Base de datos

-- Creación de la tabla Usuario

CREATE TABLE Usuario (

UserID INT PRIMARY KEY,

NombreUsuario VARCHAR(100),

Email VARCHAR(100) UNIQUE,

Contraseña VARCHAR(100),

TipoUsuario VARCHAR(20),

RolID INT,

FOREIGN KEY (RolID) REFERENCES Roles(RolID)

);

-- Creación de la tabla Roles

CREATE TABLE Roles (

RolID INT PRIMARY KEY,

NombreRol VARCHAR(50)

);

-- Creación de la tabla Proyecto

CREATE TABLE Proyecto (

ProyectoID INT PRIMARY KEY,

Nombre VARCHAR(100),

Descripcion TEXT,

FechaInicio DATE,

GerenteID INT,

FOREIGN KEY (GerenteID) REFERENCES Usuario(UserID)

);

-- Creación de la tabla AsignacionProyectoUsuario

CREATE TABLE AsignacionProyectoUsuario (

AsignacionID INT PRIMARY KEY,

UsuarioID INT,

ProyectoID INT,

RolUsuario VARCHAR(20),

FOREIGN KEY (UsuarioID) REFERENCES Usuario(UserID),

FOREIGN KEY (ProyectoID) REFERENCES Proyecto(ProyectoID)

);

-- Creación de la tabla HistoriaUsuario

CREATE TABLE HistoriaUsuario (

HistoriaID INT PRIMARY KEY,

DetallesHistoria TEXT,

CriteriosAceptacion TEXT,

Estado VARCHAR(20),

ProyectoID INT,

FOREIGN KEY (ProyectoID) REFERENCES Proyecto(ProyectoID)

);

-- Creación de la tabla Tarea

CREATE TABLE Tarea (

TareaID INT PRIMARY KEY,

DescripcionTarea TEXT,

EstadoTarea VARCHAR(20),

HistoriaID INT,

FOREIGN KEY (HistoriaID) REFERENCES HistoriaUsuario(HistoriaID)

);

-- Creación de la tabla CambiosEstadoHistoria

CREATE TABLE CambiosEstadoHistoria (

CambioHistoriaID INT PRIMARY KEY,

HistoriaID INT,

NuevoEstado INT,

UsuarioID INT,

FechaHoraCambio DATETIME,

FOREIGN KEY (HistoriaID) REFERENCES HistoriaUsuario(HistoriaID),

FOREIGN KEY (UsuarioID) REFERENCES Usuario(UserID),

FOREIGN KEY (NuevoEstado) REFERENCES Estado(EstadoID)

);

-- Creación de la tabla CambiosEstadoTarea

CREATE TABLE CambiosEstadoTarea (

CambioTareaID INT PRIMARY KEY,

TareaID INT,

NuevoEstadoTarea INT,

UsuarioID INT,

FechaHoraCambio DATETIME,

FOREIGN KEY (TareaID) REFERENCES Tarea(TareaID),

FOREIGN KEY (UsuarioID) REFERENCES Usuario(UserID),

FOREIGN KEY (NuevoEstadoTarea) REFERENCES Estado(EstadoID)

);

-- Creación de la tabla Estado

CREATE TABLE Estado (

EstadoID INT PRIMARY KEY,

NombreEstado VARCHAR(20)

);  
  
  
Detalle

1. **Tabla Usuario y Roles:**
   * La tabla **Usuario** almacena la información de los usuarios del sistema, incluyendo su nombre, email, contraseña, tipo de usuario y su rol, que está relacionado con la tabla **Roles**.
   * La tabla **Roles** contiene los diferentes roles que pueden tener los usuarios en el sistema, como gerente o desarrollador.
2. **Tabla Proyecto:**
   * La tabla **Proyecto** almacena la información sobre los proyectos, incluyendo su nombre, descripción, fecha de inicio y el ID del gerente asociado al proyecto.
   * El campo **GerenteID** en la tabla **Proyecto** está vinculado al **UserID** en la tabla **Usuario**, lo que permite identificar al gerente del proyecto.
3. **Tabla AsignacionProyectoUsuario:**
   * Esta tabla establece la relación entre los usuarios y los proyectos a los que están asignados, junto con sus roles en esos proyectos.
   * Permite asignar múltiples usuarios a un proyecto y asignar diferentes roles a cada usuario en cada proyecto.
4. **Tabla HistoriaUsuario:**
   * Almacena las historias de usuario asociadas a cada proyecto.
   * Cada historia de usuario tiene un ID único, detalles de la historia, criterios de aceptación, estado y está asociada a un proyecto específico.
5. **Tabla Tarea:**
   * Almacena las tareas asociadas a las historias de usuario.
   * Cada tarea tiene un ID único, una descripción, estado y está asociada a una historia de usuario específica.
6. **Tabla Estado:**
   * Contiene los diferentes estados que pueden tener las historias de usuario y las tareas, como "nueva", "en desarrollo" y "finalizada".
7. **Tablas CambiosEstadoHistoria y CambiosEstadoTarea:**
   * Registra los cambios de estado de las historias de usuario y las tareas.
   * Almacena quién realizó el cambio, a qué hora se realizó y el nuevo estado.
   * Estas tablas están vinculadas con las tablas **Usuario** y **Estado** para identificar al usuario que realizó el cambio y el nuevo estado respectivamente.

**Pruebas de Unidad para el Módulo de Autenticación**

**TestLogin(unittest.TestCase)**

* **setUp(self):** Método utilizado para configurar el entorno de prueba.
* **test\_login\_correct\_credentials(self):** Prueba para verificar el inicio de sesión con credenciales correctas.
  + Se envía una solicitud POST al punto final de inicio de sesión con credenciales válidas.
  + Se comprueba si la respuesta es exitosa (código de estado 200) y si contiene el mensaje esperado de inicio de sesión exitoso.
* **test\_login\_incorrect\_credentials(self):** Prueba para verificar el manejo de credenciales incorrectas.
  + Se envía una solicitud POST al punto final de inicio de sesión con credenciales inválidas.
  + Se comprueba si la respuesta indica credenciales incorrectas (código de estado 401) y si contiene el mensaje adecuado de error.