**TRABAJO PRÁCTICO**

GESTIÓN DE DATOS

**GRUPO: STR\_NOMBRE\_GRUPO**

INTEGRANTES:

|  |  |
| --- | --- |
| AGUSTÍN ALABARCE |  |
| ARIEL VILA |  |
| MELANIE GUTERMAN |  |
| OSVALDO FERNÁNDEZ |  |

**Diagrama Entidad – Relación**

Para la resolución del trabajo práctico realizamos un Diagrama Entidad-Relación (DER).

A continuación se detallan las entidades creadas, con sus atributos, cuáles son sus claves primarias (PK), con qué entidades se relacionan mediante claves foráneas (FK), los tipos de datos de cada uno de los atributos y cuando es pertinente realizamos aclaraciones en base a nuestra estrategia de resolución.

1. **Usuario**

Atributos:

* Id
* Username
* Password
* Intentos\_login
* Tipo
* Baja

Aclaraciones:

* El campo password es una cadena de caracteres que siempre estará encodeada en SHA-256, es decir, al consultar la tabla en la base de datos no se podrá leer correctamente.
* Intentos\_login empieza en 0 por default. Cuando llega a 3, porque el usuario se loggeó incorrectamente 3 veces, el usuario se inhabilita.
* El campo Tipo es un char, que puede ser ‘C’ (Cliente) o ‘E’ (Empresa) en el TP.

1. **Rol**

Atributos:

* Id
* Nombre
* Baja

Aclaraciones:

* Los roles existentes son “Administrador”, “Cliente” y “Empresa”. No es muy probable que los mismos cambien, pero el enunciado del TP aclara que se podrían crear, modificar o eliminar roles, entonces decidimos tratar al rol como una entidad más.

1. **Usuario\_Rol**

Atributos:

* Usuario\_id
* Rol\_id

Aclaraciones:

* Usuario\_Rol es una entidad muchos-a-muchos. Ambos atributos son claves foráneas de usuario y de rol respectivamente. Decidimos realizarlo de esta manera porque un usuario puede tener más de un rol y viceversa, un rol puede estar asignado a muchos usuarios.

1. **Funcionalidad**

Atributos:

* Id
* Nombre

Aclaraciones:

* Cada objeto de la entidad funcionalidad es una acción que el usuario puede realizar en el sistema.

1. **Rol\_Funcionalidad**

Atributos:

* Rol\_id
* Funcionalidad\_id

Aclaraciones:

* Rol\_Funcionalidad es una entidad muchos-a-muchos. Ambos atributos son claves foráneas de rol y de funcionalidad respectivamente. Decidimos realizarlo de esta manera porque un rol puede tener (y generalmente tiene…) muchas funcionalidades, y una funcionalidad puede ser realizada por más de un rol diferente.

1. **Empresa**

Atributos:

* Id
* Usu\_id
* Razón\_social
* Mail
* Teléfono
* Calle
* Calle\_nro
* Piso
* Dpto
* Localidad
* Cod\_postal
* Ciudad
* Cuit
* Nom\_contacto
* Fecha\_creacion
* Baja

Aclaraciones:

* Una empresa siempre va a estar asociada a un usuario, ya que si un usuario se registra como empresa se crean a la vez ambas entidades, y si el administrador da de alta una nueva empresa, automáticamente se crea la entidad usuario correspondiente a la misma. Por esto, el campo usu\_id (clave foránea correspondiente a una relación uno-a-uno con usuario) es obligatorio.
* Los atributos razón social y CUIT son únicos. Igualmente decidimos crear el campo ID (autoincremental ) como clave primaria, ya que es más costoso relacionar datos string, que ocupan más lugar.
* El campo Depto no es obligatorio.
* Decidimos poner un campo “ciudad” varchar, ya que la tabla maestra no otorgaba datos de la misma, y el TP no incluía búsqueda de empresas/clientes por ciudad, ni se usaba el mismo en ningún lado, en cuyo caso sería más óptimo separar a la ciudad en otra entidad aparte, y relacionar a la empresa mediante una fk con la ciudad correspondiente.

1. **Cliente**

Atributos:

* Id
* Usu\_id
* Nombre
* Ape
* Doc
* Tipo\_Doc
* Mail
* Telefono
* Calle
* Calle\_nro
* Piso
* Depto
* Localidad
* Cod\_Postal
* Fecha\_Nac
* Baja

Aclaraciones:

* Usu\_id: idem empresa.
* El campo “Teléfono” es único.
* La combinación entre los campos Doc y Tipo\_Doc es única.
* El campo Depto no es obligatorio.

**Aclaraciones Generales:**

* Llamamos ID a la clave primaria de cada entidad, como una cuestión de nomenclatura. De acuerdo con cada entidad, la misma puede ser un número autoincremental, un código que se ingresa, o un carácter.
* El campo BAJA es de tipo bit, es decir, que puede estar en 0 o en 1. A todas las entidades que les correspondía este campo, les pusimos que por default el mismo valga 1. Cuando se da de baja lógica a la misma, pasa a valer 1.
* A los campos clave foráneas (FK) decidimos por convención llamarlo por el formato: ENTIDAD\_ID, para darnos cuenta que efectivamente se trata de una clave foránea.
* Para resolver el tema de las entidades usuario, cliente y empresa, surgieron varias opciones. La opción elegida nos pareció la más óptima, porque nos daba la posibilidad de relacionar al usuario con la empresa o cliente correspondiente, sin dejar campos null innecesarios, pudiendo acceder a los datos de las tres entidades rápidamente, utilizando únicamente un join.