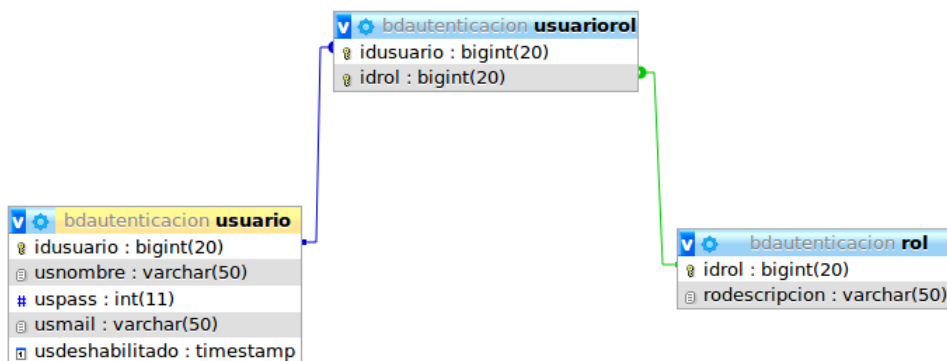


## ***Trabajo Práctico No 4 - Autenticación / Seguridad App***

La autenticación de un usuario, consiste en verificar su identidad. En las aplicaciones web comúnmente se utiliza un login con algún dato que identifique a la persona como: nombre de usuario/ dni/dirección de mail... y una clave.



*Ilustración 1:(ER) Entidad Relación - Autenticación*

1. Crear la capa de los datos, implementando el ORM (Modelo de datos) correspondiente al ER de la ilustración 1. Recordar que se debe generar al menos, una clase php por cada tabla. Cada clase debe contener las variables de instancia y sus métodos get y set; además de los métodos que nos permitan seleccionar, ingresar, modificar y eliminar los datos de cada tabla.
2. Crear la capa de control, que nos permitan acceder al ORM (Modelo de datos) y entregarle la información a las paginas de la interface.
3. Implementar dentro de la capa de Control la clase Session con los siguientes métodos:
  - `__construct()`. Constructor que. Inicia la sesión.
  - `iniciar($nombreUsuario,$psw)`. Actualiza las variables de sesión con los valores ingresados.
  - `validar()`. Valida si la sesión actual tiene usuario y psw válidos. Devuelve true o false.
  - `activa()`. Devuelve true o false si la sesión está activa o no.
  - `getUsuario()`. Devuelve el usuario logeado.
  - `getRol()`. Devuelve el rol del usuario logeado.
  - `cerrar()`. Cierra la sesión actual.
4. Implementar en la capa de la vista:
  1. un script Vista/listarUsuario.php que liste los usuario registrados y permita actualizar sus datos o realizar un borrado lógico. Las acciones que se van a poder invocar son: accion/actualizarLogin.php y accion/eliminarLogin.php
  2. un script Vista/login.php que invoque al script accion/verificarLogin.php el cual redirecciona al script Vista/paginaSegura.php si los datos ingresados se corresponden con un usuario/pass registrados. Caso contrario se redirecciona nuevamente al script Vista/login.php