IDHabitacion int NOT NULL,  
  PRIMARY KEY (IDReserva, IDMascota, IDHabitacion),  
  KEY IDReserva (IDReserva),  
  KEY IDMascota (IDMascota),  
  KEY mascota_reserva_ibfk_3_idx (IDHabitacion),  
  CONSTRAINT tiene_ibfk_1 FOREIGN KEY (IDReserva) REFERENCES reserva  
(ID_Reserva),  
  CONSTRAINT tiene_ibfk_2 FOREIGN KEY (IDMascota) REFERENCES aloja  
(ID_Mascota),  
  CONSTRAINT tiene_ibfk_3 FOREIGN KEY (IDHabitacion) REFERENCES aloja  
(ID_Habitación)  
);
```

```
CREATE TABLE medicamento (  
  ID_Medicamento int NOT NULL AUTO_INCREMENT,  
  Nombre_Medicamento varchar(30) DEFAULT NULL,  
  PRIMARY KEY (ID_Medicamento)  
);
```

```
CREATE TABLE consume (
  Horario time NOT NULL,
  ID_Mascota int NOT NULL,
  ID_Medicamento int NOT NULL,
  Dosis varchar(40) DEFAULT NULL,
  PRIMARY KEY (Horario,ID_Mascota,ID_Medicamento),
  KEY Horario (Horario),
  KEY ID_Mascota (ID_Mascota),
  KEY ID_Medicamento (ID_Medicamento),
  CONSTRAINT consume_ibfk_1 FOREIGN KEY (ID_Mascota) REFERENCES
mascota (ID_Mascota),
  CONSTRAINT consume_ibfk_2 FOREIGN KEY (ID_Medicamento)
REFERENCES medicamento (ID_Medicamento)
);
```

```
INSERT INTO `camioneta` VALUES (1,'Chevrolet'),(2,'Chevrolet');
```

```
INSERT INTO `persona` VALUES (16599980,'Blair','Foley','1999-11-08','Ap
#824-8144 Nunc Ave','61922166),(19698885,'Ciaran','Kemp','1999-05-25','P.O.
Box 749, 2295 Mollis. Street','32487985),(25798523,'Keane','Peck','1999-05-
17','8841 Gravida St.','66004721),(29915858,'Gisela','Buck','1999-12-01','142-1631
Purus Avenue','67139035),(34452777,'Tyrone','Hurley','1999-11-26','4867 Sem,
Avenue','75306523),(51499855,'Manuel','Muñis','2000-01-
01','Ninguna','95642456),(52059150,'Fabrizio','Bravo','2000-06-01','Agustin
Muños','63874594),(54149985,'Manuel','Muñis','2000-05-03','No
tiene','46568094),(55140213,'Conan','Mcconnell','1999-02-03','Ap #334-8563
Scelerisque, Road','64879925),(56124981,'Agustin','Robaina','2000-02-03','No
tiene','2602644),(60471511,'Aurelia','Hayden','1999-11-22','P.O. Box 451, 3404
Donec St.','45773118),(61262303,'Upton','Langley','1999-08-06','502 Vulputate
Rd.','34657969),(73316678,'Mari','Greene','1999-12-22','256-9763 Pharetra.
St.','54619909);
```

```
INSERT INTO `cliente` VALUES
(16599980),(19698885),(25798523),(29915858),(34452777),(51499855),(5514021
3),(60471511),(61262303),(73316678);
```

```
INSERT INTO `empleado` VALUES
(16599980,'noraherandez','12345678','R',NULL),(52059150,'fabribravo','123456
7890','G',NULL),(54149985,'manuelmuniz','1234567890','G',NULL),(55140213,'w
altertano','123456789','O',NULL),(61262303,'cristianperez','123456789','O',NULL)
,(73316678,'juanreyes','12345678','R',NULL);
```



```
INSERT INTO `tipo_habitación` VALUES
(1,1,34),(1,2,51),(1,3,68),(2,1,31),(2,2,46.5),(2,3,62),(3,1,31),(3,2,46.5),(3,3,62),(4,1,
28),(4,2,40),(4,3,56);
```

```
INSERT INTO `descripción_tipo` VALUES
(1,'S',1,1),(2,'J',0,0),(3,'M',0,0),(4,'E',0,0);
```

```
INSERT INTO `habitación` VALUES
(1,1),(2,1),(3,1),(4,1),(5,1),(6,1),(7,1),(8,1),(9,1),(10,1),(11,1),(12,1),(13,1),(14,1),(15,1),
(16,1),(17,1),(18,1),(19,2),(20,2),(21,2),(22,2),(23,2),(24,2),(25,2),(26,2),(27,2),(28,
2),(29,2),(30,2),(31,2),(32,2),(33,2),(34,2),(35,2),(36,2),(37,3),(38,3),(39,3),(40,3),
(41,3),(42,3),(43,3),(44,3),(45,3),(46,3),(47,3),(48,3),(49,3),(50,3),(51,3),(52,3),(5
3,3),(54,3),(55,3),(56,3),(57,3),(58,3),(59,3),(60,4),(61,4),(62,4),(63,4),(64,4),(65,
4),(66,4),(67,4),(68,4),(69,4),(70,4),(71,4),(72,4),(73,4),(74,4),(75,4),(76,4),(77,4),
(78,4),(79,4),(80,4),(81,4),(82,4),(83,4);
```

```
INSERT INTO `horario` VALUES
(1,'11:00:00','13:00:00'),(2,'14:00:00','16:30:00'),(3,'17:00:00','19:00:00');
```

```
INSERT INTO `reserva` VALUES (1,'Ca',2,'2020-11-08 16:32:23','2020-11-09
16:25:59','2020-11-15 16:25:59','T',1,1,52059150),(2,'Ca',1,'2020-11-08
16:35:01','2020-11-15 16:32:56','2020-11-29
16:32:56','C',0,0,52059150),(3,'C',1,'2020-11-08 18:01:29','2020-11-12
18:00:43','2020-11-28 18:00:43','T',0,0,52059150),(4,'P',2,'2020-11-09
17:01:55','2020-11-20 16:55:16','2020-11-29
16:55:16','T',1,1,52059150),(5,'P',1,'2020-11-09 17:05:27','2020-11-21
17:04:07','2020-11-30 17:04:07','C',0,0,52059150),(6,'P',2,'2020-11-09
17:17:30','2020-11-13 17:15:27','2020-11-30
17:15:27','T',0,0,52059150),(7,'P',3,'2020-11-09 17:23:13','2020-11-13
17:18:45','2020-11-29 17:18:45','T',0,0,52059150),(8,'P',2,'2020-11-09
17:24:39','2020-11-09 17:23:49','2020-11-26
17:23:48','C',0,0,52059150),(9,'P',1,'2020-11-09 17:26:16','2020-11-26
17:24:56','2020-11-29 17:24:56','T',0,1,52059150);
```

```
INSERT INTO `mascota` VALUES (1,'Firulais','2000-06-
21','T','Premium',40,'Ovejero Aleman','M',73316678),(2,'Pepe','2004-05-
11','T','Premium',40,'Ovejero Aleman','M',73316678),(3,'Juani','2018-12-
31','T','Premium',45,'Caniche','H',25798523),(4,'Chuchi','2004-06-
09','T','Ninguna',49,'Podul','M',60471511),(5,'Luigi','2019-01-
01','T','Carne',40,'Ovejero Aleman','M',16599980),(6,'Mario','2015-07-
10','T','Premium',45,'Caniche','M',16599980),(7,'Mario','2000-06-
22','T','Arroz',48,'Podul','M',19698885),(8,'Rya','2015-06-
11','T','Carne',49,'Podul','H',29915858),(9,'Ouchi','2014-06-
10','T','Premium',34,'Doberman','M',34452777),(10,'Richi','2016-07-
```

15,'T','Premium',34,'Doberman','M',34452777),(11,'Riky','2013-05-16','T','Premium',55,'Doberman','M',34452777),(12,'Cuchi','2014-06-12','T','Premium',37,'Ovejero Aleman','M',55140213),(13,'Lori','2015-11-12','F','Arroz',49,'Podul','H',61262303);

INSERT INTO `transporte` VALUES (1,1,'2020-11-09','M','F','T','T',1),(2,1,'2020-11-09','M','F','T','T',1),(3,1,'2020-11-15','M','F','T','T',1),(5,1,'2020-11-20','M','P','T','T',1),(6,1,'2020-11-20','M','P','T','T',1),(8,2,'2020-11-30','Ta','P','LL','LL',2),(10,2,'2020-11-13','M','P','T','T',1),(11,2,'2020-11-13','M','P','T','T',1),(13,1,'2020-11-16','N','P','LL','LL',3);

INSERT INTO `aloja` VALUES (1,1),(2,1),(3,37),(4,38),(5,1),(6,1),(7,37),(8,39),(9,41),(10,41),(11,41),(12,60),(13,3);

INSERT INTO `tiene` VALUES (1,1,1),(1,2,1),(2,3,37),(3,4,38),(4,5,1),(4,6,1),(5,7,37),(6,8,39),(7,9,41),(7,10,41),(7,11,41),(8,12,60),(9,13,3);

INSERT INTO `confirma` VALUES (5,1,'2020-11-20','M',52059150),(6,1,'2020-11-20','M',52059150);

INSERT INTO `medicamento` VALUES (1,'Antiparasitarios'),(2,'ADAPTIL'),(5,'Calcio');

INSERT INTO `consume` VALUES ('12:00:00',1,1,'1'),('12:00:00',4,1,'1'),('12:00:00',5,5,'1'),('12:00:00',12,1,'2'),('15:00:00',5,5,'1');

6. Usuarios

Se contará con un total de 4 usuarios:

Administrador (DBA)

Nombre: dba

Contraseña: administrador123

Gerente

Nombre: gerente

Contraseña: gerente123

Oficinista

Nombre: oficinista

Contraseña: oficinista123

Recepcionista de Garaje

Nombre: recepcionista

Contraseña: recepcionista123

6.1. Creación de los usuarios

```
create user 'dba'@'localhost' identified by 'administrador123';  
create user 'gerente'@'localhost' identified by 'gerente123';  
create user 'oficinista'@'localhost' identified by 'oficinista123';  
create user 'recepcionista'@'localhost' identified by 'recepcionista123';
```

7. Permisos

7.1. Permisos de los diferentes roles

Administrador (DBA):

- Puede efectuar todas las acciones posibles en la base de datos y crear, eliminar y modificar usuarios con diversos roles (Gerente, Oficinista, Recepcionista de Garaje).

Gerente:

- Consulta, modifica, agrega y elimina clientes.
- Consulta, modifica, agrega y elimina mascotas.
- Consulta, modifica, agrega y elimina medicamentos.
- Consulta, modifica, agrega y elimina los medicamentos que le corresponde a cada mascota (horario, dosis).
- Consulta, modifica, agrega los transportes en donde se transportan las mascotas.
- Consulta todas las camionetas que se encuentran disponibles.
- Consulta todos los horarios en donde se efectúa un transporte.
- Consulta, modifica y agrega reservas.
- Consulta todas las habitaciones.
- Consulta y modifica en donde se alojan las mascotas.
- Consulta y modifica el precio por día correspondiente a cada tipo de habitación.
- Consulta la descripción correspondiente a cada tipo de habitación (Imagen, Nombre)
- Consulta las fotos correspondientes a cada tipo de habitación.

Oficinista:

- Consulta, modifica, agrega y elimina clientes.
- Consulta, modifica, agrega y elimina mascotas.
- Consulta, modifica, agrega y elimina medicamentos.
- Consulta, modifica, agrega y elimina los medicamentos que le corresponde a cada mascota (horario, dosis).

- Consulta, modifica, agrega los transportes en donde se transportan las mascotas.
- Consulta todas las camionetas que se encuentran disponibles.
- Consulta todos los horarios en donde se efectúa un transporte.
- Consulta, modifica y agrega reservas.
- Consulta todas las habitaciones.
- Consulta y modifica en donde se alojan las mascotas.
- Consulta el precio por día correspondiente a cada tipo de habitación.
- Consulta la descripción correspondiente a cada tipo de habitación (Imagen, Nombre)
- Consulta las fotos correspondientes a cada tipo de habitación.

Recepcionista de Garaje:

- Consulta las mascotas que se encuentran en un transporte.
- Consulta las camionetas que realizan un transporte.
- Consulta los transportes en donde se transportan las mascotas.
- Modifica el estado del transporte (Pendiente, Confirmado y Finalizado)

7.2. Matriz de control de acceso

OBJETOS					
---------	--	--	--	--	--

SUJETOS	Cliente Tabla	Empleado Tabla	Reserva Tabla	Tiene Tabla	Foto Tabla
Gerente	select, insert, update, delete		select, insert, update	select, insert, update	select
Oficinista	select, insert, update, delete		select, insert, update	select, insert, update	select
Recepcionista					

SUJETOS	PrecioDia Columna	Aloja Tabla	Mascota Tabla	Tipo_Habitación Tabla	Descripcion_Tipo Tabla
Gerente	select, update	select, update	select, insert, update, delete	select, update	select
Oficinista	select	select, update	select, insert, update, delete	select, update	select
Recepcionista			select		

SUJETOS	Camioneta Tabla	Horario Tabla	Consume Tabla	Medicamento Tabla	Transporte Tabla
Gerente	select	select	select, insert, update, delete	select, insert, update, delete	select, insert, update
Oficinista	select	select	select, insert, update, delete	select, insert, update, delete	select, insert, update
Recepcionista	select				select

Sujetos	Confirma Tabla
Gerente	Select, insert, update
Oficinista	Select, Insert, update
Recepcionista	Select, update

7.3. Sentencias utilizadas para asignar permisos

Administrador (dba):

Grant all on *.* to dba@localhost
with grant option;

Gerente:

Grant insert, select, update, delete on HotelGuau.Cliente to gerente@localhost;
Grant insert, select, update, delete on HotelGuau.mascota to gerente@localhost;
Grant insert, select, update, delete on HotelGuau.consume to gerente@localhost;
Grant insert, select, update, delete on HotelGuau.medicamento to
gerente@localhost;
Grant insert, select, update on HotelGuau.transporte to gerente@localhost;
Grant select on HotelGuau.camioneta to gerente@localhost;
Grant select on HotelGuau.horario to gerente@localhost;
Grant insert, select, update on HotelGuau.reserva to gerente@localhost;
Grant insert, select, update on HotelGuau.tiene to gerente@localhost;
Grant select on HotelGuau.habitación to gerente@localhost;
Grant select, update on HotelGuau.aloja to gerente@localhost;
Grant select, update(preciodia) on HotelGuau.tipo_habitación to
gerente@localhost;
Grant select on HotelGuau.descripcion_tipo to gerente@localhost;
Grant select on HotelGuau.foto to gerente@localhost;
Grant insert, select, update on HotelGuau.Confirma to gerente@localhost;

Oficinista:

Grant insert, select, update, delete on HotelGuau.Cliente to oficinista@localhost;
Grant insert, select, update, delete on HotelGuau.mascota to oficinista@localhost;
Grant insert, select, update, delete on HotelGuau.consume to oficinista@localhost;
Grant insert, select, update, delete on HotelGuau.medicamento to
oficinista@localhost;
Grant insert, select, update on HotelGuau.transporte to oficinista@localhost;
Grant select on HotelGuau.camioneta to oficinista@localhost;
Grant select on HotelGuau.horario to oficinista@localhost;
Grant insert, select, update on HotelGuau.reserva to oficinista@localhost;
Grant insert, select, update on HotelGuau.tiene to oficinista@localhost;
Grant select on HotelGuau.habitación to oficinista@localhost;
Grant select, update on HotelGuau.aloja to oficinista@localhost;
Grant select(preciodia) on HotelGuau.tipo_habitación to oficinista@localhost;
Grant select on HotelGuau.descripcion_tipo to oficinista@localhost;
Grant select on HotelGuau.foto to oficinista@localhost;
Grant insert, select, update on HotelGuau.Confirma to oficinista@localhost;

Recepcionista de Garaje:

Grant select on HotelGuau.mascota to recepcionista@localhost;
Grant select on HotelGuau.camioneta to recepcionista@localhost;
Grant select on HotelGuau.transporte to recepcionista@localhost;
Grant select, update on HotelGuau.Confirma to recepcionista@localhost;

8. Consultas S.Q.L. solicitadas por el Docente

1) SELECT now() 'Fecha Hoy', ID_Mascota 'Nro Mascota', M.Nombre, Raza, M.FechaNac 'Nacimiento', Sexo, Peso, P.Nombre 'Cliente' FROM mascota M join persona P on M.CICliente = P.CI;

2) SELECT ID_Reserva, FechaYHoraRegistrada, NOW() 'Hora Actual', CICliente, CIEmpleado from reserva R join tiene T ON R.ID_Reserva = T.IDReserva join Mascota M on T.IDMascota = M.ID_Mascota where Estado='P' and DATEDIFF(NOW(), FechaYHoraRegistrada) >= 1;

3) SELECT DISTINCT ID_Reserva, CIEmpleado, FechaYHoraRegistrada, FechaYHoraDesde, FechaYHoraHasta, (datediff(R.FechaYHoraHasta, R.FechaYHoraDesde) * PrecioDia) 'Total' from reserva R join tiene T on R.ID_Reserva = T.IDReserva join mascota M on T.IDMascota = M.ID_Mascota JOIN habitación H on H.ID_Habitación = T.IDHabitacion join tipo_habitación TI on TI.ID_TipoHabitación = H.ID_TipoHabitación where R.CantidadDeMascotas = TI.CantidadDeMascotas and month(FechaYHoraDesde) = month(date_add(NOW(), Interval 1 Month));

4) SELECT M.ID_Mascota, M.nombre 'Mascota', peso, A.ID_Habitación 'Habitación', P.nombre 'Cliente', P.telefono 'Telefono' from mascota M join aloja A on A.ID_Mascota = M.ID_Mascota join persona P on P.CI = M.CICliente join habitación H on H.ID_Habitación = A.ID_Habitación where ID_TipoHabitación=3 and peso > 12;

5) Select Nombre_Tipo 'Nombre del Tipo de Habitación', count(h.ID_TipoHabitación) 'Cantidad de Mascotas' from habitación H join aloja A on H.ID_Habitación = A.ID_Habitación join descripción_tipo D on D.ID_TipoHabitación = H.ID_TipoHabitación group by H.ID_TipoHabitación;

6) Select Nombre_Tipo, CantidadDeMascotas, PrecioDia from tipo_habitación T join descripción_tipo D ON T.ID_TipoHabitación = D.ID_TipoHabitación ORDER BY Nombre_Tipo;

7) SELECT DISTINCT CICliente 'Nro Documento', datediff(R.FechaYHoraHasta, R.FechaYHoraDesde) 'Dias de Alojamiento', IDHabitacion 'Habitación', PrecioDia 'Precio x dia', (datediff(R.FechaYHoraHasta, R.FechaYHoraDesde) * PrecioDia) 'Total' from reserva R join tiene T on R.ID_Reserva = T.IDReserva join mascota M on T.IDMascota = M.ID_Mascota JOIN habitación H on H.ID_Habitación = T.IDHabitacion join tipo_habitación TI on TI.ID_TipoHabitación = H.ID_TipoHabitación where R.CantidadDeMascotas = TI.CantidadDeMascotas;

8) SELECT P.Nombre 'Nombre Cliente', M.nombre 'Nombre Mascota' from aloja A join mascota M on A.ID_Mascota = M.ID_Mascota join persona P ON M.CICliente = P.CI;

9) SELECT Horario, Dosis, Nombre_Medicamento, C.ID_Mascota, M.nombre from consume C join mascota M on C.ID_Mascota = M.ID_Mascota join medicamento ME on ME.ID_Medicamento = C.ID_Medicamento order by Horario and C.ID_Mascota;

10) SELECT ID_Mascota, nombre, CICliente from mascota WHERE ID_Mascota NOT IN (SELECT ID_Mascota from consume);

11) SELECT T.ID_Vehículo, C.MarcaVehículo, T.FechaDEtransporte, T.Horario from transporte T join camioneta C WHERE T.ID_Vehículo = C.ID_Vehículo and date (T.FechaDEtransporte) = date (now()) and T.ID_Vehículo not in (select (ID_Vehículo) from transporte where (date(FechaDEtransporte) = date(now()) and Horario= 'M') or (date(FechaDEtransporte) = date(now()) and Horario= 'Ta') or (date(FechaDEtransporte) = date(now()) and Horario= 'N')) group by ID_Vehículo having count(ID_Vehículo) > 2);

(Debido a que nuestra BD de datos no cuenta con un Sistema que agende cada transporte en un horario en específico la manera que encontramos para resolver la consulta es mostrar las camionetas disponibles en el día de hoy.)

12) select T.ID_Mascota, nombre, CICliente, FechaDEtransporte from transporte T join mascota M on T.ID_Mascota = M.ID_Mascota where day(FechaDEtransporte)=day(NOW());

13) select ID_Mascota, nombre, CICliente from mascota where ID_Mascota NOT IN (SELECT ID_Mascota from transporte);

14) SELECT usuario, nombre, FechaDEtransporte from confirma C join Mascota M on C.ID_Mascota = M.ID_Mascota join empleado E on E.CIEmpleado = C.CIEmpleado where nombre='Firulais';

15) SELECT CICliente 'Cliente', COUNT(CICliente) 'Mascotas' from mascota group by CICliente having COUNT(CICliente) > 1;

9. Respaldo de Datos

Consideramos de suma importancia realizar siempre el respaldo de una base de datos.

En este caso la base de datos del hotel canino GUAU es el pilar fundamental para poder brindar sus servicios de manera rápida y eficaz, la importancia de los datos almacenados allí radica en lo esencial que resulta mantener el respaldo actualizado de los mismos, así como también de toda la base de datos en general.

Hemos tenido en cuenta que la magnitud de los datos es amplia ya que, día a día se gestionaran varias operaciones que modificaran los existentes y agregaran nuevos.

Teniendo en cuenta todas estas cualidades de este caso en particular hemos decidido que lo ideal es realizar 7 copias de seguridad semanales, de tal manera que al final de cada día se realice un respaldo de naturaleza diferencial, el cual solo almacenará las modificaciones realizadas a la Base de Datos en ese día, excepto por el día final de la semana, en el cual se realizará un respaldo total, el cual guardará todas las modificaciones hechas en la semana.

El respaldo estará automatizado a partir de un script al que solo tendrá acceso el superusuario (root), así garantizamos la seguridad de la base de datos. El mismo se ejecutará en segundo plano todos los días a las 2 am.



Consejo de Educación
Técnico - Profesional
(Universidad del Trabajo del Uruguay)



Instituto Tecnológico Superior
UTU

HOJA TESTIGO