



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

Enunciado general

8 de marzo de 2025

1º semestre 2025 - Profesores E. Bustos - C. Álvarez

Objetivos generales del proyecto

- Manejar el lenguaje de programación de PHP.
- Construir modelos relacionales de datos e implementarlos en el contexto de bases de datos.
- Elaborar consultas de mediana complejidad utilizando el lenguaje de consulta SQL.
- Escribir procedimientos almacenados y triggers sencillos usando el lenguaje nativo.
- Lograr que una interfaz web se comunique con la base de datos.

1. El proyecto

El presente proyecto, se basa en la gestión de la base de datos de una página web¹ la cual permite a las personas que la utilicen organizar sus viajes de manera eficiente, integrando la reserva de transportes, actividades recreativas (denominadas panoramas) y hospedajes en un solo sistema centralizado :-).

Cada usuario podrá crear y gestionar múltiples agendas de viaje. Estas agendas servirán como contenedores para las distintas reservas que un usuario realice y podrán ser etiquetadas para facilitar su identificación y clasificación. Una agenda puede contener múltiples reservas, pero cada reserva está asociada a una única agenda. En pocas palabras, una agenda representa el conjunto de todas las reservas de un viaje en específico, para así facilitar su identificación al momento de consultar información de viajes.

¹no trabajaremos con la página web, pero sí con su base de datos

Las reservas pueden clasificarse en tres categorías principales:

- **Transporte:** Incluye la reserva de pasajes en trenes, buses o aviones (cada pasaje está asociado a una única reserva). Cada transporte tiene asociado un empleado responsable de su operación, identificado dentro del sistema mediante su correo corporacional (puede haber un empleado en varios transportes).
- **Panoramas:** Representan actividades recreativas o turísticas. Cada reserva de panorama realizada por un usuario permite el registro de acompañantes como participantes, donde solo se necesita su nombre completo.
- **Hospedaje:** Permite la reserva de estancias en hoteles o alojamientos tipo Airbnb. Cada reserva de hospedaje estará vinculada a un establecimiento específico y su respectiva disponibilidad.

Todas las reservas tienen un estado de disponibilidad, el cual indica 'Disponible' si es que se puede tomar la reserva o 'No disponible' si la reserva ya está tomada (cuando un usuario crea su propia reserva, se crea con estado 'No disponible').

El sistema maneja una entidad fundamental llamada **persona**, la cual se divide en dos subcategorías: **usuarios** y **empleados**. Los usuarios son aquellos que utilizan la plataforma para gestionar sus viajes y realizar reservas, además puede realizar reviews de las reservas y comprar seguros asociados a la reserva, mientras que los empleados están vinculados exclusivamente a los medios de transporte, desempeñando roles operativos en la conducción de los mismos.

A través de esta plataforma, se busca proporcionar un sistema estructurado y eficiente para la gestión de reservas, permitiendo a los usuarios organizar sus itinerarios de manera intuitiva y personalizada. Además, la centralización de las reservas en un solo sistema facilitará la administración y el control de los diferentes servicios ofrecidos en este mundo ficticio.

Es importante mencionar que para cada entrega, se les dará un set de datos con el cual van a tener que trabajar según se les pida en el enunciado.

2. Importante: Corrección de la tarea

Para las entregas, el carácter funcional del programa será el pilar de la corrección. Se recomienda realizar con tiempo pruebas periódicas en el servidor, no se revisarán entregas de forma local.

3. Restricciones y alcances

- La tarea es estrictamente individual.

- El programa debe estar desarrollado en PHP
- Se deben respetar las especificaciones del enunciado.
- No se permite el uso de librerías externas no autorizadas.
- Se debe incluir un archivo `README.md` con información clara del proyecto.
- No se puede utilizar inteligencia artificial.

4. Fechas de entrega

Evaluación	Publicación	Entrega	Puntaje
Entrega 0	7 de marzo	30 de marzo	10pts
Entrega 1	31 de marzo	13 de abril	20pts
Entrega 2	14 de abril	27 de abril	30pts
Entrega 3	5 de mayo	8 de junio	40pts

5. Política para entregas atrasadas

Cada día de atraso en la entrega de la etapa tiene un descuento de $1/6$ del puntaje de la etapa con un máximo de 3 días, luego de eso la calificación de la etapa es 0 puntos. Cada estudiante cuenta con 3 cupones de atraso en el semestre, cada uno equivale a 1 día extra en el plazo de entrega de la etapa. Para ello se levantará un formulario al final de cada etapa y tendrá 3 días de plazo para enviarlo.

6. Aclaración de dudas

Cualquier consulta que tengan relacionada al enunciado o dudas de código deben ser realizadas mediante **issues de github**. Estas deben venir con su label indicando la entrega a la que corresponde. El título de la issue debe tener su tipo indicado como: [DUDA ENUNCIADO] o [DUDA CODIGO], luego debe tener una breve descripción del problema y luego en la descripción de la issue su problema en profundidad y agregando imágenes si gustan para facilitar la ayuda. Las issues se contestarán de manera acumulativa, es decir, se contestarán los días miércoles y viernes, y se recibirán solo hasta las fechas indicadas en el programa como "Fin de consultas".