



# PROYECTO TELELLAMADAS

## Especificaciones



UNAM Faculta de Ciencias  
F.B.D (Fundamentos de Bases  
de Datos)



## EQUIPO DE TRABAJO

- Barón Herrera Victoria Elena 315650383 victoriaebaronh@ciencias.unam.mx
- Brito Juárez Britny 421074668 britny\_brito@ciencias.unam.mx
- Castrejón Martínez Isaías 317203356 isaiascm@ciencias.unam.mx
- Preciado Márquez David Salvador 421091670 dpreciado3@ciencias.unam.mx
- Vargas Bravo Paola 318074755 Paola\_VB@ciencias.unam.mx



## Documentos Entregados

1. Especificaciones : En este documento se da la justificación del diseño del proyecto, además de tener las **dependencias funcionales**
2. Diccionario de Datos : En este documento contenga definiciones de campos y las abreviaturas o formatos de la base de datos
3. Consultas : En este documento se contienen las 15 consultas solicitadas

## Contenido

En este documento se encontraran especificaciones que considero el equipo de trabajo para poder dar como resultado la base de datos presentada y deseada.

Cabe aclarar que únicamente son las especificaciones más importantes y no realmente las justificación de todo lo presentado.

LA MAFIA DEL PODER ~

## Herramienta de Generación de Datos

Mockaroo

# Justificación de Entidades, Relaciones y Atributos

## Entidades

### 1. Empleado

Notemos que existe una herencia de Empleado con Entrenador y Agente, esto dado que ambos son empleados en el centro de contacto “TeleLlamadas” es por ello nuestra **SuperEntidad** es Empleado , las SubEntidades 1 y 2 son Entrenador y Agent e.

Entonces podemos ver que si sacamos un termino en común de los atributos de Entrenador y Agente , se tienen 8 iguales y solo difieren en 2, es por ello que se decidió usar **Herencia** en esta parte.

### 2. Asistencia Curso

Se opto por una entidad dado que se debe de llevar un registro de las asistencias de los agentes a los cursos

### 3. Acceso

(EntradaEntrenador, SalidaEntrenador, EntradaAgente y SalidaAgente)

Dado que se requiere tener las fechas (que incluyan también la hora y minuto) en las que los entrenadores y agentes entran y salen de su piso, y al ser probable que las veces que un entrenador o agente ingrese a su piso sean varias, creamos las entidades EntradaEntrenador, SalidaEntrenador, EntradaAgente y SalidaAgente. No guardamos las fechas de entrada y salida en una sola tabla pues eso implicaría tener que guardar en un principio la fecha de salida como nula para después editarla.

### 4. Horario

Se decidió crear una entidad puesto que el número de horas de cada semana puede variar y nos interesa llevar un registro de las horas que tienen cada semana cada entrenador.

### 5. Edificio

Se decidió crear una entidad fuerte que represente cada uno de los edificios en donde se imparten los cursos.

### 6. Piso

Se decidió crear una entidad que representa los pisos de cada uno de los edificios donde se imparten los cursos. Por esta razón, esta entidad es débil y se identifica con el edificio al que pertenece.

### 7. Mouse, Headset, Teclado (Periféricos)

Se decidió crear una entidad para los perifericos de cada estación, ya que se necesita tener cada elemento inventariado.

### 8. Estacion

Se decidió crear una entidad que representara cada estación de trabajo, ya que necesitamos saber en qué sala, piso y edificio se localiza y a la vez necesitamos una manera de saber a qué máquina están asignados los periféricos.

### 9. TenerMouse, TenerTeclado, TenerHeadset

Se decidió crear esta relación para tener un inventario de los dispositivos periféricos. Como cada uno está asignado a una Estación es por eso que se utiliza como llave primaria en la tabla.

## 10. Cliente

La entidad Cliente fue creada principalmente para ser relacionada con Curso, pues se requiere guardar información tanto de los clientes que contratan un curso con TeleLlamadas, como de los clientes que no tienen cursos contratados.

## 11. Curso

Hay mucha información que se requiere almacenar sobre los cursos, por lo que decidimos crear su entidad. Esta entidad estará asociada a otras entidades: Cliente, que es quien contrata el curso; Sala, que es donde se imparte el curso; Entrenador, que es quien imparte el curso; Agente, que es quien asiste al curso.

## 12. Sala

Se decidió crear una entidad que represente cada uno de las salas en donde se imparten los cursos, esta entidad es débil con respecto a piso, que a su vez es débil con respecto a edificio, ya que una sala requiere del piso donde está situada para identificarse y un piso requiere del edificio donde se encuentra para identificarse.

## Atributos

### 1. **DiasEntrenamiento - AgenteTele**

En este caso podemos ver que el mismo es multivaluado y calculado ya que dependerá de los días que asista sus cursos serán los días de sus entrenamiento pero el mismo puede tener más de un día considerando que los cursos duran de 4 días, además no se traduce en el relacional por que tiene el peso de ser calculado y los atributos calculados no se ponen en el modelo Relacional

### 2. **Semana - Horario**

En este caso es compuesto pues queremos tanto la fecha en la que comienza como en la que termina la semana. Y es que si bien actualmente manejamos registros de semanas de 7 días, cabe la posibilidad de que en algún momento se quiera acortar esta cantidad al manejar semanas con únicamente 4,5,6, etc. días laborales.

### 3. **Proposito - Piso**

Es un atributo calculado que nos dice si el piso esta reservado para mantenimiento, para operaciones, o si tiene un uso mixto.

### 4. **AliasCliente - Cliente**

Es un atributo que contiene el nombre del Cliente. Se llama AliasCliente porque el nombre que tendrá puede ser de persona moral o de persona física (una empresa o una persona).

### 5. **Tipo - Sala**

Es un atributo que contiene una de estas dos cadenas: Operaciones o Capacitacion, pues la sala puede ser de uno de esos dos tipos.

## Relaciones

### 1. Impartir

Esta relación asocia a las entidades Curso y Entrenador. Su cardinalidad es *muchos a uno*, pues un Entrenador puede impartir muchos Cursos (bajo la restricción de impartir 1 curso a la vez), pero un Curso únicamente es impartido por un Entrenador. Además, la participación es parcial para Entrenador, ya que un Entrenador no requiere de impartir un Curso para existir, y es total para Curso, pues un Curso únicamente existe si es impartido por un Entrenador.

### 2. Tomar

Esta relación asocia a las entidades Curso y AgenteTele. Su cardinalidad es *muchos a uno*, pues muchos agentes toman un Curso y un Curso se da a muchos Agentes. Además, la participación es total para Curso y para AgenteTele, pues AgenteTele necesariamente deberá tomar un curso y un curso debe tener agentes que tomen el curso.

### 3. Contratar

Esta relación asocia a las entidades Curso y Cliente. Su cardinalidad es *muchos a uno*, pues un Cliente puede contratar muchos Cursos, pero un Curso únicamente es contratado por un Cliente. Además, la participación es parcial para Cliente, ya que un Cliente no requiere de la contratación de un Curso para existir, y es total para Curso, pues un Curso únicamente existe si es Contratado por un Cliente.

### 4. Estar

Esta relación asocia a las entidades Curso y Sala. Su cardinalidad es *muchos a uno*, pues en una Sala se pueden impartir muchos Cursos, pero un Curso únicamente puede ser asignado a una Sala. Además, la participación es parcial para Sala, ya que una Sala no necesita ser el lugar donde se imparte un Curso para existir, y es total para Curso, pues un Curso únicamente puede ser impartido si se le es asignada una Sala.

### 5. Pertener

Esta relación asocia a las entidades Piso y Sala. Su cardinalidad es *muchos a uno*, pues a un Edificio se pueden pertenecer muchas Salas, pero una Sala únicamente puede pertenecer a un Edificio. Además, la participación es total para Sala, ya que una Sala necesita pertenecer a un Edificio para existir, y es parcial para Edificio, pues un Edificio no necesita tener salas para existir.

### 6. TenerAgente

Esta relación asocia a AgenteTele y Asistencia Curso. Su cardinalidad es *uno a muchos*, ya que un agente puede tener muchas asistencias pues el curso al que está inscrito en este caso tiene una duración de 4 días en los que se le tomará asistencia.

Además cuenta con una participación de parcial del lado de agentes y obligatoria del lado de Asistencia Curso, ya que un agente puede nunca asistir al curso pero una asistencia se genera solo si el agente va al curso.

### 7. Generar

Esta relación asocia a las entidades Curso y Asistencia Curso. Su cardinalidad es *muchos a uno*, esto ya que un curso puede generar en el mismo muchas asistencias (con generar ese hace alusión a que al asistir un agente al curso hace que se genere la asistencia a curso), mientras que su participación es parcial del lado de curso y total del lado de Asistencia de Curso, ya que un curso podría no tener generar ninguna asistencia si los agentes no van al mismo, pero la Asistencia si necesita del curso, ya que es la asistencia al mismo.

## 8. Tomar

Esta relación asocia a AgenteTele y Curso. Su cardinalidad es *uno a muchos*, esto ya que un agente puede tomar más de un curso de acuerdo a las especificaciones dadas.

Es obligatoria de ambos lados ya que al menos el agente de tomar un curso por que es su trabajo y del lado de curso por que necesita de los Agentes para poder impartirse de otra forma no tiene caso tener un curso sin ningún Agente.

## 9. Tener

Esta relación asocia a Periferico y Estacion. Su cardinalidad es *uno a uno*, esto ya que cada periférico está en una estación.

Es opcional de ambos lados ya que puede que un periférico no esté asignado a una estación o que una estación no tenga periféricos.

## 10. Asignar

La relación sirve para asignarle un piso al entrenador.

Puesto que el entrenado debe tener un único piso asignado, pero a un mismo piso pueden acceder varios entrenadores, la cardinalidad es uno a muchos.

Se nos indica que todos los entrenadores tienen un piso asignado, por lo que la relación es de participación total del lado de Entrenador, sin embargo puede haber pisos que no tengan actividad por el momento y por eso la relación es de participación parcial del lado de Piso.

## 11. Ingresar

Tanto agentes como entrenadores ya tienen asignadas una cantidad de horas que deben de laborar. Sin embargo nos interesa saber exactamente cuando es que entran y salen de su piso asignado, pues no necesariamente ingresan al piso cuando deben, además de que puede que pasen más tiempo del obligatorio.

Una entrada o salida de un piso (acceso) puede ser realizada únicamente por un empleado, sin embargo ese mismo empleado puede hacer varios accesos a un piso por lo que la cardinalidad es uno a muchos.

Todos los accesos deben ser realizados por alguien, así tenemos que para Acceso la participación es total. Los entrenadores no necesariamente cumplen con sus horas asignadas en un piso, además de que pueden trabajar en línea, así la relación es de participación parcial del lado de entrenador.

## 12. Trabajar

La relación sirve para que podamos conocer el número de horas que se le asignan a la semana a un entrenador. Como esta cantidad no es necesariamente siempre la misma resulta necesario crear una relación con una tabla y no simplemente almacenar un dato único en la tabla Entrenador.

Cada horario de trabajo es específico de un entrenador, sin embargo el mismo entrenador tiene varios horarios (uno por cada semana que labora), así la cardinalidad es uno a muchos.

Todos los entrenadores tienen que tener horas asignadas puesto que se les contrato con la intención de que trabajaran, y todos los horarios deben pertenecer a un entrenador. Así la relación es de participación total para ambos lados.

# Llaves Foráneas

## 1. AgenteTele

### ■ **ON UPDATE** : CASCADE.

Esto es por que cuando cambien IdCurso, RFCCliente,IdSala,IdPiso,IdEdificio, CURPEntrenador en su tabla padre, queremos que en las tablas hijas lo haga también (en la tablas a las que hace referencia), ya que por ejemplo si cambia el Id del Curso queremos que el agente que tenga este Id siga teniendo la referencia de esta actualización pues el mismo toma el curso.

### ■ **ON DELETE** : SET DEFAULT

De acuerdo a nuestra implementación no sería adecuado poner como nulos los valores de las Fk's que forma parte de alguna PK's, por lo que un **SET NULL** no podría tener sentido alguno, mientras que en el caso de **CASCADE** nos eliminarla toda la fila referenciada por lo que el modo de arreglar estos vacíos es poner un **DEFAULT**.

## 2. Entrenador

### ■ **ON UPDATE** : CASCADE

Esto es por que cuando cambien IdPiso,IdEdificio, en su tabla padre queremos que en las tablas hijas lo haga también (en la tablas a las que hace referencia), ya que por ejemplo si cambia el Id del Piso queremos que el agente que tenga este Id siga teniendo la referencia de esta actualización pues el mismo esta asignado a ese Piso

### ■ **ON DELETE** :NOT NULL

Esto ya que en general si de la tabla padre se quita el IdPiso, realmente no se debería de quitar la fila referenciada si no más bien asignarla en NULL al IdPiso, para poder posteriormente asignarle otro Piso

## 3. Asistencia Curso

### ■ **ON UPDATE** : CASCADE

Esto es por que cuando cambien IdCurso, RFCCliente,IdSala,IdPiso,IdEdificio, CURPEntrenador, CURPAgente en su tabla padre queremos que en las tablas hijas lo haga también, ya que por ejemplo si cambia el Id del Curso queremos la asistencia que tenga este Id siga teniendo la referencia de esta actualización pues el mismo toma la asistencia a este curso en particular del agente.

### ■ **ON DELETE** : CASCADE

Por que si algunas de las llaves foráneas es quitada en su tabla padre , realmente no se podría asignar otro valor a futuro pues la asistencia ya hubiera sido tomada , así que simplemente se elimina la fila referencia en la tabla hija.

## 4. TelefonoCelAgente

### ■ **ON UPDATE** : CASCADE

Si por algún motivo de equivocación se tiene que actualizar el CURP ya sea que se escribió mal etc, entonces el teléfono de la persona sería el mismo por lo que se debería de actualizar en esta tabla en la que esta referenciado (tabla hija)

### ■ **ON DELETE** : CASCADE

Si no existe ya el CURP del agente entonces no sería necesaria seguir teniendo sus números de teléfono

## 5. CorreoElectronicoAgente

### ■ **ON UPDATE** : CASCADE

Si por algún motivo de equivocación se tiene que actualizar el CURP ya sea que se escribió mal etc, entonces el correo de la persona sería el mismo por lo que se debería de actualizar en esta tabla en la que esta referenciado (tabla hija)

### ■ **ON DELETE** : CASCADE

Si no existe ya el CURP del agente entonces no sería necesaria seguir teniendo sus correos electrónicos

#### 6. TelefonoCelEntrenador

- **ON UPDATE** : CASCADE

Si por algún motivo de equivocación se tiene que actualizar el CURP ya sea que se escribió mal etc, entonces el teléfono de la persona sería el mismo por lo que se debería de actualizar en esta tabla en la que esta referenciado (tabla hija)

- **ON DELETE** : CASCADE

Si no existe ya el CURP del Entrenador entonces no sería necesaria seguir teniendo sus números de teléfono

#### 7. CorreoElectronicoAgente

- **ON UPDATE** : CASCADE

Si por algún motivo de equivocación se tiene que actualizar el CURP ya sea que se escribió mal etc, entonces el correo de la persona sería el mismo por lo que se debería de actualizar en esta tabla en la que esta referenciado (tabla hija)

- **ON DELETE** : CASCADE

Si no existe ya el CURP del Entrenador entonces no sería necesaria seguir teniendo sus correos electrónicos

#### 8. CorreoElectronicoAgente

- **ON UPDATE** : CASCADE

Si por algún motivo de equivocación se tiene que actualizar el CURP ya sea que se escribió mal etc, entonces el correo de la persona sería el mismo por lo que se debería de actualizar en esta tabla en la que esta referenciado (tabla hija)

- **ON DELETE** : CASCADE

Si no existe ya el CURP del Entrenador entonces no sería necesaria seguir teniendo sus correos electrónicos

#### 9. TenerMouse , TenerTeclado , TenerHeadset

- **ON UPDATE** : CASCADE

Si se modifica la estación en la que estaba el periférico entonces necesita verse reflejado este cambio en la relación tener de los periféricos.

- **ON DELETE** : CASCADE

Si se borra la estación en la que estaba el periférico entonces ya no necesita existir la relación tener de los periféricos y la estación.

#### 10. Estacion

- **ON UPDATE** : CASCADE

Si se modifica la sala, el piso o el edificio en el que estaba la estación, estos cambios se deben de ver reflejados en la estación.

- **ON DELETE** : SET DEFAULT

Si se borra la sala, piso o edificio en el que estaba la estación entonces envíamos por default al piso 0, sala 0 y edificio 0.

## 11. Horario

- **ON UPDATE** : CASCADE

Esto es porque si modificamos el CURPEntrenador por cualquier razón, necesitamos que para el entrenador cuyo CURPEntrenador fue modificado se pueda seguir teniendo acceso a las horas que tiene que trabajar cada semana.

- **ON DELETE** : CASCADE

En caso de que eliminemos el CURPEntrenador de algún entrenador, es por que eliminamos al entrenador y ya no nos interesa seguir almacenando su información, incluyendo las horas que pudo haber trabajado en la empresa.

## 12. EntradaEntrenador, SalidaEntrenador

- **ON UPDATE** : CASCADE

Si modificamos el CURPEntrenador, necesitamos que para el entrenador cuyo CURPEntrenador fue modificado se pueda seguir teniendo acceso a su información sobre entradas y salidas a su piso.

- **ON DELETE** : CASCADE

En caso de que eliminemos el CURPEntrenador de algún entrenador, es por que eliminamos al entrenador y ya no nos interesa seguir almacenando su información, incluyendo su información sobre entradas y salidas de su piso

## 13. EntradaAgente, SalidaAgente

- **ON UPDATE** : CASCADE

Si modificamos el CURPAGente, necesitamos que el Curso que está establecido siga haciendo referencia a la sala que fue modificada.

- **ON DELETE** : CASCADE

En caso de que eliminemos alguna sala, requerimos conservar la información del curso.

## 14. Sala

- **ON UPDATE** : CASCADE

Si modificamos el edificio o el piso al que pertenece, tiene sentido que se refleje esta modificación en las salas correspondientes a este piso o edificio dado que comparten la misma localidad física.

- **ON DELETE** : CASCADE

Si eliminamos un edificio, dado que el piso ya no tiene una localidad física a la cual pertenecer y a su vez, la sala tampoco, tiene sentido que este se elimine también.

## 15. TelefonoCliente, CorreoCliente, PersContCliente

- **ON UPDATE** : CASCADE

Si por algún motivo se tiene que actualizar el RFC del Cliente, ya sea que se escribió mal etc, entonces el telefono, el correo y la persona de contacto del Cliente sería el mismo por lo que se debería de actualizar en esta tabla en la que esta referenciado (tabla hija)

- **ON DELETE** : CASCADE

Si no existe ya el RFC del Cliente entonces no sería necesario seguir teniendo su teléfono, el correo y la persona de contacto.

## 16. Curso

- **ON UPDATE** : CASCADE

Si por algún motivo se tiene que actualizar alguna llave foránea que conforme la llave primaria de Curso, entonces este cambio deberá verse reflejado en la tupla correspondiente en Curso.

- **ON DELETE** : SET DEFAULT

Si se borra alguna tabla que contenta algun atributo de conforma la llave primaria del curso, nos gustaría seguir manteniendo la información del curso, así sea con un atributo por default en la columna correspondiente.

## 17. FechasCurso

- **ON UPDATE : CASCADE**

Si por algún motivo se tiene que actualizar algun atributo de otra tabla que compone la llave primaria de la tabla FechasCurso, como pueden ser RFCCliente en Cliente, IdSala, IdPiso, IdEdificio en Sala o CURPEntrenador en Entrenador, entonces las Fechas del Curso serían las mismas, por lo que se debería de actualizar en esta tabla en la que esta referenciado (tabla hija)

- **ON DELETE : CASCADE**

Si no existe ya el alguno de estos artibutos: RFCCliente en Cliente, IdSala, IdPiso, IdEdificio en Sala o CURPEntrenador en Entrenador, entonces no sería necesario seguir teniendo su sus Fechas.

18. Piso

- **ON UPDATE : CASCADE**

Si modificamos el edificio al que pertenece, tiene sentido que se refleje esta modificación en todos los pisos correspondientes a este edificio dado que comparten la misma localidad física.

- **ON DELETE : CASCADE**

Si eliminamos un edificio, dado que el piso ya no tiene una localidad física a la cual pertenecer tiene sentido que este se elimine también.

## **Generales**

En esta sección se encuentran restricciones generales que no podrían ponerse en las subSecciones anterior.

- Se nos dice que a un Agente se le asigna un Cliente , lo que se considero en el diseño es dado que el Agente puede tomar más de un curso se le asignara un único Cliente por cada curso en el que este.

# Dependencias funcionales

- **Tabla Edificio**

**IdEdificio** ← Pais, Estado,Ciudad,Calle, Numero,CP

- **Tabla Piso**

**IdPiso, IdEdifico** ← Estatus

- **Tabla Sala**

**IdSala, IdPiso, IdEdificio** ← Costo, Tipo

- **Tabla Mouse**

**IdPeriferico** ← FechaAdqui, Modelo

- **Tabla Headset**

**IdPeriferico** ← FechaAdqui, Modelo

- **Tabla Teclado**

**IdPeriferico** ← FechaAdqui, Modelo

- **Tabla Estacion**

**IdEstacion, IdSala, IdPiso, IdEdificio** ← SistemaOp

- **Tabla AgenteTele**

**CURPAgente** ← NombreC,FechaNac,CP,Ciudad,Estado,Calle,Numero, Pais,Fotografia,  
PagoAgente

- **Tabla Entrenador**

**CURPEntrenador** ← NombreC,FechaNac,FechaIngreso,NumeroSeguroSoc,Fotografia,  
CP,Ciudad,Estado,Calle,Numero, Pais

- **Tabla Cliente**

**RFC** ← AliasCliente,RazonSocial

- **Tabla Curso**

**IdCurso, RFCCliente, IdSala, IdPiso, IdEdificio, CURPEntrenador** ← Nom-  
bre,Modalidad,HorasDeEntrenamiento,PagoEntrenador

- **Asistencia Curso**

**IdAsistenciaCurso,IdCurso,RFCCiente, IdSala, IdPiso, IdEdificio, CURPEn-  
trenador,CURPAgente** ← Fecha, HorasAsistencia



