

Instrucciones de Implementación

Para implementar este programa a un dispositivo se necesitan dos aplicaciones en el sistema donde se quiera utilizar. La primera es IntelliJ, un IDE que te permite programar con Java para el back end y JavaFX para el front end.

A parte, también es necesario tener instalada la aplicación MySQL WorkBench 8.0 CE para poder crear las bases de datos.

Pasos para seguir

Una vez esten estas dos aplicaciones instaladas en el sistema, el primer paso es la creación de la base de datos. Para facilitar este apartado y no haga falta descargar e importar una base de datos por medio de Internet, este es el código requerido para la creación de la base de datos con sus tablas y referencias:

```
CREATE DATABASE gestion_proyectos;

CREATE DATABASE gestion_proyectos; CREATE DATABASE gestion_proyectos; USE
gestion_proyectos;

-- Modified usuarios table CREATE TABLE usuarios ( id_usuario INT AUTO_INCREMENT PRIMARY KEY,
nombre_usuario VARCHAR(255) NOT NULL, contrasena VARCHAR(255) NOT NULL, administrador
BOOLEAN DEFAULT FALSE, escritura BOOLEAN DEFAULT FALSE, lectura BOOLEAN DEFAULT FALSE );

-- Modified proyectos table CREATE TABLE proyectos ( id_proyecto INT AUTO_INCREMENT PRIMARY
KEY, nombre_proyecto VARCHAR(255) NOT NULL, fecha_creacion DATE, fecha_inicio DATE, fecha_fin
DATE, codigo_proyecto VARCHAR(255), palabras_clave TEXT, tipo_proyecto VARCHAR(50), activo
BOOLEAN DEFAULT FALSE, calificacion VARCHAR(255), codigo VARCHAR(255), en_cooperacion
BOOLEAN DEFAULT FALSE, bajada_calificacion BOOLEAN DEFAULT FALSE, fase_proyecto
VARCHAR(255) );

-- Unchanged usuarios_proyectos table CREATE TABLE usuarios_proyectos ( id_usuario INT,
id_proyecto INT, PRIMARY KEY (id_usuario, id_proyecto), FOREIGN KEY (id_usuario) REFERENCES
usuarios(id_usuario) ON DELETE CASCADE, FOREIGN KEY (id_proyecto) REFERENCES
proyectos(id_proyecto) ON DELETE CASCADE );

-- New documentos table CREATE TABLE documentos ( id_documento INT AUTO_INCREMENT
PRIMARY KEY, nombre VARCHAR(255) NOT NULL, id_proyecto INT, archivo LONGBLOB, FOREIGN KEY
(id_proyecto) REFERENCES proyectos(id_proyecto) ON DELETE CASCADE );

-- New auditorias table CREATE TABLE auditorias ( id_auditoria INT AUTO_INCREMENT PRIMARY KEY,
informacion VARCHAR(255) NOT NULL, accion VARCHAR(20) NOT NULL, nombre_usuario
VARCHAR(30), fecha_auditoria TIMESTAMP DEFAULT CURRENT_TIMESTAMP );

-- New logs table CREATE TABLE logs ( id_log INT AUTO_INCREMENT PRIMARY KEY, nombre_usuario
VARCHAR(30), fecha TIMESTAMP DEFAULT CURRENT_TIMESTAMP );
```

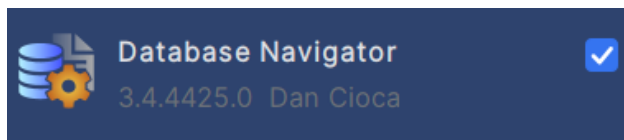
Una vez la base de datos esté funcionando, hay que importar el programa gestión de documentación en IntelliJ.

A continuación, en la parte superior izquierda de la pantalla hay que acceder a “Project Structure”, en la siguiente ventana se selecciona “Modules” y por último hay que acceder a “Dependencies”.

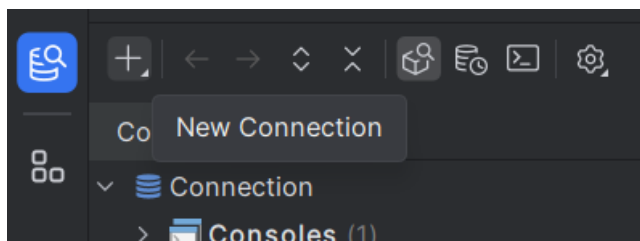
Antes de continuar hay que instalar un conector entre Java y MySQL, el nombre de este conector es: mysql-connector-j-9.0.0. Para su instalación se puede utilizar este enlace: <https://repo1.maven.org/maven2/com/mysql/mysql-connector-j/9.0.0/>

Una vez este instalado este conector se añade a el apartado de “Dependencies” presionando el botón con un símbolo de un más.

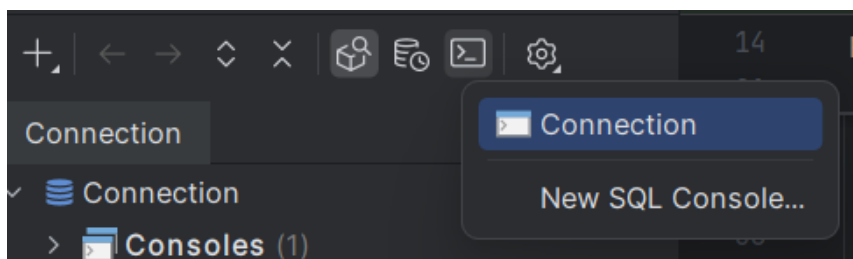
Posteriormente hay que instalar un plugin en IntelliJ llamado “Database Navigator”, como referencia esta es la apariencia del plugin en el sistema.



Una vez instalado aparecera un símbolo en la parte izquierda de IntelliJ, hay que presionar el botón de añadir conexión, seleccionar MySQL y rellenar los campos user y password.



Una vez la conexión está establecida hay que presionar el botón con apariencia de consola y elegir la opción “Connection”.



Para poder acceder a la aplicación y pasar la pantalla de inicio de sesión se necesita una cuenta existente en la base de datos, para facilitar este proceso te

proporcionamos un usuario ejemplo para poder acceder a la parte funcional e interactiva de la aplicación con acceso a todas las funciones.

```
INSERT INTO usuarios (nombre_usuario, contrasena, administrador, escritura, lectura)
VALUES
('admin', 'admin', true, true, true),
```