



PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
DEPARTAMENTO DE CIENCIA DE LA COMPUTACIÓN
IIC2413 - BASES DE DATOS

Proyecto semestral Etapa 3 2025-2

17 de octubre de 2025

Administrativo

- LEA LAS INSTRUCCIONES DEL ENUNCIADO GENERAL
- **Fecha de inicio:** viernes 17 de octubre
- **Fecha de entrega:** lunes 3 de noviembre de 2025, 23:59.
- **Fecha y hora máxima de entrega atrasada:** jueves 6 de noviembre de 2025, 23:59.
- **Consultas solo por ISSUES** se responden en forma consolidada los jueves y sábado recepción de issues hasta miércoles 29 de octubre
- Revise las DISCUSSIONS de Github antes de preguntar. Solo se responderán las consultas originales.
- **Entregables:** Informe en formato "Markdown" (README.md) según la plantilla entregada. Programa PHP de data cleansing, archivo sql de creación y carga de BDD.
- **Lugar y forma de entrega:** En su cuenta personal <usuario >.e3 directorio E3 del servidor stonebraker.ing.uc.cl.
- **Seguridad:** SIEMPRE guarde un respaldo del trabajo (por ejemplo en Onedrive). No se otorgará tiempo extra por problemas o indisponibilidad de los computadores personales. Recuerde que trabajar en el computador personal es OPCIONAL, se puede hacer todo el trabajo en el server.
- **Corrección del trabajo:** Los programas PHP y SQL que se usará para su entrega serán 'main.php' y 'carga.sql' que serán ejecutadas en forma secuencial mediante la línea de comandos de linux de su cuenta en el servidor.
- **Buenas prácticas** Use las buenas prácticas de programación Avanzada, de lo contrario puede tener descuento en el puntaje. *Solo se permite el uso de PHP y SQL, no está*

permitido la manipulación manual de los datos en los archivos originales (o copias de ellos) o por medio de algún otro software (P.Ej. Excel)

- **Integridad** Solo está permitido el código generado por usted, el entregado en clases o ayudantía, no está permitido el uso de códigos de otras personas, repositorios públicos o privados ni código generado por inteligencia artificial. En caso de cometer una falta a la integridad está disponible el **Formulario de arrepentimiento temprano**: [Link Arrepentimiento Temprano](#) (hasta 6 de octubre)

Objetivo de la etapa

El objetivo para esta etapa del proyecto es que el estudiante se familiarice con el lenguaje PHP, en el sistema operativo Linux-Ubuntu mediante el procesamiento y limpieza de datos almacenados en archivos y cargarlos en una base de datos nueva vacía.

1. Contexto

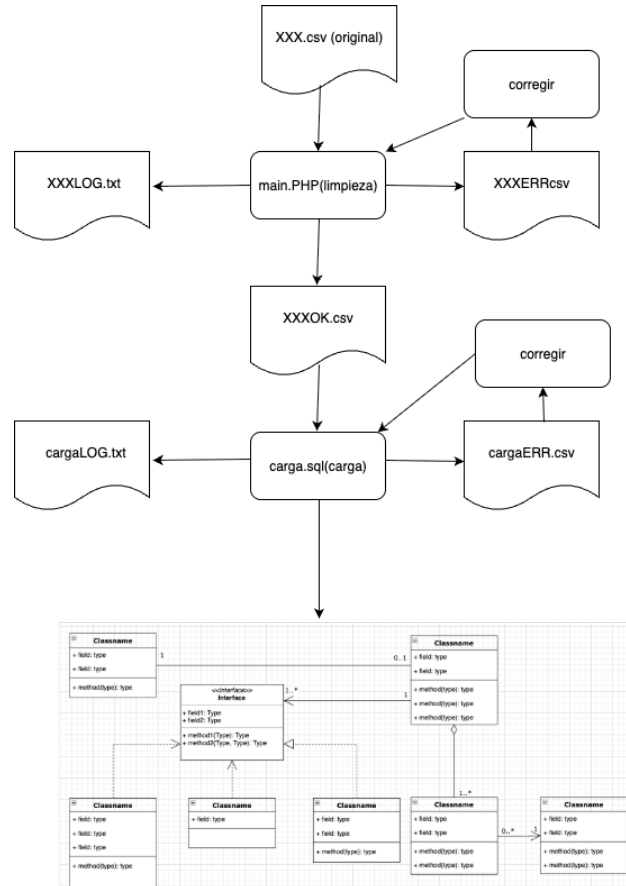
Usted ya sabe como modelar un problema de base de datos usando los modelos Entidad Relación (E1) y también trabajar manipulando datos de una BDD a través de dos herramientas, un ERD Tool y PSQL (E2). Ahora le encargan construir la base de datos del centro médico DCColita a partir de datos exportados de la base de datos real del centro médico.

El proveedor actual, quien perdió la licitación para la renovación del sistema, no le va a facilitar el trabajo, así que le entrega una serie de archivos similares, pero no iguales a los de la etapa 1 para que Ud. construya la base de datos desde cero. Estos datos estaban almacenados con criterios diferentes a los que se requieren en este nuevo sistema (otras restricciones de integridad), por lo que pueden tener errores de formato (sintaxis), tipo, duplicaciones, datos faltantes e incoherentes. Su misión es tomar estos archivos y, mediante exclusivamente PHP y SQL, realizar un proceso de limpieza (data cleaning) y carga de los mismos en la base con un mínimo de pérdida de información. Las acciones a realizar pueden ser las siguientes:

- Si el registro se puede reparar, por ejemplo, con información adicional, se repara y se reporta. Ejemplo correo mal formado jdoe@gmail..com (el . está repetido).
- Si no hay información como para repararlo, pero el campo es obligatorio, puede ingresar un registro que cumpla el formato, pero el valor indica que no es válido. Ejemplo: teléfono campo obligatorio, pero aparece 2419 lo que no cumple con ser un teléfono válido, entonces se cambia a 100000000.
- En último caso y solo si están todas las posibilidades agotadas, si el registro es irreparable y esencial para la tupla se elimina en cascada y se reporta.

2. Encargo para la etapa 3

Ud. realizará un proceso de data cleaning usando PHP previo a la carga de los datos en SQL según el procedimiento de la figura.



- (2 pts.) Ud. basándose en el esquema de la E2, deberá crear un (o modificar el) esquema de bases de datos en 3NF o BCNF para almacenar los datos provenientes de los archivos XXX.csv entregados. Muestre el nuevo esquema en el informe.
- (5 pts.) Ud. mediante PHP, deberá revisar, identificar y corregir en forma iterativa, hasta donde sea posible, los datos de cada archivo de entrada XXX.csv y entregar un archivo con datos XXXOK.csv con todos los datos correctos o corregidos. Los registros de datos malformados que sean irreparables por su programa PHP deberá traspasarlos al archivo XXXERR.csv y registrarla en el archivo XXXLOG.txt (si hay una modificación que se realiza repetidamente se reporta solo una vez indicando los registros afectados). Luego deberá inspeccionar el archivo XXXERR.csv en busca de errores reparables que no fueron capturados en la ejecución del programa, modificarlo y volver a correrlo. Este ciclo se debe hacer cuantas veces sea necesario hasta que la mayoría de los registros pasen a la etapa siguiente. El resultado es un programa PHP que reuna contenga todos los cambios que Ud. hizo en las iteraciones de modo que los

registros generados en los XXOK.csv tome la mayoría de los casos.

3. (10 pts.) Crear el archivo carga.sql que mediante sentencias SQL i) cree el esquema con todas sus tablas, atributos, dominios, restricciones, PK, FK, SK, etc. ii) cargue, a partir de todos los archivos XXXOK.csv, los datos a la instancia de la BD. El resultado del proceso de carga entrega (al igual que antes) el archivo (esta vez solo uno) cargaERR.csv y cargaLOG.txt conteniendo las tuplas con errores (en cargaERR) y la acción tomada (en cargaLOG), el proceso carga.sql podrá ser ejecutado y modificado cuantas veces sea necesario para que la pérdida de información sea mínima (el mínimo de tuplas descartadas). Al final Ud. entregará solo un SQL carga.sql conteniendo el acumulado de todas las modificaciones creado por usted
4. (3 pts.) Informe conteniendo:
 - a) archivo Readme.md conteniendo y justificando i) el esquema de bases de datos creado ii) todas las modificaciones a cada archivo XXX.csv originales (están en XXXLOG.txt) iii) todas las acciones realizadas a los datos del archivo XXXOK.csv efectuadas para ser cargadas en la base de datos.
 - b) Todos los archivos .csv originales y procesados, los .txt y .sql en el directorio E3.
 - c) Como Ud. ya corrió numerosas veces main.php y carga.sql los programas finales están suficientemente afinados. Para evaluar, se ejecutarán tanto el mail.php y luego carga.php UNA SOLA VEZ sobre los archivos xxx.csv originales y se espera tener como resultado i) la instancia de la base de datos cargada, los archivos XXXERR y cargaERR con el mínimo de tuplas irreparables. Asimismo, los XXXLOG y cargaLOG con las modificaciones hechas a las tuplas reparadas y eliminadas.
 - d) Instrucciones para ejecutar los dos programas: main.PHP y carga.sql

Por ejemplo, el Readme debe contener una tabla como la siguiente que se construye a partir del XXXLOG.csv y cargalog.csv

Error	archivo encontrado	solución	solucionado por PHP	Solucionado SQL
mal formato de correo	Persona	corregir	PHP	

Ud. NO DEBERÁ entregar otros archivos que no correspondan exactamente a la tarea (versiones anteriores, respaldos, pruebas, etc.) si quiere hacer pruebas use un directorio diferente a E3 o la cuenta .bdd y una vez finalizado traspase su trabajo definitivo a la cuenta usuario.e3/E3

3. Reglas de negocio

1. Una persona puede ser paciente, trabajar en el centro médico o ambos. Los trabajadores se dividen en staff médico o administrativo.
2. Solo los médico/as tienen especialidad y realizan atenciones médicas, emiten recetas y órdenes.
3. Una persona con rol staff médico o administrativo también puede ser paciente, pero en este caso solo puede ser titular en la Institución de salud previsional
4. Una persona (independiente del rol) puede estar afiliada a una institución de salud previsional (una ISAPRE o FONASA) o no tener previsión, en cuyo caso se considera como particular.
5. Cada ISAPRE tiene un solo plan de salud que especifica el porcentaje de bonificación por cada grupo del Arancel FONASA. Es decir, todos los beneficiarios de cada ISAPRE tienen el mismo plan. Sin embargo, las bonificaciones por grupo varían de una ISAPRE a otra.
6. En el contexto de la institución de salud previsional, una persona puede ser Beneficiario titular o solo beneficiario. En este segundo caso la persona tiene asociado un beneficiario titular. Un beneficiario solo puede estar registrado en una institución y el beneficiario está en la misma institución que el titular
7. Si el beneficiario (titular o no titular) está en ISAPRE (no FONASA) paga el valor de la atención y del arancel del centro médico menos la bonificación. Si el beneficiario (titular o no titular) está en FONASA paga solo el arancel FONASA, no tiene bonificación. Si la persona es particular (o sea no es FONASA NI ISAPRE) paga el valor completo de la atención. Los medicamentos se pagan al precio de lista de la farmacia independiente de su tipo y previsión de salud.
8. Toda atención comienza con una consulta médica, en esta el médico realiza un diagnóstico y pueden generar una o más recetas médicas y una o más órdenes de procedimientos o exámenes. Todos los medicamentos de las recetas están registrados en el Maestro farmacia y todos los procedimientos o exámenes están registrados en el Arancel FONASA.
9. las órdenes pueden contener muchos exámenes y procedimientos
10. las recetas de medicamentos no psicotrópicos pueden tener muchos medicamentos, los psicotrópicos van uno por receta.

4. Formato de archivos de datos

Persona

- ID: integer PK, Identificador único de la persona en el centro médico

- RUN: text 10 NOT NULL formato XDDDDDD-A X=1..9 D=0..9 A=0..9+K, Rol único nacional de la persona
- nombre: text 30 NOT NULL, nombres de la persona
- apellido: text 30 NOT NULL, apellidos de la persona
- dirección: text 100, Domicilio de la persona
- correo: text formato partelocal@dominio.TLD RFC 3696, dirección de email
- telefono: formato XDDDDDDDD X=1..9, D=0..9, número telefónico de la persona
- tipo: { beneficiario, titular, NULL}, Tipo de beneficiario registrado en la Institución de salud previsional
- titular: {formato RUN XDDDDDD-A X=1..9 D=0..9 A=0..9+K, NULL}, RUN del titular en la institución de salud previsional.
- rol: {Staff médico, administrativo, paciente, NULL}<,Paciente>, Rol(es) de la persona en el centro médico
- profesion: {TENS, enfermero/a, Kinesiólogo/a, médico/a, NULL}, Profesión de la persona
- especialidad: text 30, Especialidad médica, exclusivamente para personas de profesión médico/a
- firma: text: 30, Nombre y ubicación de un archivo conteniendo la imagen de la firma de la persona
- InsSalPrev: text 30, Nombre de la institución de salud previsional

Instituciones previsionales de salud

- codigo: integer NOT NULL, Código registrado en MINSAL
- nombre: text 30 NOT NULL, Nombre de la institución
- tipo: {abierta, cerrada}, Tipo de ISAPRE, abierta recibe a cualquier persona, cerrada solo un grupo seleccionado, por ejemplo una empresa.
- RUT: text Formato XD.DDD.DDD-A X=6..9 D=0..9 A=0..9 o K NOT NULL, Rol único tributario
- enlace: text Formato {<http<s>//>, NULL}, dirección del servidor.TLD en internet TLD=Top level domain

Farmacia

- cod: integer NOT NULL, Código genérico del MINSAL
- nombre: text 100 NOT NULL, Nombre producto genérico

- descripcion: text 256 NOT NULL, Descripción producto genérico
- tipo: {Alimentos, Equipamiento, Fármacos, insumos, psicotrópicos, Refrigerados, Sueros}, Tipo producto
- codonu: integer, Código ONU
- clasonu: text 30 acepta nulos si el codonu es nulo, Clasificación ONU
- clasificacion: text 50 NOT NULL, clasificación interna
- estado: {activo, inactivo}, Estado del código genérico. Si está inactivo el producto está discontinuado, ya no disponible pero en el pasado existió.
- esencial: {0,1}, Producto miembro de la canasta Esencial de Medicamentos
- precio: integer, valor de venta del producto.

Arancel DCColita

- codigo: integer, Código interno de la consulta o atención médica en DCColita
- codFonasa: text Formato integer<-integer>, Código Fonasa y código adicional si existe
- atencion: text 100, Consultas y Atención Médica
- valor: integer, valor DCColita de la Consulta y Atención Médica

Arancel Fonasa

- codF: integer, Código único FONASA de la Consulta y Atención médica
- codA: integer, Código adicional FONASA que tienen algunos codF
- atencion: text 100, Consultas y Atención Médica
- valor: integer, valor FONASA de la Consulta y Atención Médica
- grupo: text 30, Grupo de Consultas y Atención Médica
- tipo: text 30, Tipo de Consultas y Atención Médica

Atencion

- ID PK: integer, identificación de la atención
- runpaciente: NOT NULL, RUN del paciente en formato RUN visto anteriormente
- runmedico: NOT NULL, RUN del médico/a en formato RUN visto anteriormente
- diagnostico: text 100, {diagnóstico del paciente, NULL}, Diagnóstico resultado de la atención, solo en caso que la consulta haya sido efectuada,
- efectuada: boolean true si la consulta se efectuó lo que implica que tiene diagnóstico

Medicamento

- IDAtencion: integer FK: ID de la atención
- nombre: text 100, Nombre producto genérico en farmacia
- posología: text 100, dosis y frecuencia de administración del medicamento
- psicotropico: boolean, true para medicamento psicotrópico

Orden

- IDAtencion: integer FK: ID de la atención
- IDArancel: integer FK codigo, Código interno DCColita
- consulta: text 100, Consultas y Atención Médica

Plan

- bonificación: integer, porcentaje de descuento sobre el precio del arancel DCColita el rango 0 a 100, NULL(equivalente a 0)
- Grupo: text 100, Grupo de consulta y atención médica