**SISTEMA DE GESTIÓN DE DONANTES DE SANGRE**

**Especificación de la Base de Datos**



Versión 1.0

**Control de Versiones**



**Lima, abril del 2023**

**ÍNDICE**

[**1. Descripción de la base de datos 3**](#_qxwt3v6w2fmf)

[Breve descripción de la base de datos 3](#_jzsrjram35iw)

[Tablas que contendrá la base de datos 3](#_v9bf7je8yql9)

[**2. Diseño de la base de datos 4**](#_2npzgpqd4sl8)

[Diagrama de entidad-relación (ER) 4](#_r5a8tcxzkyh2)

[Explicación de las entidades y relaciones 4](#_ofijnpffcptk)

[**3. Especificaciones de la base de datos 6**](#_i4hdmrpms3nb)

[Descripción de las tablas 6](#_ljhjv46uofee)

[Tipos de datos y restricciones 6](#_k0y3zsc6kw4g)

[**4. Seguridad de la base de datos 8**](#_aw1gu0dj18i4)

[Descripción de las políticas de seguridad y privacidad 8](#_h3smtukc4u4x)

[**5. Limitaciones 9**](#_ou3ms5vkjr4o)

[Limitaciones y posibles mejoras a futuro 9](#_27m27l9zq843)

# 1. Descripción de la base de datos

## Breve descripción de la base de datos

La base de datos diseñada es un sistema de gestión para donar sangre que contiene información de hospitales, donantes, citas y credenciales. La base de datos está implementada en SQL y se accede a través de Python utilizando la biblioteca SQLite3. La base de datos consta de 8 tablas: hospitales, credenciales, citas,donantes,condiciones, horarios de atención y beneficios. Cada tabla tiene campos específicos que permiten el almacenamiento y gestión de la información relacionada con el banco de sangre. Los campos de las tablas están relacionados de manera que se puedan establecer conexiones entre las diferentes entidades. La base de datos permitirá el ingreso, edición y consulta de la información de manera eficiente y segura para el usuario.

## Tablas que contendrá la base de datos

La siguiente es una lista de las 8 tablas que conforman la base de datos para el sistema de donación de sangre. Cada tabla contiene información esencial para el correcto funcionamiento del sistema, desde información de hospitales y horarios de atención, hasta datos personales de donantes de sangre, así como la información de citas y donaciones. La interconexión de estas tablas a través de claves primarias y foráneas asegura la integridad y coherencia de los datos en la base de datos.

1. **Tabla [Hospital]**
2. **Tabla [Condicion]**
3. **Tabla [HorarioDeAtencion]**
4. **Tabla [Beneficio]**
5. **Tabla [Credencial]**
6. **Tabla [Donante]**
7. **Tabla [Cita]**

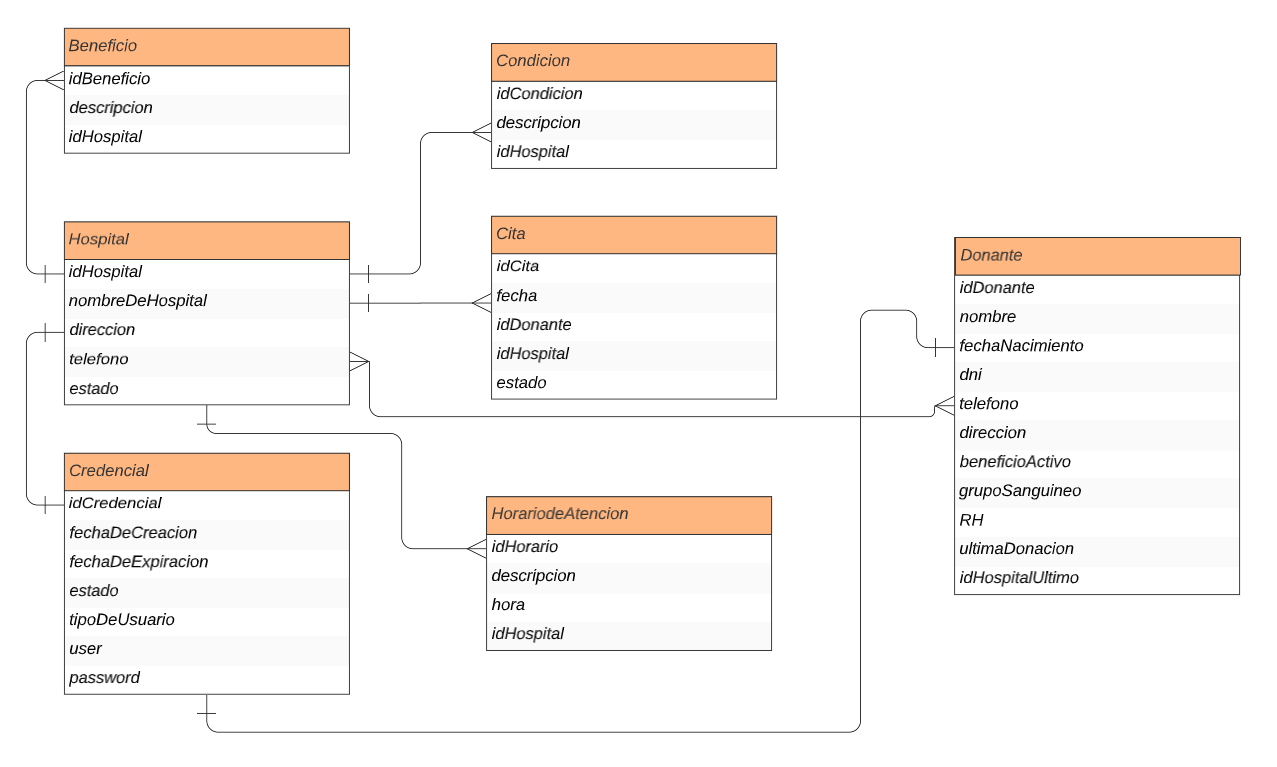
## 

# 2. Diseño de la base de datos

## Diagrama de entidad-relación (ER)

**Figura 1**

*Diagrama de entidad-relación de la base de datos del Sistema de Gestión de Donantes de Sangre*



## Explicación de las entidades y relaciones

**Relaciones**

En esta base de datos, se han definido diferentes tipos de relaciones, como la relación uno a uno. Para establecer estas relaciones, se han creado claves primarias y foráneas, que permiten vincular los registros de diferentes tablas en base a un valor en común. Estas relaciones se han representado en el diagrama entidad-relación y se han implementado en el modelo de base de datos utilizando SQL en Python.

1. La tabla "Hospital" tiene una relación uno a muchos con las tablas "Cita", “Beneficios”,”Horario” y “Condicion”. Esto significa que un hospital puede tener muchas citas, beneficios, condiciones y horarios, pero cada cita,beneficio, horario y condición solo pertenece a un hospital.
2. La tabla "Credencial" tiene una relación uno a uno con la tabla "Donante" y “Hospital”. Esto significa que un donante u hospital solo puede tener una credencial y una credencial solo pertenece a un donante u hospital.
3. La tabla "Donante" tiene una relación muchos a muchos con la tabla "Hospital". Esto significa que muchos hospitales pueden tener muchos donantes.

**Entidades**

En la base de datos, cada tabla representa una entidad, que es un objeto o concepto en el mundo real que puede ser identificado y almacenado. En este caso, las entidades incluyen hospitales, credenciales, citas, donantes, condiciones, horarios de atención y beneficios. Cada entidad tiene sus propias características únicas, que se representan como campos en la tabla correspondiente. Las tablas están conectadas mediante relaciones, lo que permite que la información se relacione y se recupere de manera eficiente a través de múltiples tablas.

1. **Hospital:** representa los hospitales en los que se pueden realizar donaciones de sangre. Contiene información como el nombre del hospital, su dirección, teléfono, estado (activo o inactivo), condiciones que deben cumplir los donantes, beneficios que reciben los donantes y horarios de atención.
2. **Credencial:** representa las credenciales de los usuarios del sistema. Contiene información como el id de la credencial, la fecha de creación y expiración, el estado (activo o inactivo), el tipo de usuario (donante) y los datos de inicio de sesión (usuario y contraseña).
3. **Cita:** representa las citas para donar sangre en un hospital. Contiene información como el id de la cita, la fecha y hora de la cita, el donante que asistirá a la cita, el hospital en el que se realizará la donación y el estado de la cita (confirmada o cancelada).
4. **Donante:** representa a los donantes de sangre. Contiene información como el grupo sanguíneo, RH, la última donación, el id del hospital en el que ha donado anteriormente, nombre, fecha de nacimiento, DNI, teléfono y dirección.
5. **Condicion:** representa las condiciones que deben cumplir los donantes para poder donar sangre en un hospital. Contiene información como una descripción y un id de la condición.
6. **HorarioDeAtencion:** representa los horarios de atención de los hospitales. Contiene información como una descripción y la hora de atención.
7. **Beneficio:** representa los beneficios que reciben los donantes al donar sangre en un hospital. Contiene información como una descripción y un id del beneficio.

# 3. Especificaciones de la base de datos

## Descripción de las tablas

* **Hospital:** contiene información sobre los hospitales registrados en el sistema, incluyendo su nombre, dirección, teléfono y estado de activación.
* **Condicion:**esta tabla almacena las condiciones de salud que los pacientes pueden tener. Cada condición está vinculada a un hospital específico.
* **HorarioDeAtencion:** aquí se almacenan los horarios de atención de cada hospital registrado en el sistema, incluyendo una breve descripción, la hora de inicio y fin, y la referencia al hospital correspondiente.
* **Beneficio:** contiene información sobre los beneficios que los hospitales pueden ofrecer a los pacientes, incluyendo una breve descripción y la referencia al hospital correspondiente.
* **Credencial:** esta tabla almacena las credenciales de acceso para cada usuario registrado en el sistema, incluyendo la fecha de creación, la fecha de expiración, el estado de activación, el tipo de usuario, y el nombre de usuario y contraseña correspondientes.
* **Donante:** contiene información específica de los donantes de sangre, incluyendo su nombre, fecha de nacimiento, DNI, teléfono, dirección, grupo sanguíneo, RH, la fecha de su última donación y el id del hospital en donde realizó su última donación.
* **Cita:** finalmente, esta tabla almacena información sobre las citas agendadas para donar sangre, incluyendo la fecha y hora, la referencia al donante correspondiente, la referencia al hospital correspondiente y el estado de activación.

## Tipos de datos y restricciones

**Tabla Hospital:**

* idHospital (int, clave primaria)
* nombreDeHospital (varchar)
* direccion (varchar)
* telefono (varchar)
* estado (bool)

**Tabla Condicion:**

* idCondicion (int, clave primaria)
* descripcion (varchar)
* idHospital (int, clave foránea referenciando a Hospital)

**Tabla HorarioDeAtencion:**

* idHorario (int, clave primaria)
* descripcion (varchar)
* hora (varchar)
* idHospital (int, clave foránea referenciando a Hospital)

**Tabla Beneficio:**

* idBeneficio (int, clave primaria)
* descripcion (varchar)
* idHospital (int, clave foránea referenciando a Hospital)

**Tabla Credencial:**

* idCredencial (int, clave primaria)
* fechaDeCreacion (date)
* fechaDeExpiracion (date)
* estado (bool)
* tipoDeUsuario (varchar)
* user (varchar)
* password (varchar)

**Tabla Donante:**

* idDonante (int, clave primaria)
* nombre (varchar)
* fechaNacimiento (date)
* dni (varchar)
* telefono (varchar)
* direccion (varchar)
* grupoSanguineo (varchar)
* RH (varchar)
* ultimaDonacion (date)
* idHospitalUltimo(int)

**Tabla Cita:**

* idCita (int, clave primaria)
* fecha (datetime)
* idDonante (int, clave foránea referenciando a Donante)
* idHospital (int, clave foránea referenciando a Hospital)
* estado (bool)

En cuanto a las restricciones, los campos que tienen una clave primaria son únicos y no pueden estar vacíos. Además, las claves foráneas hacen referencia a registros en otras tablas y, por lo tanto, deben estar relacionadas con registros válidos en esas tablas.

# 4. Seguridad de la base de datos

## Descripción de las políticas de seguridad y privacidad

La seguridad y privacidad de los datos es de suma importancia en cualquier sistema de base de datos. En este proyecto, se han implementado políticas de seguridad y privacidad que garantizan la protección de la información almacenada en la base de datos. Se utilizará un algoritmo de encriptación para las contraseñas, asegurando que solo los desarrolladores del código tengan acceso a la manipulación sensible de la base de datos.

Algunas políticas de seguridad y privacidad implementadas en este proyecto son:

* Encriptación de datos sensibles, como contraseñas o información personal.
* Restricción de acceso a la base de datos solo a usuarios autorizados y autenticados.
* Respaldos y copias de seguridad periódicas para prevenir la pérdida de información en caso de algún incidente.
* Restricción de permisos de acceso según roles y responsabilidades de los usuarios.
* Monitoreo y actualización periódica de la base de datos y su infraestructura para asegurar la integridad y disponibilidad de la información.

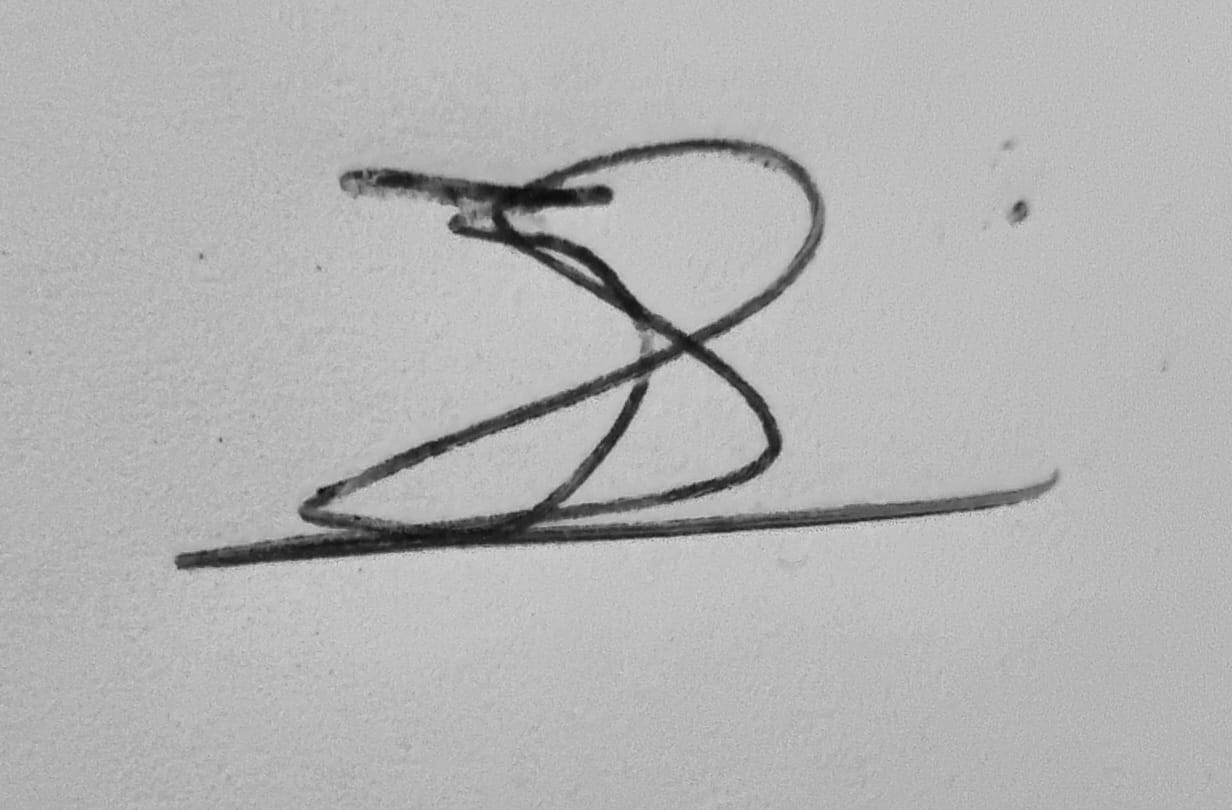
# 5. Limitaciones

## Limitaciones y posibles mejoras a futuro

En cuanto a las limitaciones de la base de datos, una restricción es el rendimiento y la escalabilidad debido a la gran cantidad de datos que se pueden llegar a almacenar. Además, la capacidad de procesamiento del servidor podría verse comprometida si se manejan múltiples solicitudes concurrentes.

Para mejorar estas limitaciones, se podrían implementar técnicas de optimización de consultas, así como una distribución más eficiente de los recursos de hardware. También se podría considerar una arquitectura de base de datos distribuida para mejorar la escalabilidad y el rendimiento.

En cuanto a las posibles mejoras a futuro, se podría implementar un sistema de copias de seguridad y recuperación en caso de fallas en el sistema. También se podrían agregar nuevas funcionalidades como la gestión de inventarios de suministros médicos y la planificación de citas para optimizar la atención médica.



—————————————

Sanchez Wong, Jatziry Fernanda

**Administradora de la Base de Datos**