

# MVC LITE 3.3

*Documentación breve*

# TABLA DE CONTENIDO

## Contenido

Manual de migración para proyectos con Core 2.x a 3x	1
--	---

## Manual de migración para proyectos con Core 2.x a 3x

## Introducción

Para migrar de la versión 2.x a la versión 3.x es necesario tomar en cuenta que, como cambio primordial se encuentra una modificación de los esquemas de administradores. Tal actualización permite a los administradores seleccionar varios departamentos sobre los cuales un usuario se puede establecer. Por ejemplo, si existen los departamentos, Administración, Informática y Recepción, se puede seleccionar a un usuario en específico para que participe en uno, dos o tres de ellos. Como resultado, cuando el usuario se haya loggeado en el sistema via /admin, podrá seleccionar con que rol estará haciendo uso del mismo.

## Manual de desarrollo general

### INTRODUCCIÓN

Esta sección contendrá a manera general la forma en cómo opera el MVC Lite, con ejemplos y tips de desarrollo. La publicación del archivo irá a la par con las publicaciones del MVC Lite.

La estructura del archivo estará determinada por el modelo de desarrollo MVC.

### COMPONENTES

#### LOG

El componente de log permite guardar desde cualquier ubicación de controlador donde se haya declarado su carga un registro en la tabla de Logs.

#### MODOS DE OPERACIÓN

Dicho componente opera de dos diferentes formas: La primera está orientada a su implementación en eventos o acontecimientos de sistema, donde el usuario es irrelevante, y el segundo donde es necesario atribuir el registro a algún operario de backend.

#### **LogComponent::write**

Parametros

- logType::string
-

## Update de versión 2.x a 3.x

### CONSULTA BÁSICA DE CONTROL DE SCHEMA

Lo primero que se debe hacer para esta actualización es correr la consulta de edición de esquema:

```
-- MySQL Workbench Synchronization
-- Generated: 2015-07-30 11:29
-- Model: Viajo México - Modelo General
-- Version: 1.0
-- Project: MVC Lite 3.x
-- Author: Daniel Lepe 2015

SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='TRADITIONAL,ALLOW_INVALID_DATES';

ALTER TABLE `administradores`
DROP FOREIGN KEY `fk_administradores_departamentos1`;

ALTER TABLE `administradores`
DROP COLUMN `departamentos_id`,
DROP PRIMARY KEY,
ADD PRIMARY KEY (`id`),
DROP INDEX `fk_administradores_departamentos1_idx` ;

CREATE TABLE IF NOT EXISTS `departamentos_has_administradores` (
  `departamentos_id` INT(11) NOT NULL,
  `administradores_id` INT(11) NOT NULL,
  PRIMARY KEY (`departamentos_id`, `administradores_id`),
  INDEX `fk_departamentos_has_administradores_administradores1_idx` (`administradores_id` ASC),
  INDEX `fk_departamentos_has_administradores_departamentos1_idx` (`departamentos_id` ASC),
  CONSTRAINT `fk_departamentos_has_administradores_departamentos1`
    FOREIGN KEY (`departamentos_id`)
    REFERENCES `departamentos` (`id`)
    ON DELETE CASCADE
    ON UPDATE CASCADE,
  CONSTRAINT `fk_departamentos_has_administradores_administradores1`
    FOREIGN KEY (`administradores_id`)
```

```
REFERENCES `administradores` (`id`)
ON DELETE CASCADE
ON UPDATE CASCADE)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8
COLLATE = utf8_general_ci;

// TABLA DE LOGS

CREATE TABLE IF NOT EXISTS `logs` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `administradores_id` int(11) DEFAULT NULL,
  `type` varchar(45) NOT NULL,
  `description` varchar(255) NOT NULL,
  `value` varchar(12) DEFAULT NULL COMMENT 'Campo para guardado arbitrario de números. Su uso es
opcional',
  `created` datetime NOT NULL,
  PRIMARY KEY (`id`),
  KEY `fk_logs_administradores1_idx` (`administradores_id`),
  CONSTRAINT `fk_logs_administradores1` FOREIGN KEY (`administradores_id`) REFERENCES
`administradores` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB AUTO_INCREMENT=9 DEFAULT CHARSET=utf8;

// DATOS NECESARIOS PARA EL UPDATE

INSERT INTO `permisos` (`clave`, `titulo`) VALUES ('logs_read', 'LOGS | Permitir acceso al módulo de
Logs');
INSERT INTO `permisos` (`clave`, `titulo`) VALUES ('logs_clean', 'LOGS | Permitir limpiar los Log');
INSERT INTO `modulos` (`padre_id`, `permisos_clave`, `titulo`, `url`, `glyphicon`, `posicion`) VALUES ('1',
'logs_read', 'Registros', 'administradores/logs', 'entypo-list', '3');

SET SQL_MODE=@OLD_SQL_MODE;
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;

// FIN DEL SQL
```

# MVC LITE 3.X

## ACTUALIZACIÓN DE PERMISOS BÁSICOS

Lo primero que se debe verificar después de haber corrido exitosamente el script anterior, es confirmar que en la tabla de “permisos” se encuentran los siguientes permisos básicos:

ID	CLAVE	TITULO
*	administradores_root	ADMIN   CRUD de administradores, departamentos y permisos
*	logs_read	LOGS   Permitir acceso al módulo de Logs
*	logs_clean	LOGS   Permitir limpiar los Log

Si el caso anterior no es satisfecho, habrá que agregarlos manualmente.

## DESCARGA DE ARCHIVOS

Si cambia los encabezados de este documento o agrega otros nuevos, puede actualizar la tabla de contenido rápidamente. Para ver los nuevos encabezados, haga clic dentro de la tabla de contenido y en Actualizar tabla.

Daniel Alberto Lepe Ayala  
Backend  
30/07/2015