

NORMALIZACIÓN HOTEL P1

Servicios:

S(idServicio, nombreServicio, descripción, horario, capacidad, costo, menu)

DF:

```
{
idServicio-> nombreServicio, descripción, horario, capacidad, costo, menu
}
```

- La relación servicios está en forma normal de Boyce Codd porque todos sus atributos son atómicos, no existen dependencias parciales desde los atributos primos, no existen dependencias transitivas entre atributos no primos y no hay dependencias no triviales entre atributos primos. Todo esto se puede ver más claramente porque hay un único atributo que compone la llave primaria y todos los atributos no primos dependen de este.

Planes:

P(nombrePlan, descuento, periodoVigenciaInicial, periodoVigenciaFinal)

DF:

```
{
nombrePlan-> descuento, periodoVigenciaInicial, periodoVigenciaFinal
}
```

- La relación planes está en forma normal de Boyce Codd porque todos sus atributos son atómicos, no existen dependencias parciales desde los atributos primos, no existen dependencias transitivas entre atributos no primos y no hay dependencias no triviales entre atributos primos. Todo esto se puede ver más claramente porque hay un único atributo que compone la llave primaria y todos los atributos no primos dependen de este.

PlanesServicios:

PS(idServicio, nombrePlan)

DF:

```
{
}
```

- La relación PlanesServicios no tiene dependencias funcionales porque esta solo cuenta con sus llaves primarias. Por ende, como no hay dependencias funcionales ni

transitivas para considerar, la relación automáticamente cumple con los requisitos para estar en forma normal de Boyce-Codd.

ReservasServicios:

RS(idServicio, cedulaUsuario, fechaReserva)

DF:

```
{  
idServicio, cedulaUsuario-> fechaReserva  
}
```

- La relación ReservasServicios está en la forma normal de Boyce Codd porque: i) todos sus atributos son atómicos, ii) todas las dependencias son totales desde los atributos primos, iii) no existen dependencias transitivas y iv) no hay dependencias no triviales entre atributos primos.

Usuarios:

U(cedula, tipoDocumento, nombreUsuario, correo, tipoUsuario)

DF:

```
{  
Cedula->tipoDocumento, nombreUsuario  
}
```

- La relación usuarios está en forma normal de Boyce Codd porque todos sus atributos son atómicos, no existen dependencias parciales desde los atributos primos, no existen dependencias transitivas entre atributos no primos y no hay dependencias no triviales entre atributos primos. Aunque no todos los atributos no primos dependen de la llave primaria (o los atributos primos) esto no viola los principios previamente mencionados ya que estos criterios se evalúan sobre dependencias existentes.

TiposUsuarios:

TU(nombre)

DF:

```
{  
}
```

- La relación TiposUsuarios no tiene dependencias funcionales porque esta solo cuenta con su llave primaria. Por ende, como no hay dependencias funcionales ni

transitivas para considerar, la relación automáticamente cumple con los requisitos para estar en forma normal de Boyce-Codd.

Consumos:

C(idConsumo, descripción, fecha, costo, idServicio, idReserva)

DF:

{
idConsumo -> descripción, fecha, costo, idServicio, idReserva }

- La relación consumos está en forma normal de Boyce Codd porque todos sus atributos son atómicos, no existen dependencias parciales desde los atributos primos, no existen dependencias transitivas entre atributos no primos y no hay dependencias no triviales entre atributos primos. Todo esto se puede ver más claramente porque hay un único atributo que compone la llave primaria y todos los atributos no primos dependen de este.

Reservas:

R(idReserva, fechaEntrada, fechaSalida, cantidadPersonas, usuariosCedula, nombrePlan, numeroHabitacion, fechaCheckIn, fechaCheckOut)

DF:

{
idReserva -> fechaEntrada, fechaSalida, cantidadPersonas, usuariosCedula, nombrePlan, numeroHabitacion }

- La relación reservas está en forma normal de Boyce Codd porque todos sus atributos son atómicos, no existen dependencias parciales desde los atributos primos, no existen dependencias transitivas entre atributos no primos y no hay dependencias no triviales entre atributos primos. Todo esto se puede ver más claramente porque hay un único atributo que compone la llave primaria y todos los atributos no primos dependen de este excepto los últimos. Sin embargo, como estos no dependen de ningún otro atributo no “dañan” la normalización porque no violan los principios previamente expresados.

Habitaciones:

H(numero, tipo)

DF:

{numero -> tipo }

- La relación reservas está en forma normal de Boyce Codd porque todos sus atributos son atómicos, no existen dependencias parciales desde los atributos primos, no existen dependencias transitivas entre atributos no primos y no hay dependencias no triviales entre atributos primos. Todo esto se puede ver más claramente porque hay un único atributo que compone la llave primaria y un único atributo no primo que depende de este atributo primo.

TipoH:

TH(nombreTipo, dotación, capacidad, costoNoche)

DF:

{ nombreTipo -> dotación, capacidad, costoNoche }

- En este caso, la relación TipoH es la única que se podría argumentar que no está ni siquiera en primera forma normal porque el atributo dotación considerarse un atributo compuesto. (Nota: sin embargo, como en este punto del proyecto no creemos que esto nos impida cumplir ningún requerimiento funcional y, al contrario, va a simplificar el modelo.