Pontificia Universidad Católica de Chile
Departamento de Ciencia de la Computación
IIC2413 - Bases de Datos

Proyecto semestral Etapa 0

17 de marzo de 2025

Administrativo

- **El proyecto es individual**
- **Fecha de entrega:** lunes 30 de marzo de 2025, 23:59.
- **Fecha y hora máxima de entrega atrasada:** míercoles 2 de abril de 2025, 23:59.
- **Entregables:** Informe en formato README.md, archivos PHP y archivos CSV con los resultados. Se indica la estructura de entrega en el item 9.
- **Lugar de entrega: Directorio E0 de su cuenta personal** del servidor del curso (bdd1.ing.puc.cl). **NO SE EVALUARÁN OTROS DIRECTORIOS. Los archivos contenidos en el directorio deben ser exclusivamente los necesarios para la tarea, no se deben incluir versiones anteriores o respaldos de sus archivos.**
- **Formularios importantes:**
  - **Formulario de uso de cupones:** Link Cupones (hasta 2 de abril)
  - **Formulario de arrepentimiento temprano:** Link Arrepentimiento Temprano (hasta 2 de abril)

Solo está permitido el código generado por usted y 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.

El programa PHP que se ejecutará para su entrega es 'main.php' y será ejecutado mediante la linea de comandos de linux default de su cuenta en el servidor.

Ejecución de la tarea: Aunque la tarea puede ser desarrollada íntegramente en forma local en un equipo propio, se debe tomar en cuenta que los sistemas operativos (MacOS, Windows, etc.) son diferentes en muchos aspectos a Linux Ubuntu, que es el sistema operativo del

servidor de bases de datos. La tarea será ejecutada **EXCLUSIVAMENTE** en el servidor mencionado. Es por esto que usted debe asegurarse que su trabajo se ejecute correctamente en el servidor.

Para esta entrega, el carácter funcional del programa PHP será el pilar de la corrección y **NO SE REVISARÁN** entregas en otro lenguaje de programación.

Recomendaciones de seguridad:

Se recomienda subir y probar su entrega en el servidor al menos 3 días antes de la fecha de término para cada etapa, de este modo verificando que funcione correctamente. Usted también puede ir subiendo diferentes versiones (draft, release candidate, etc) de su entrega al servidor e ir afinando sus resultados.
Finalmente, alternativamente se puede trabajar directamente desde el servidor provisto para el curso. Por lo que no se aceptarán argumentos de indisponibilidad del PC donde estaba el trabajo ni que el servidor estaba sobrecargado los últimos días. Tome todas las precauciones con antelación y suba respaldos periodicamente.

Consultas

Las consultas referentes al proyecto se recibirán EXCLUSIVAMENTE a través de ISSUES de github y se responderán de forma acumulada los días miércoles y viernes por Discussions en la misma plataforma. Esto ayuda a que las respuestas sean comunes y consistentes a todos en el curso. Los ayudantes no tienen autorización para resolver consultas del proyecto fuera de estas instancias.
Las issues tienen que venir con su label indicando la entrega y en el titulo de la issue indicar qué tipo de problema tienen dentro de los siguientes: [DUDA ENUNCIADO] o [DUDA CÓDIGO] y luego una breve descripción de su problema y en la descripción su problema en profundidad, agregando imagenes si gustan. Si hay alguna propuesta de solución, también indicarla (por ejemplo: "Podrían instalar en el servidor el editar emacs").
Existe una fecha de fin de consultas que corresponde a unos días antes del fin de cada etapa y posterior a esa fecha no se responderán consultas nuevas. Las fechas de fin de las etapas están publicadas en el programa del curso.

Objetivos de la etapa

El objetivo para esta etapa inicial del proyecto es que el estudiante aprenda y se familiarice con el lenguaje PHP, el manejo básico del sistema operativo Linux-Ubuntu mediante el procesamiento y limpieza de datos almacenados en archivos. Lo que le servirá posteriormente para seguir avanzando en el uso y procesamiento de datos usando un sistema administrador de bases de datos.
Por otro lado, en esta etapa, mediante la construcción de una aplicación simple en PHP,

deberá:

- Realizar conexiones y subir archivos al servidor del curso desde su computador de trabajo.
- Manejar aspectos básicos de Linux.
- Construir una aplicación en lenguaje PHP básico.
- Programar algoritmos para el procesamiento y limpieza de datos, no se permite procesar manualmente los datos mediante excel u otros programas.

  - Cargar mediante sftp en su cuenta del servidor Linux-Ubuntu los archivos entregados en el repositorio (scp)
  - Leer los datos desde los archivos entregados y detectar los errores usando tanto la definición de los datos como las reglas de negocio.
  - Limpiar los datos que no sean compatibles con las reglas de negocio expuestas más adelante
  - Cargar de los datos limpios en tablas implementadas como archivos csv, se usa el mismo nombre de la tabla pero con el sufijo OK, por ejemplo: usuarios -¿usuariosOK.csv
  - Además deberá entregar un README con la información requerida, estrategias de corrección utilizadas y la forma de ejecutar el programa.

1. Contexto

Usted trabaja para la empresa **Booked.com**, un comisionista de viajes y actividades recreativas que apoya a sus clientes en la exploración, selección, reserva y pago de viajes y actividades recreativas, lamentablemente la empresa se enfrenta al siguiente problema: La base de datos principal de la página web **booked.com** sufrió un evento de corrupcion que afectó gravemente la integridad de la información almacenada. Este problema se originó ya que el sistema de almacenamiento sufrió una falla por el apagón que sufrió Chile el día 25 de febrero del 2025. Como resultado, varias tablas importantes quedaron inaccesibles, lo que impidió el funcionamiento normal del sistema.

Ante esta situación, el programador Senior de la **booked** tomó acciones inmediatas para mitigar el impacto de la pérdida de datos. Utilizando copias parciales almacenadas en registros de auditoría y respaldos intermitentes, logró reconstruir una parte significativa de la información referente a usuarios y empleados. Finalmente, consolidó estos datos en dos archivos CSV, permitiendo recuperar una porción sustancial de la base de datos afectada.

Sin embargo, durante este proceso se identificó que los datos extraídos presentan inconsistencias y errores de formato. Algunos registros contienen valores incompletos, caracteres inválidos o no siguen las reglas de negocio establecidas por la empresa. Para restaurar la operatividad del sistema, es necesario realizar un proceso exhaustivo de limpieza y validación de los datos

recuperados, asegurando su integridad y conformidad con los estándares definidos.

Este proceso de limpieza incluirá la identificación y corrección de datos malformados, la eliminación[1] de registros irreparables y la modularización de los registros para que sean compatibles con la estructura de la base de datos. Una vez completada esta etapa, se podrá proceder con la reimportación[2] segura de la información al sistema, permitiendo restablecer el correcto funcionamiento de la plataforma y garantizando que no se presenten problemas futuros debido a inconsistencias en los datos.

2. Encargo para la etapa 0

Como Junior recién contratado en la empresa, te han solicitado **limpiar estos datos** por lo que haz decidido indagar sobre el uso de PHP y limpieza de datos, como también revisarás el material del curso antes de comenzar el proceso. Es por esto que primero te aseguras de comprender bien las reglas de negocio de la página web junto con la estructura de sus archivos para posteriormente embarcarte a limpiar los datos de los archivos csv entregados por el programador Senior.

3. Reglas de negocio

Como se mencionó anteriormente, a usted le entregan una serie de archivos de datos con las características indicadas en la sección de archivos de datos y tiene que procesarlos para que cumplan con las siguientes reglas de negocios.
Las reglas de negocio son la especificación o restricción que define cómo se debe realizar una operación dentro de una organización. Las reglas de negocio de la empresa son:

- El archivo relacionado a usuarios contiene información de las reservas agendadas por los usuarios. Ejemplo: El archivo relacionado a usuarios contiene información de las reservas agendadas por los usuarios. Llamado usuarios.csv al ser limpiado se llama usuariosOK.csv
- El archivo relacionado a empleados contiene información de las reservas de transporte donde un empleado es el conductor asignado.
- Las personas se identifican por correo.
- Las personas pueden ser empleado y/o usuario, siempre que estén identificados por correos distintos, por lo que pueden tener el mismo RUN.
- Las reservas están asignadas solo a un usuario.

[1] La eliminación de registros consiste en quitar el dato de la tabla original y llevarlo al archivo datos_descartados.csv

[2] Disclaimer: esta parte de reimportación no se debe realizar, es solo parte del contexto

- Para simplificar el problema asuma que los intervalos entre eventos (reservas) son instantáneos, es decir, se puede realizar un traslado inmediatamente luego de haber finalizado otro (no va a perder el avión)
- Todas las ciudades en las que trabaja la empresa se encuentran en el mismo país ficticio y se encuentran a distancias plausibles.
- Las reservas no disponibles son reservas tomadas/hechas por usuarios, mientras que las disponibles son reservas no tomadas aún.
- Todos los datos deben contar con su identificador para que el dato sea válido.
- Las fechas deben estar en formato YYYY-MM-DD.
- Los dominios de correos permitidos en la página web son 'viajes.cl', 'tourket.com', 'wass.com', 'marmol.com', 'outluc.com', 'edubus.cal' y 'viajesanma.com'.
- En los archivos existen atributos que tienen formato de lista, estos atributos deben ir sin comillas ni corchetes, separados por ";" entre ellos.

4. Archivos de datos

- `usuarios_rescatados.csv`: Datos recopilados relacionados a usuarios.
- `empleados_rescatados.csv`: Datos recopilados relacionados a empleados.
- `README.md`: Archivo con una plantilla que deben rellenar para explicar el desarrollo y ejecución de su tarea.

5. Importante: Corrección de la tarea

Para esta entrega, el carácter funcional del programa en PHP será el pilar de la corrección. No se revisarán entregas en otros lenguajes de programación. Se recomienda realizar con tiempo pruebas periódicas en el servidor.

6. 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 está permitido el uso de inteligencia artificial.

7. Formato de los datos (por archivo)

A continuación, se describirá el formato de los atributos de cada archivo entregado. Se dará la descripción de la siguiente forma: Nombre_atributo: formato, descripción, nulo/no nulo.

1. usuarios_rescatados.csv
    - Nombre: texto, es el nombre completo del usuario, admite nulo.
    - Run: Rol Único Nacional, numero natural que parte desde el 1, no nulo.
    - Dv: caracter, dígito verificador del Run, valores posibles 0 a 9, K o k, no nulo.
    - Correo: texto, correo electrónico del usuario, contiene al menos una letra, el caracter @ y un texto (dominio) Ejemplos a@tourket.com, asdfghj@wass.com, no nulo.
    - Nombre_usuario: String, nombre del usuario en la web, admite nulo.
    - Contrasena: String, contraseña del usuario, admite nulo.
    - Telefono_contacto: String, teléfono de contacto de largo 9 en formato [código de país] X XXXX XXXX, no nulo.
    - Puntos: Integer, puntuación del usuario en la aplicación, admite nulo.
    - Codigo_agenda: Integer, código único para identificar una agenda, no nulo.
    - Etiqueta: String, representa la temática que le da el usuario a su agenda de viaje en la aplicación, admite nulo.
    - Codigo_reserva: Integer, código único para identificar una reserva, no nulo.
    - Fecha: Date, fecha de reserva, admite nulo.
    - Monto: Float, precio en pesos de la reserva, no nulo.
    - Cantidad_personas: Integer, cantidad de personas de la reserva, admite nulo.
2. empleados_rescatados.csv
    - Nombre: String, es el nombre completo del usuario, admite nulo.
    - Run: Integer, Rol Único Nacional, numero natural que parte desde el 1, no nulo.
    - Dv: String, dígito verificador del Run, valores posibles 0 a 9, K o k.
    - Correo: texto, correo electrónico del usuario, contiene al menos una letra, el caracter @ y un texto (dominio) Ejemplos a@tourket.com, asdfghj@wass.com, no nulo.

- Nombre_usuario: String, nombre del usuario en la web, admite nulo.
- Contrasena: String, contraseña del usuario, no nulo.
- Telefono_contacto: String, teléfono de contacto de largo 9 en formato [codigo de país] X XXXX XXXX, no nulo.
- Jornada: String, diurno o nocturno, admite nulo.
- Isapre: String, nombre de la isapre, admite nulo.
- Contrato: String, tipo de contrato (part time o full time), admite nulo.
- Codigo_reserva: Integer, código único para identificar una reserva, no nulo.
- Codigo_agenda: Integer, código único para identificar una agenda, admite nulos.
- Fecha: Date, fecha de reserva, admite nulo.
- Monto: Float, precio en pesos de la reserva, no nulo.
- Cantidad_personas: Integer, cantidad de personas de la reserva, admite nulo.
- Estado_disponibilidad: Boolean, estado que representa si la reserva ha sido tomada, admite nulo.
- Numero_viaje: Integer, identificador del viaje, no nulo.
- Lugar_origen: String, solo admite letras (sin símbolos), admite nulo.
- Lugar_llegada: String, solo admite letras (sin símbolos), admite nulo.
- Fecha_salida: Date, fecha de comienzo del transporte, admite nulo.
- Fecha_llegada: Date, fecha de llegada del transporte, no nulo.
- Capacidad: Integer, cantidad máxima del viaje, admite nulo.
- Tiempo_estimado: Integer, tiempo en minutos del viaje, admite nulo..
- Precio_asiento: Integer, valor individual del viaje, no nulo.
- Empresa: String, empresa de transporte, admite nulo.
- Tipo_de_bus: String, normal, semi-cama o cama, admite nulos.
- Comodidades: String array, lista de comodidades del transporte, admite nulos.
- Escalas: String array, lista de escalas de un avión, admite nulos.
- Clase: String, tipo de asiento en avión (primera clase, clase ejecutiva, clase económica), admite nulos
- Paradas: String array, lista de paradas de un tren, admite nulos.

8. Archivos esperados

Con los datos que se encuentran en los archivos deberá devolver los archivos en formato CSV utilizando el lenguaje PHP

1. Personas: Datos comunes de la persona
   - Nombre
   - Run
   - Dv
   - Correo
   - Contraseña
   - Nombre usuario
   - Teléfono de contacto
2. Usuarios: Datos de usuarios
   - Nombre
   - Run
   - Dv
   - Correo
   - Contraseña
   - Nombre usuario
   - Teléfono contacto
   - Puntos
3. Empleados: Datos de empleados
   - Nombre
   - Run
   - Dv
   - Correo
   - Contraseña
   - Nombre usuario
   - Teléfono de contacto
   - Jornada

- Isapre
- Contrato

4. Agenda: Datos sobre las etiquetas de agenda
    - Correo usuario
    - Código agenda
    - Etiqueta
    - Fecha de creación
5. Reservas: Datos sobre las reservas
    - Código agenda
    - Código reserva
    - Fecha
    - Monto
    - Cantidad personas
    - Estado diponibilidad
6. Transportes: Datos sobre los transportes
    - Correo empleado
    - Código reserva
    - Número viaje
    - Lugar origen
    - Lugar llegada
    - Capacidad
    - Tiempo estimado
    - Precio asiento
    - Empresa
    - Fecha salida
    - Fecha llegada
7. Buses: Datos sobre los buses
    - Correo empleado

- Código reserva
- Numero viaje
- Lugar origen
- Lugar llegada
- Capacidad
- Tiempo estimado
- Precio asiento
- Empresa
- Tipo
- Comodidades
- Fecha salida
- Fecha llegada

8. Trenes: Datos sobre los trenes
    - Correo empleado
    - Código reserva
    - Número viaje
    - Lugar origen
    - Lugar llegada
    - Capacidad
    - Tiempo estimado
    - Precio asiento
    - Empresa
    - Comodidades
    - Paradas
    - Fecha salida
    - Fecha llegada
9. Aviones: Datos sobre los Aviones
    - Correo empleado

- Código reserva
- Número viaje
- Lugar origen
- Lugar llegada
- Capacidad
- Tiempo estimado
- Precio asiento
- Empresa
- Escalas
- Clase
- Fecha salida
- Fecha llegada

9. Formato de carpetas de la entrega

La estructura de carpetas debe seguir el siguiente formato:

```
Sites
 |-- E0
 |   |-- CSV_sucios
 |   |   |-- usuarios_rescatados.csv
 |   |   |-- empleados_rescatados.csv
 |   |
 |   |-- CSV_limpios
 |   |   |-- personasOK.csv
 |   |   |-- usuariosOK.csv
 |   |   |-- empleadosOK.csv
 |   |   |-- agendasOK.csv
 |   |   |-- reservasOK.csv
 |   |   |-- transportesOK.csv
 |   |   |-- busesOK.csv
 |   |   |-- trenesOK.csv
 |   |   |-- avionesOK.csv
 |   |   |-- datos_descartados.csv
 |   |
 |   |-- archivos
 |       |-- main.php        // Recorre los CSV y llama funciones para limpiar
 |       |-- funciones.php   // Archivo con funciones para limpiar
 |       |-- README.md       // Instrucciones y documentación
```

Es importante que sigan esta estructura, ya que la corrección la requiere. **Si no se sigue el formato, se descontará un punto**. Los únicos archivos que pueden tener un nombre distinto son los de la carpeta `archivos` (`main.php` y `funciones.php`).

Además, es fundamental que en su `README` mencionen qué archivos ejecutar para poder revisar su tarea correctamente.

10. Flujo de trabajo recomendado

- Familiarizarse con PHP, el servidor y limpieza de datos.
- Leer nuevamente el enunciado y anotar puntos clave para la limpieza de datos.
- Hacer un análisis de los datos entregados.
- Hacer un programa que pueda leer los archivos entregados.
- Crear funciones de limpieza de datos cumpliendo con las reglas de negocio y el formato de los datos.

- Probar que funcione el programa principal en conjunto con la limpieza de los datos.
- Subir los archivos al servidor.
- Probar que funcione el programa en el servidor.