

Proyecto FixApp- Base de Datos.....	2
Introducción.....	2
Tipo de Negocio.....	2
Situación Problemática.....	2
Objetivo.....	2
Diagrama Entidad-Relación (DER).....	3
Descripción de Tablas.....	4
Tabla Rol.....	4
Tabla Tipo_Equipo.....	4
Tabla Estados_Tickets.....	5
Tabla Users.....	5
Tabla Cod_Equipo.....	5
Tabla Tickets.....	6
Tabla Historial_Tickets.....	6
Justificación de la tecnología:.....	6

Proyecto FixApp- Base de Datos

Introducción

La base de datos para el proyecto FixApp está diseñada para gestionar de manera eficiente los tickets de soporte técnico en un entorno de laboratorio de informática. Este sistema permite rastrear y administrar los problemas reportados en equipos, así como mantener un historial detallado de cada ticket y las acciones realizadas por los técnicos.

Tipo de Negocio

Soporte técnico y mantenimiento de equipos de laboratorio.

Situación Problemática

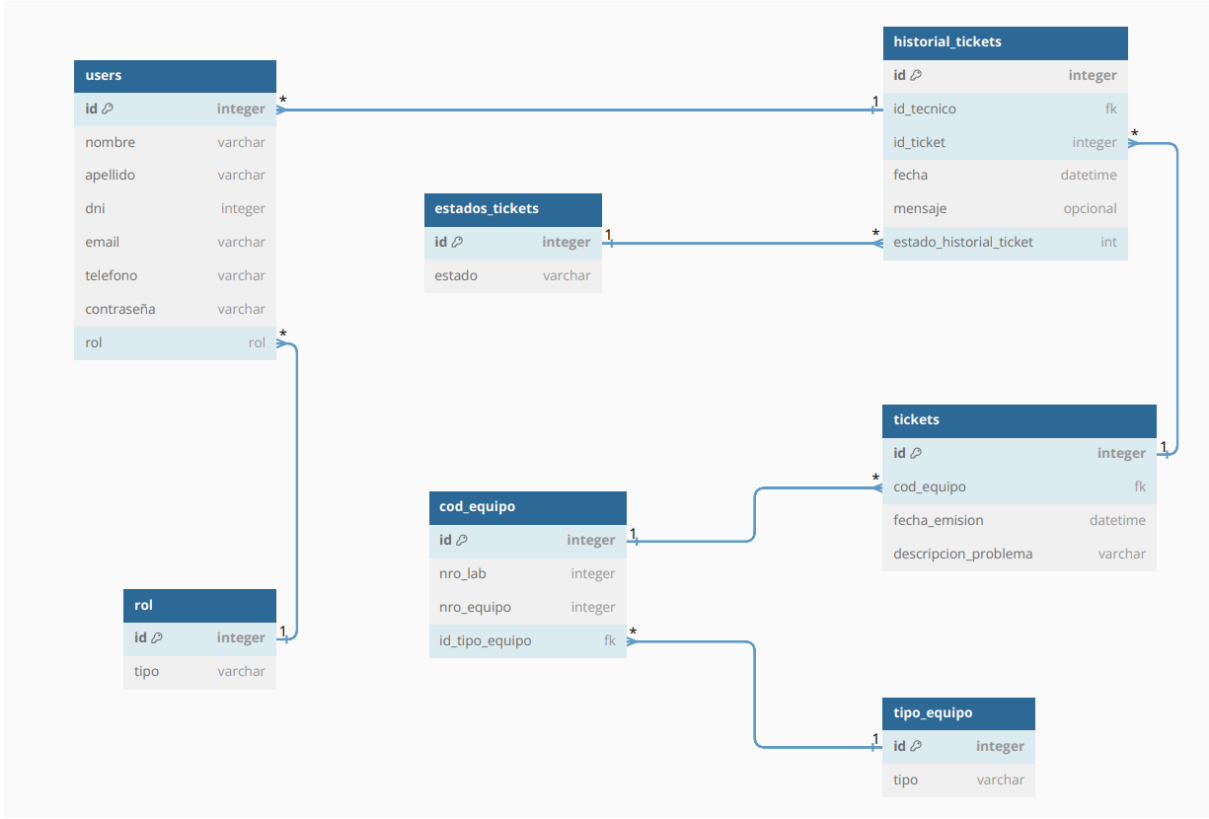
Mejorar la eficiencia en la gestión de tickets de soporte técnico, asegurando que todos los problemas reportados sean atendidos de manera oportuna y que exista un registro claro y completo de cada incidente y su resolución.

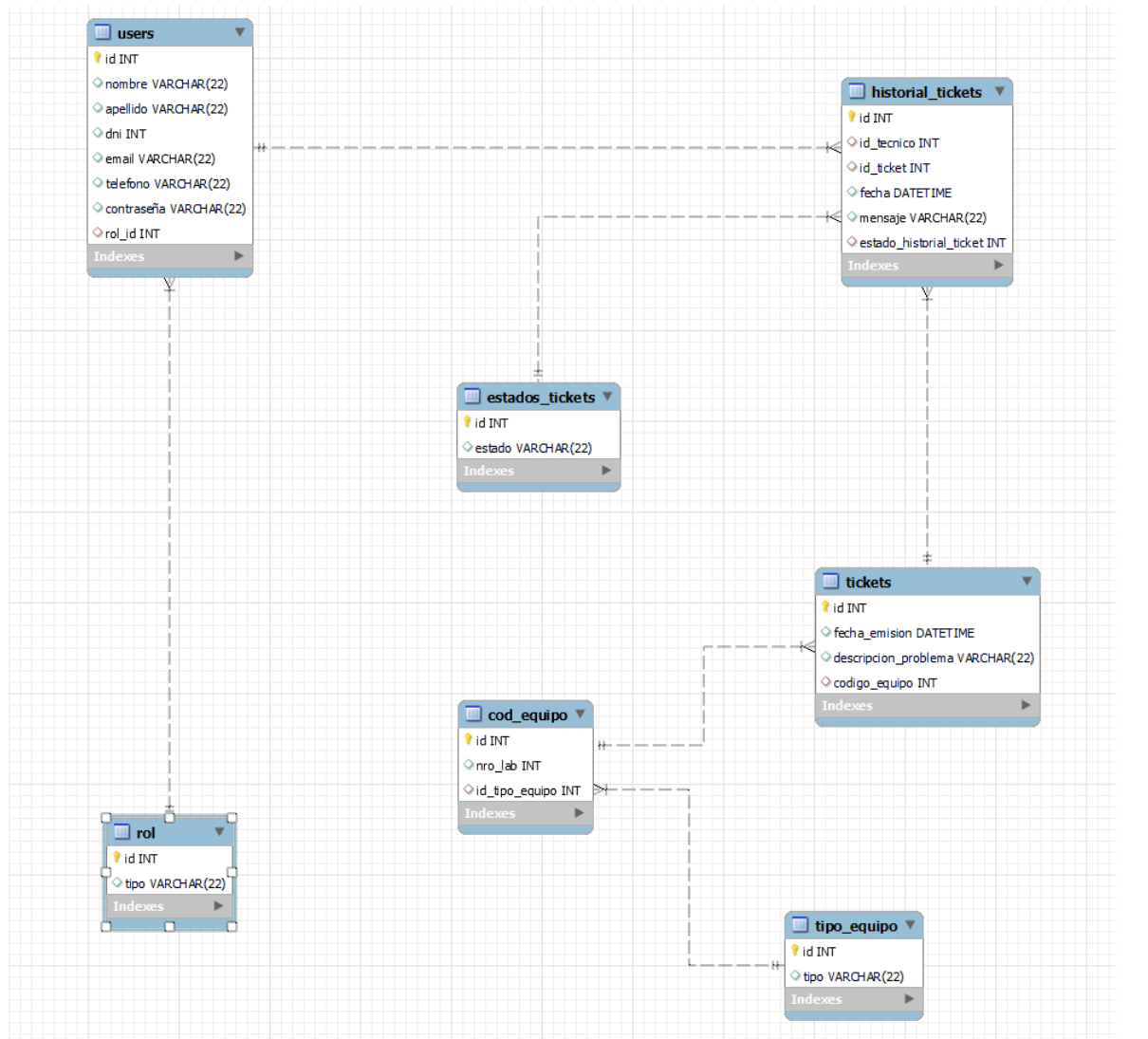
Objetivo

Crear una base de datos que permita:

1. Registrar y gestionar tickets de soporte técnico.
2. Rastrear el estado y la resolución de cada ticket.
3. Mantener un historial detallado de las acciones realizadas sobre cada ticket.
4. Facilitar la asignación de técnicos a los tickets y el seguimiento de su desempeño.

Diagrama Entidad-Relación (DER)





Descripción de Tablas

Tabla Rol

Descripción: Tabla que almacena los diferentes tipos de roles que pueden tener los usuarios en el sistema.

Campos:

- **id:** Identificador único del rol. (INT, PRIMARY KEY)
- **tipo:** Tipo de rol (VARCHAR(22)).

Tabla Tipo_Equipo

Descripción: Tabla que almacena los diferentes tipos de equipos en el laboratorio.

Campos:

- **id**: Identificador único del tipo de equipo. (INT, PRIMARY KEY, AUTO_INCREMENT)
- **tipo**: Tipo de equipo (VARCHAR(22)).

Tabla Estados_Tickets

Descripción: Tabla que almacena los diferentes estados que puede tener un ticket.

Campos:

- **id**: Identificador único del estado del ticket. (INT, PRIMARY KEY, AUTO_INCREMENT)
- **estado**: Estado del ticket (VARCHAR(22)).

Tabla Users

Descripción: Tabla que almacena la información de los usuarios del sistema.

Campos:

- **id**: Identificador único del usuario. (INT, PRIMARY KEY, AUTO_INCREMENT)
- **nombre**: Nombre del usuario. (VARCHAR(22))
- **apellido**: Apellido del usuario. (VARCHAR(22))
- **dni**: Documento Nacional de Identidad del usuario. (INT)
- **email**: Correo electrónico del usuario. (VARCHAR(22))
- **telefono**: Teléfono del usuario. (VARCHAR(22))
- **contraseña**: Contraseña del usuario. (VARCHAR(22))
- **rol_id**: Identificador del rol del usuario. (INT, FOREIGN KEY)
 - FOREIGN KEY (**rol_id**) REFERENCES **rol(id)**.

Tabla Cod_Equipo

Descripción: Tabla que almacena los códigos de los equipos en el laboratorio.

Campos:

- **id**: Identificador único del código de equipo. (INT, PRIMARY KEY, AUTO_INCREMENT)
- **nro_lab**: Número del laboratorio. (INT)
- **id_tipo_equipo**: Identificador del tipo de equipo. (INT, FOREIGN KEY)
 - FOREIGN KEY (**id_tipo_equipo**) REFERENCES **tipo_equipo(id)**.

Tabla Tickets

Descripción: Tabla que almacena los tickets de soporte técnico.

Campos:

- **id:** Identificador único del ticket. (INT, PRIMARY KEY, AUTO_INCREMENT)
- **fecha_emision:** Fecha y hora de emisión del ticket. (DATETIME)
- **descripcion_problema:** Descripción del problema reportado. (VARCHAR(22))
- **codigo_equipo:** Identificador del código de equipo asociado. (INT, FOREIGN KEY)
 - FOREIGN KEY (**codigo_equipo**) REFERENCES **cod_equipo(id)**.

Tabla Historial_Tickets

Descripción: Tabla que almacena el historial de acciones realizadas sobre cada ticket.

Campos:

- **id:** Identificador único del historial de ticket. (INT, PRIMARY KEY, AUTO_INCREMENT)
- **id_tecnico:** Identificador del técnico que realizó la acción. (INT, FOREIGN KEY)
 - FOREIGN KEY (**id_tecnico**) REFERENCES **users(id)**.
- **id_ticket:** Identificador del ticket asociado. (INT, FOREIGN KEY)
 - FOREIGN KEY (**id_ticket**) REFERENCES **tickets(id)**.
- **fecha:** Fecha y hora en que se realizó la acción. (DATETIME)
- **mensaje:** Mensaje o descripción de la acción realizada. (VARCHAR(22))
- **estado_historial_ticket:** Identificador del estado del ticket en ese momento. (INT, FOREIGN KEY)
 - FOREIGN KEY (**estado_historial_ticket**) REFERENCES **estados_tickets(id)**.

Justificación de la tecnología:

Decidimos utilizar Sql debido a que consideramos que nuestra información sería más manejable si se encontraba estructurada, además , necesitábamos tener la capacidad de relacionar mucha información de manera sencilla.

Otra de las razones de nuestra elección es el gran soporte y comunidad que tiene MySQL, siendo que esta es la base de datos más grande y usada del mundo.