



**MKStudio**

**Desarrollo de un Sistema Software**

# Manual de Usuario

Miguel Rodriguez Sosa

Versión: v.1.0

Fecha: 14/05/2021

# Índice

<u>Iniciar aplicación</u>	<u>3</u>
<u>Aplicaciones necesarias</u>	<u>3</u>
<u>Preparación de Wamp</u>	<u>3</u>
<u>Preparación de IntelliJ</u>	<u>7</u>
<u>Futuro desarrollo</u>	<u>8</u>

## Iniciar aplicación

### Aplicaciones necesarias

Para usar la aplicación MKStudio, en primera instancia estaba planteado que se llevase a cabo una base de datos desarrollada con MongoDB en el Back pero por falta de tiempo, se ha desarrollado completamente su uso en el Front con una base de datos MySQL. Por lo cual, para realizar el despliegue se necesita tener instalado:

- Wamp
- IntelliJ

En un futuro se plantea el finalizar el desarrollo del Back completo y se necesitaría un contenedor en Docker por lo que lo tendríamos que tener instalado.

### Preparación de Wamp

Para poner en funcionamiento nuestra aplicación, debemos tener generada una base de datos en Wamp usando MySQL. Dejo aquí el código que se debe introducir en la consola de MySQL.

```
-- phpMyAdmin SQL Dump
-- version 3.5.1
-- http://www.phpmyadmin.net
--
-- Servidor: localhost
-- Tiempo de generación: 04-06-2021 a las 18:39:40
-- Versión del servidor: 5.5.24-log
-- Versión de PHP: 5.4.3

SET SQL_MODE="NO_AUTO_VALUE_ON_ZERO";
SET time_zone = "+00:00";

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8 */;

--
-- Base de datos: `mkstudio`
--
```

-- -----

--

-- Estructura de tabla para la tabla `espacio`

--

```
CREATE TABLE IF NOT EXISTS `espacio` (  
  `id` int(11) NOT NULL AUTO_INCREMENT,  
  `nombre` text NOT NULL,  
  `descripcion` text NOT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=7 ;
```

--

-- Volcado de datos para la tabla `espacio`

--

```
INSERT INTO `espacio` (`id`, `nombre`, `descripcion`) VALUES  
(1, 'Insonorizada', 'Sala insonorizada con cabina 2.4'),  
(2, 'Insonorizada 2.0', 'Paneles 3/4');
```

-- -----

--

-- Estructura de tabla para la tabla `material`

--

```
CREATE TABLE IF NOT EXISTS `material` (  
  `id` int(11) NOT NULL AUTO_INCREMENT,  
  `nombre` text NOT NULL,  
  `descripcion` text NOT NULL,  
  `marca` text NOT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=9 ;
```

```
--  
-- Volcado de datos para la tabla `material`  
--
```

```
INSERT INTO `material` (`id`, `nombre`, `descripcion`, `marca`) VALUES  
(1, 'Pack 1', 'Microfono, Midi', 'Focusrite'),  
(2, 'Pack 2', 'Microfono, Pie de Micro', 'Rode');
```

```
-- -----
```

```
--  
-- Estructura de tabla para la tabla `reserva`  
--
```

```
CREATE TABLE IF NOT EXISTS `reserva` (  
  `id` int(11) NOT NULL AUTO_INCREMENT,  
  `nombre` text NOT NULL,  
  `fecha` date NOT NULL,  
  `descripcion` text NOT NULL,  
  `usuario` int(11) NOT NULL,  
  `espacio` text NOT NULL,  
  `material` text NOT NULL,  
  `aprobado` tinyint(1) NOT NULL,  
  PRIMARY KEY (`id`),  
  KEY `usuario` (`usuario`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=8 ;
```

```
--  
-- Volcado de datos para la tabla `reserva`  
--
```

```
INSERT INTO `reserva` (`id`, `nombre`, `fecha`, `descripcion`, `usuario`, `espacio`,  
  `material`, `aprobado`) VALUES  
(1, 'Cancion Burrito Sabanero', '2021-06-18', 'Remix con Ozuna', 2, '4', '1', 1),  
(2, 'sad', '2021-06-06', 'dsads', 2, '1', '7', 1);
```

```

-----

--
-- Estructura de tabla para la tabla `usuario`
--

CREATE TABLE IF NOT EXISTS `usuario` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `nombre` text NOT NULL,
  `password` text NOT NULL,
  `email` text NOT NULL,
  `fecha_nacimiento` date NOT NULL,
  `telefono` int(9) NOT NULL,
  `rol` text NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=5 ;

--
-- Volcado de datos para la tabla `usuario`
--

INSERT INTO `usuario` (`id`, `nombre`, `password`, `email`, `fecha_nacimiento`, `telefono`,
`rol`) VALUES
(1, 'admin', 'admin', 'admin', '2021-06-01', 666777888, 'admin'),
(2, 'user', 'user', 'user', '2021-06-01', 666777888, 'user'),
(3, 'lolo', 'lolo', 'lolo', '2021-06-03', 123, 'user'),
(4, 'Nico', 'nico', 'nico', '2021-06-03', 123123123, 'user');

--
-- Restricciones para tablas volcadas
--

--
-- Filtros para la tabla `reserva`
--

ALTER TABLE `reserva`

```

```
ADD CONSTRAINT `reserva_ibfk_1` FOREIGN KEY (`usuario`) REFERENCES `usuario`  
(`id`) ON DELETE CASCADE ON UPDATE CASCADE;
```

```
/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;  
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;  
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
```

Después lanzamos esta sentencia en la consola de nuevo:

```
SET GLOBAL TIME_ZONE="+3:00";
```

De esta forma ya tendremos configurada la base de datos con algunos datos de ejemplo.

Lo más importante es conocer los dos usuarios clave:

- Usuario administrador("admin"):      Nombre -> admin      Password->admin
- Usuario usuario("user"):      Nombre -> user      Password->user

También existen dos usuarios más pero tienen el mismo rol que user.

## Preparación de IntelliJ

Se importa el front del proyecto y se configura con Java 8. A continuación en la pestaña de la derecha se pulsa sobre Database. Se pulsa sobre el signo + y se selecciona MySQL. Lo rellenamos como se muestra en la imagen:

The screenshot shows the 'Database' configuration window in IntelliJ IDEA. The 'Name' field is set to 'mkstudio'. The 'Comment' field is empty. The 'General' tab is selected, showing the following settings:

- Connection type: default
- Driver: MySQL
- Host: localhost
- Port: 3306
- User: root
- Password: <hidden>
- Save: Forever
- Database: mkstudio
- URL: jdbc:mysql://localhost:3306/mkstudio

Below the URL field, it says 'Overrides settings above'. At the bottom, there is a 'Test Connection' button.

Presionamos sobre “Test Connection” para comprobar que todo está correcto y si nos muestra un tick en verde, presionamos sobre “Apply” y “Ok”.

**y tras esto todo estaría listo para lanzar MKStudio.**

## **Futuro desarrollo**

A pesar de haber querido implementar una base de datos no relacional como es MongoDB en un contenedor Docker en el Back, por falta de tiempo ha sido esto imposible por lo que se ha desarrollado dicha aplicación de forma completa en el Front con una base de datos MySQL. En el GitHub también se encuentra el Back a pesar de no tener funcionalidad pero estar programado.

En el front ha faltado únicamente el SMTP para los envíos de correos al correo del administrador para controlar los Log-In de los usuarios.