

# Reto 5, Ciclo 2 - Misión TIC 2022

## Cohorte 2021

### Reporte Solución

Sebastian Camilo Puerto

C.C.: 1010017342

email: sebastian.cohnan@gmail.com

Katherine Casanova Urueña

C.C.: 66987995

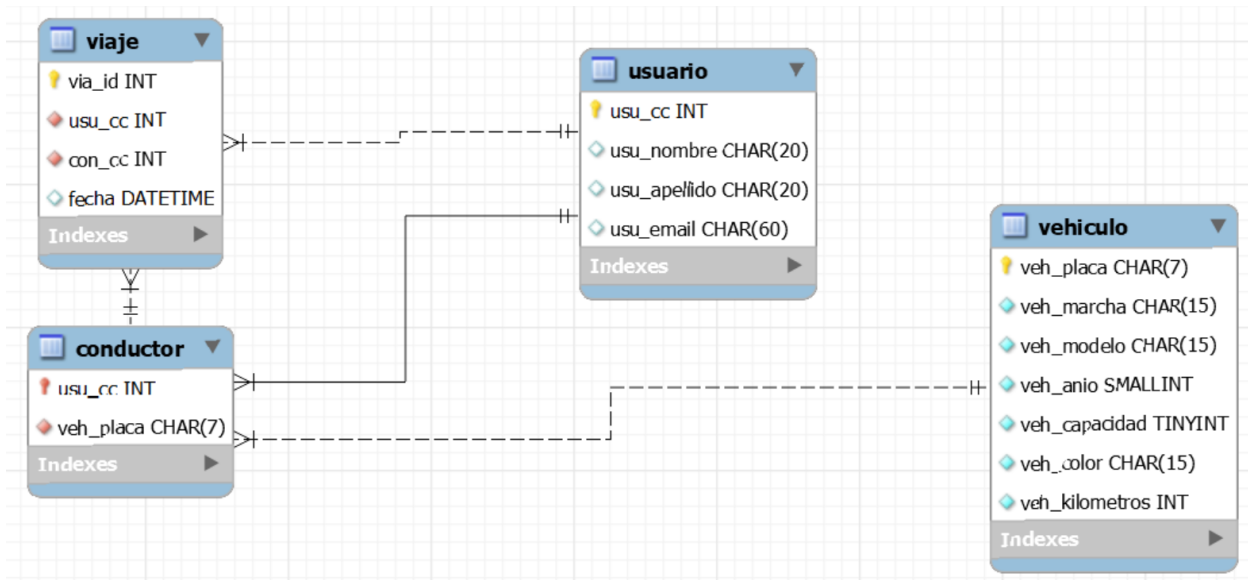
email: kecumail@gmail.com

Video:

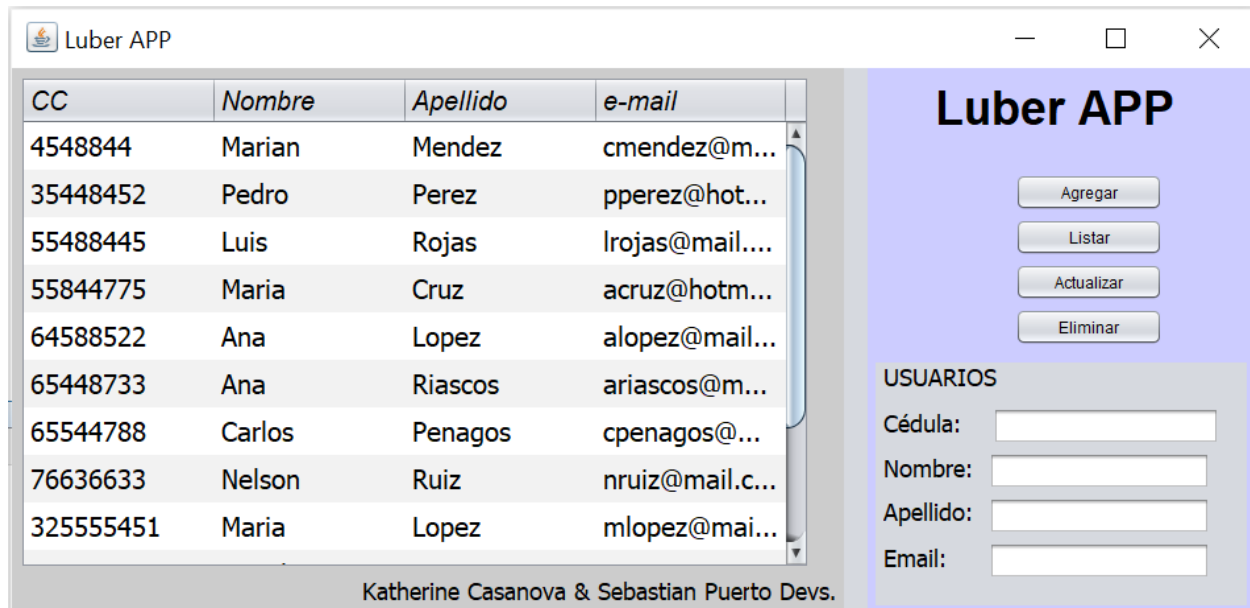
📺 Presentacion Reto 5, Ciclo 2 - MisionTIC 2022 Cohorte 202...

### Introducción

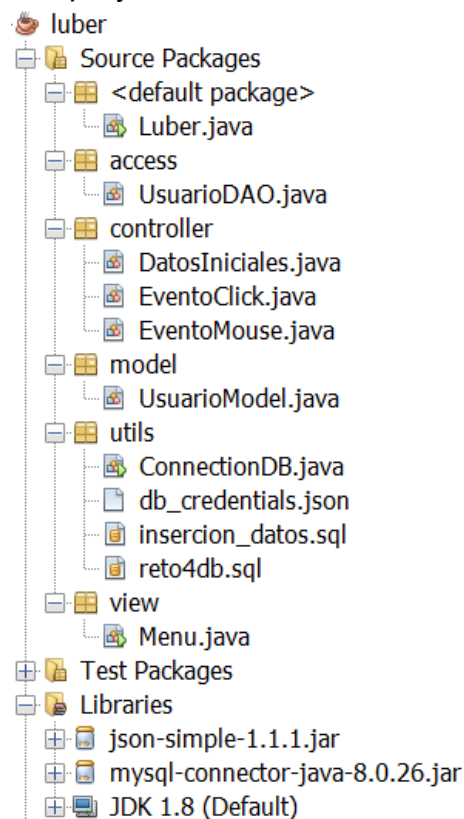
El presente proyecto es una aplicación de interacción con la base de datos de la StartUp Luber, introducida en el Reto 4 del Ciclo 2. La aplicación permite la interacción con la tabla *usuario* de la base de datos ilustrada por el siguiente diagrama EER:



La aplicación implementa los servicios CRUD (*Create, Read, Update, Delete*, o Crear, Leer, Actualizar, y Borrar en español) de la tabla *usuario* con la interfaz gráfica mostrada a continuación:



El proyecto se implementó en *Java 8* utilizando el modelo MVC (*Model-View-Controller*, o *Modelo-Vista-Controlador* en español) para separar la implementación de la interfaz gráfica, en este caso realizada con la librería *Swing*, de la implementación de la lógica del programa. La estructura del proyecto se observa a continuación:



Más detalles sobre esta estructura y sobre el funcionamiento de la aplicación se encuentran en el [video](#) enlazado al inicio del documento.

## Preparación

Para la compilación de la aplicación son necesarias las librerías *Swing* y *json-simple*. Los archivos *.jar* correspondientes se encuentran en la carpeta 'librerías' del proyecto adjunto.

Previo a la ejecución de la aplicación, es necesario primero tener un servidor de MySQL corriendo, dentro del cual haya una base de datos o esquema que contenga al menos la tabla *usuario* mencionada previamente. Adicionalmente, para poder hacer uso de esta tabla es necesario que en el archivo *db\_credentials.json* en la carpeta 'src/utills' se registren correctamente la **dirección** ip del servidor, el **puerto** de comunicación con el servidor de MySQL, el **esquema** en el que está contenida la tabla *usuario*, un **usuario** que tenga acceso de lectura y escritura sobre este esquema, junto con la **contraseña** del usuario; **este archivo debe ser modificado por el usuario para incluir credenciales adecuadas a la base de datos**. Para completitud del proyecto, en la carpeta 'src/utills' se incluyen dos scripts de MySQL: el primero, *reto4db.sql*, crea una base de datos 'reto4db' con las tablas *usuario*, *vehiculo*, *conductor* y *viaje*, las cuales generan el diagrama EER mostrado previamente; el segundo script, *insercion\_datos.sql*, inserta algunos datos dentro de estas cuatro tablas.

## Ejecución

Habiendo realizado la preparación de la sección anterior, la aplicación puede ahora ser ejecutada tomando como clase *Main* del proyecto al archivo *Luber.java* en la carpeta 'src'. Éste a su vez instanciará y ejecutará el método *Main* de la clase *Menu* en el paquete *view* del proyecto, que genera toda la interfaz gráfica del programa. Un tutorial sobre el uso adecuado de esta aplicación se encuentra en el [video](#) enlazado al inicio del documento.

## Comentarios

- Una vez se oprimen los botones de *Agregar*, *Actualizar* y *Borrar* la base de datos es actualizada de inmediato (si el query subyacente es ejecutado satisfactoriamente), pero para visualizar la base de datos actualizada resultante dentro de la aplicación es necesario oprimir el botón *Listar*. Esta es una característica del programa, que puede ser fácilmente modificada si se desea.

- Al momento de *Actualizar* un usuario, el campo de *Cédula* no debe ser cambiado por un número de cédula no presente en la tabla, pues la determinación de cuál es la fila actualizar se lleva a cabo utilizando a la cédula como clave. Una consecuencia de esto es que si se utiliza una cédula distinta a la del usuario seleccionado, pero presente en la tabla, la fila que se actualice no será la seleccionada, sino la identificada por la nueva cédula; si se utiliza una cédula inexistente, se mostrará un cuadro de diálogo informando esto.
- La aplicación no permite *Eliminar* registros que contengan campos utilizados como claves foráneas en otras tablas. Este es un mecanismo de protección de la base de datos, que preserva su integridad, y podrá ser desactivada solo por quien administre la base de datos.