

**ESPECIFICACIÓN DEL PROYECTO FINAL  
IF -4100 FUNDAMENTOS DE BASES DE DATOS**

**VALOR: 20%**

**Fecha de entrega: 2 de diciembre del 2020**

**Indicaciones generales:** Trabaje en grupos de tres personas. El caso por desarrollar lo encuentran en el documento llamado COMPANIA TELEFONICA. Siga las instrucciones abajo explicadas. En este proyecto deberá utilizar la metodología de desarrollo de bases de datos relacionales estudiada durante el curso y debe entregar todo lo indicados en este enunciado. Para el modelo físico utilice SQL Server o MySQL. Debe tener la base de datos montada en un servidor que los docentes puedan acceder y además de tener un respaldo local en caso de algún incidente. Se comprobará que el backup funciona, pues deberán restaurarlo en la defensa.

Importante: Dentro de su documento entregable use la misma numeración que se presenta a continuación.

**1. Nombre de los DBAs:**

---

**2. Documento de Modelado Conceptual de la base de datos (30%) con lo siguiente:**

Modelo Entidad/Relación (diagrama pata de gallo o Chen) que incluye:  
Entidades con atributos, Relaciones (con atributos y flechas, si hubiera),  
Especializaciones, llaves, cardinalidades de mapeo, participaciones.  
Sección de supuestos semánticos.  
Sección de semántica no reflejada en el diagrama.

**3. Documento de Modelado Lógico de la base de datos (del producto total) con lo siguiente:**

- a. Transformación al modelo relacional. Aplicación de las reglas de mapeo paso por paso (debe especificarse particularmente cada paso aplicado en cada relación). 5%

NOTA: **no** haga aquí el cuadro con la transformación de supuestos, sino que estas transformaciones las escribe en el Diccionario de datos (punto e, columna Restricciones)

- b. Dependencias funcionales y multivaluadas contenidas en cada relación. 5%

Incluya las siguientes dependencias en su modelo:

En LLAMADA: Costo → FechaInicial;

Id\_Franja → Costo

En FACTURA: PtsRedimidos → ImpPtsRedimidos (en caso de que se rediman puntos, se deberá cobrar un monto fijo por concepto de impuesto por aplicar puntos).

FechaFin → Desc\_epoca (depende de la estación se aplica un descuento).

CodRango, Id\_Franja → Desc\_FR\_Ran

- c. Proceso de Normalización 10%

d.1) Recubrimiento minimal (determinar si son elementales, identificar extraños, eliminar redundancia)

- d.2) Subrayar las claves para cada relación, cuando se requiera (porque no tiene la clave de antemano), debe especificar el proceso de cálculo de claves.
- d.3) Identificar la forma normal en la que está cada relación (justificar sólo cuando no está en FNBC).
- d.4) Normalizar cada relación hasta 3 FNBC. Debe desarrollar el procedimiento.

**e. Documentar un diccionario para su base de datos con lo siguiente, para cada tabla que decida programar en el punto 4.a: 10%**

- Nombre de la columna(atributo)
- Tipo de datos
- S/N, S= si acepta nulos, N= no acepta nulos
- Pistas de auditoría
- Breve descripción de la columna
- Constraints, pueden ser:  
 PK: llave primaria, U: llave única, FK# llave externa (un # para c/fk diferente),  
 Cuando la PK o FK están compuestas por más de un atributo y se validan hacia una única tabla entonces se agrupan, se especifica la tabla y entre paréntesis las claves.  
 CK: check indique en qué consiste la restricción,  
 NN: not null ,  
 Checks, Defaults, etc.

Ejemplo: Empleado

Columna	Tipo de datos	Nulabilidad	Descripción	Restricciones
Cedula	Char(9)	N	Número de cédula	PK1
Puesto	Integer	N	Número de puesto	FK1 Puesto(codPuesto)
EstCivil	Char(1)	S	Estado civil	CK EstCivil in ('S','C','V','D','U')
Depto	Integer	N	Departamento	FK2 Plaza(CodDepto, CodPlaza)
Plaza	Integer	N	Número de Plaza	
Num_Hijos	Integer	S	Número de Hijos	Constraint default a 0

**NOTA: TODA la base de datos debe estar en inglés, tal como se trabajó en los laboratorios.**

**4. Modelo físico de la base de datos (20%)**

- a. **Codifique un Script abreviado del Modelo Físico de la base de datos.** Cree por lo menos lo siguiente:
- i. Al menos 3 tablas que se relacionen (1:1, 1:N o M:N). Deben incluir las PKs y FKs respectivas.
  - ii. Configure 1 tipo de borrado y 1 tipo de actualización diferente del restringido, documentando la decisión en los supuestos semánticos de ser necesario.
  - iii. Configure pistas de auditoría (usuarios, fechas y estado de los registros). Estas pistas necesitan usar constraints default (en el caso de las fechas) y de check (en el caso del estado: 'A', 'I' (activo, inactivo, otro))

- iv. Cree una vista que consolide la visualización compleja de data de las 3 tablas, y que responda a una consulta de negocio que se podría necesitar mostrar en la aplicación que usa la BD (ustedes deciden cuál pregunta de negocio responder)

## 5. Aplicativo que utilice la base de datos (20%)

**Construya una aplicación (en el lenguaje que desee) que le permita utilizar la base de datos y realizar operaciones CRUD en ella.** La aplicación debe contar con lo siguiente:

- i. Puede tener UI o simplemente basarse en código puro. Se le sugiere tomar como referencia la app utilizada en algunos de los laboratorios del curso.
- ii. La app puede ser de escritorio o web. Se recomienda lo segundo para que vayan preparándose para IF4101.
- iii. La app debe permitir operaciones INSERT, SELECT, UPDATE y DELETE a las tablas creadas en el modelo físico. En el caso de INSERT y UPDATE, estas deben incluir todos los campos de la tabla. En el caso de que la tabla tenga FKs, no se les permite insertar o actualizar escribiendo la llave directamente. Deben mostrar en UI lo necesario para hacer esa asociación (dropdowns u otro) o bien, utilizando objetos a nivel de código si es que no codifican UI.
- iv. Los SELECT deben utilizar JOINS (del tipo que más tengan sentido y concuerden con lo que modelaron en los primeros puntos). Deben haber por lo menos 3 selects y al menos 2 deben utilizar JOINS.
- v. En alguna sección de la app se debe consumir la vista creada en el punto 4 y mostrar en pantalla los resultados.
- vi. Al borrar o actualizar tuplas de aquellas tablas que tengan borrados/updates diferentes al restringido, se debe alertar al usuario de lo que sucederá si se elimina/actualiza dicha información.
- vii. Los constraints que por su naturaleza generen algún error de control en base de datos como parte de su funcionamiento deben manejarse en la aplicación. Se debe mostrarle al usuario cuál es el inconveniente. Ejemplo: Si un campo en BD no permite nulos o permite solamente datos de un tipo en particular o solo ciertos datos (en el caso de los checks) y el usuario incurre en el error de ingresar datos incorrectos o nulos, la app debe “atrapar” esas excepciones que le lanza la BD y mostrarle al usuario un mensaje que le permita ver qué sucede y qué error es en el que está incurriendo. Esto también aplica en caso de que, por el diseño, haya alguna relación con borrado restringido y entonces se trate de borrar al “padre” que tiene “hijos”
- viii. alguna operación que se refleje en la app debe utilizar algún operador SQL que se haya estudiado en clase y otro que no. Ejemplo: SUM, AVG, etc (los que sí se estudiaron) y otros que no se hayan estudiado (UCASE, etc)
- ix. La codificación del back end de la app debe ser en inglés es su totalidad. El front-end puede ser en español (solo lo gráfico), ya que las empresas en los casos son hispanohablantes.
- x. Se les recomienda trabajarla en un repositorio versionado al que le darán acceso a los docentes (github, SVN, TFS, etc)