



SERVICIO NACIONAL DE APRENDIZAJE SENA REGIONAL CAUCA

**SCRIPT BASES DE DATOS DEL PROYECTO.
GA6-220501096-AA2-EV03**

**PRESENTA:
JOANA ANDREA SILVA MACIAS
CC: 1125181139**

**INSTRUCTOR:
DANILO ASTUDILLO**

INFORMACIÓN TECNOLOGIA EN ANALISIS Y DESARROLLO DE SOFTWARE

BOGOTA 30/04/2025



INTRODUCCION

El presente documento detalla el script de bases de datos generado para el proyecto Prados del portal 2 desarrollado mediante MySQL Workbench, este script constituye la infraestructura clave que respalda la gestión integral de información del software.

Un script es un término informal que se refiere a un lenguaje de programación utilizado para manipular, personalizar y automatizar las funcionalidades de un sistema existente. Generalmente, los scripts se almacenan en archivos de texto y se interpretan línea por línea en tiempo real durante su ejecución, a diferencia de los programas que se compilan en archivos binarios ejecutables.

OBJETIVO

El objetivo principal del script es establecer una base de datos robusta y escalable para el Software del proyecto. Se busca suministrar una estructura que permita la gestión eficiente de diferentes entidades empresariales, desde el administrador hasta propietarios y solicitudes. Cada elemento del script cumple un papel específico en la organización y administración de datos para respaldar las sistematizaciones diarias del conjunto.



SCRIPT BASES DE DATOS DEL PROYECTO PRADOS2DB.

```
MySQL Script generated by MySQL Workbench
-- Sat May 31 10:43:56 2025
-- Model: New Model    Version: 1.0
-- MySQL Workbench Forward Engineering

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='ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DAT
E,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION';

-- -----
-- Schema prados2db
-- -----

-- -----
-- Schema prados2db
-- -----

CREATE SCHEMA IF NOT EXISTS `prados2db` DEFAULT CHARACTER SET utf8 ;
USE `prados2db` ;

-- -----
-- Table `prados2db`.`propietario`
-- -----
CREATE TABLE IF NOT EXISTS `prados2db`.`propietario` (
  `id_cedula_propietario` INT NOT NULL,
  `nombre` VARCHAR(45) NULL,
  `apellido` VARCHAR(45) NULL,
  `celular` VARCHAR(45) NULL,
  `correo` VARCHAR(45) NULL,
  `observaciones` VARCHAR(45) NULL,
  `residentes_id_cedula_residente` INT NOT NULL,
  PRIMARY KEY (`id_cedula_propietario`, `residentes_id_cedula_residente`),
  UNIQUE INDEX `id_cedula_propietario_UNIQUE` (`id_cedula_propietario` ASC)
VISIBLE,
  INDEX `fk_propietario_residentes1_idx` (`residentes_id_cedula_residente`
ASC) VISIBLE,
  CONSTRAINT `fk_propietario_residentes1`
    FOREIGN KEY (`residentes_id_cedula_residente`)
    REFERENCES `prados2db`.`residentes` (`id_cedula_residente`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
```



```
ENGINE = InnoDB;
```

```
-- -----  
-- Table `prados2db`.`funcionarios`  
-- -----  
CREATE TABLE IF NOT EXISTS `prados2db`.`funcionarios` (  
  `id_cedula` INT NOT NULL,  
  `nombres` VARCHAR(45) NULL,  
  `apellidos` VARCHAR(45) NULL,  
  `direccion` VARCHAR(45) NULL,  
  `celular` VARCHAR(45) NULL,  
  `correo` VARCHAR(45) NULL,  
  `rol` VARCHAR(45) NULL,  
  UNIQUE INDEX `id_cedula_UNIQUE` (`id_cedula` ASC) VISIBLE,  
  PRIMARY KEY (`id_cedula`))  
ENGINE = InnoDB;  
  
-- -----  
-- Table `prados2db`.`trasteos`  
-- -----  
CREATE TABLE IF NOT EXISTS `prados2db`.`trasteos` (  
  `id_trasteos` INT NOT NULL,  
  `cedula` VARCHAR(45) NULL,  
  `nombres` VARCHAR(45) NULL,  
  `apellidos` VARCHAR(45) NULL,  
  `fecha_trasteo` DATE NULL,  
  `propietario_id_cedula_propietario` INT NOT NULL,  
  `propietario_residentes_id_cedula_residente` INT NOT NULL,  
  `funcionarios_id_cedula` INT NOT NULL,  
  PRIMARY KEY (`id_trasteos`, `propietario_id_cedula_propietario`,  
  `propietario_residentes_id_cedula_residente`, `funcionarios_id_cedula`),  
  UNIQUE INDEX `id_trasteos_UNIQUE` (`id_trasteos` ASC) VISIBLE,  
  INDEX `fk_trasteos_propietario1_idx` (`propietario_id_cedula_propietario`  
  ASC, `propietario_residentes_id_cedula_residente` ASC) VISIBLE,  
  INDEX `fk_trasteos_funcionarios1_idx` (`funcionarios_id_cedula` ASC)  
  VISIBLE,  
  CONSTRAINT `fk_trasteos_propietario1`  
    FOREIGN KEY (`propietario_id_cedula_propietario` ,  
  `propietario_residentes_id_cedula_residente`)  
    REFERENCES `prados2db`.`propietario` (`id_cedula_propietario` ,  
  `residentes_id_cedula_residente`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,
```



```
CONSTRAINT `fk_trasteos_funcionarios1`  
  FOREIGN KEY (`funcionarios_id_cedula`)  
  REFERENCES `prados2db`.`funcionarios` (`id_cedula`)  
  ON DELETE NO ACTION  
  ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

```
--  
-- Table `prados2db`.`apartamento`  
--  
-----
```

```
CREATE TABLE IF NOT EXISTS `prados2db`.`apartamento` (  
  `id_apartamento` INT NOT NULL,  
  `tamaño_apartamento` VARCHAR(45) NULL,  
  `porcentaje` DECIMAL(5) NULL,  
  `propietario_id_cedula_propietario` INT NOT NULL,  
  `propietario_residentes_id_cedula_residente` INT NOT NULL,  
  `trasteos_id_trasteos` INT NOT NULL,  
  `trasteos_propietario_id_cedula_propietario` INT NOT NULL,  
  `trasteos_propietario_residentes_id_cedula_residente` INT NOT NULL,  
  `trasteos_funcionarios_id_cedula` INT NOT NULL,  
  PRIMARY KEY (`id_apartamento`, `propietario_id_cedula_propietario`,  
  `propietario_residentes_id_cedula_residente`, `trasteos_id_trasteos`,  
  `trasteos_propietario_id_cedula_propietario`,  
  `trasteos_propietario_residentes_id_cedula_residente`,  
  `trasteos_funcionarios_id_cedula`),  
  UNIQUE INDEX `id_apartamento_UNIQUE` (`id_apartamento` ASC) VISIBLE,  
  INDEX `fk_apartamento_propietario1_idx`  
  (`propietario_id_cedula_propietario` ASC,  
  `propietario_residentes_id_cedula_residente` ASC) VISIBLE,  
  INDEX `fk_apartamento_trasteos1_idx` (`trasteos_id_trasteos` ASC,  
  `trasteos_propietario_id_cedula_propietario` ASC,  
  `trasteos_propietario_residentes_id_cedula_residente` ASC,  
  `trasteos_funcionarios_id_cedula` ASC) VISIBLE,  
  CONSTRAINT `fk_apartamento_propietario1`  
    FOREIGN KEY (`propietario_id_cedula_propietario` ,  
  `propietario_residentes_id_cedula_residente`)  
    REFERENCES `prados2db`.`propietario` (`id_cedula_propietario` ,  
  `residentes_id_cedula_residente`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `fk_apartamento_trasteos1`  
    FOREIGN KEY (`trasteos_id_trasteos` ,  
  `trasteos_propietario_id_cedula_propietario` ,
```



```
`trasteos_propietario_residentes_id_cedula_residente` ,  
`trasteos_funcionarios_id_cedula`)  
    REFERENCES `prados2db`.`trasteos` (`id_trasteos` ,  
`propietario_id_cedula_propietario` ,  
`propietario_residentes_id_cedula_residente` , `funcionarios_id_cedula`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

```
-- -----  
-- Table `prados2db`.`salones sociales`  
-- -----  
CREATE TABLE IF NOT EXISTS `prados2db`.`salones sociales` (  
    `id_salones_sociales` INT NOT NULL,  
    `nombres` VARCHAR(45) NULL,  
    `apellidos` VARCHAR(45) NULL,  
    `tamaño` INT NULL,  
    `sillas` INT NULL,  
    `mesas` INT NULL,  
    `valor` INT NULL,  
    `observaciones` VARCHAR(45) NULL,  
    `propietario_id_cedula_propietario` INT NOT NULL,  
    `propietario_residentes_id_cedula_residente` INT NOT NULL,  
    PRIMARY KEY (`id_salones_sociales`, `propietario_id_cedula_propietario`,  
    `propietario_residentes_id_cedula_residente`),  
    INDEX `fk_salones sociales_propietario1_idx`  
    (`propietario_id_cedula_propietario` ASC,  
    `propietario_residentes_id_cedula_residente` ASC) VISIBLE,  
    CONSTRAINT `fk_salones sociales_propietario1`  
    FOREIGN KEY (`propietario_id_cedula_propietario` ,  
    `propietario_residentes_id_cedula_residente`)  
    REFERENCES `prados2db`.`propietario` (`id_cedula_propietario` ,  
    `residentes_id_cedula_residente`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

```
-- -----  
-- Table `prados2db`.`residentes`  
-- -----  
CREATE TABLE IF NOT EXISTS `prados2db`.`residentes` (  
    `id_cedula_residente` INT NOT NULL,  
    `nombre` VARCHAR(45) NULL,
```



```
`apellido` VARCHAR(45) NULL,  
`direccion` VARCHAR(45) NULL,  
`correo` VARCHAR(45) NULL,  
`celular` VARCHAR(45) NULL,  
`estado` TINYINT NULL,  
`apartamento_id_apartamento` INT NOT NULL,  
`salones sociales_id_salones_socilales` INT NOT NULL,  
PRIMARY KEY (`id_cedula_residente`, `apartamento_id_apartamento`, `salones  
sociales_id_salones_socilales`),  
UNIQUE INDEX `cedula_UNIQUE` (`id_cedula_residente` ASC) VISIBLE,  
INDEX `fk_residentes_apartamento1_idx` (`apartamento_id_apartamento` ASC)  
VISIBLE,  
INDEX `fk_residentes_salones sociales1_idx` (`salones  
sociales_id_salones_socilales` ASC) VISIBLE,  
CONSTRAINT `fk_residentes_apartamento1`  
  FOREIGN KEY (`apartamento_id_apartamento`)  
    REFERENCES `prados2db`.`apartamento` (`id_apartamento`)  
  ON DELETE NO ACTION  
  ON UPDATE NO ACTION,  
CONSTRAINT `fk_residentes_salones sociales1`  
  FOREIGN KEY (`salones sociales_id_salones_socilales`)  
    REFERENCES `prados2db`.`salones sociales` (`id_salones_socilales`)  
  ON DELETE NO ACTION  
  ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

```
-- -----  
-- Table `prados2db`.`parqueadero`  
-- -----  
CREATE TABLE IF NOT EXISTS `prados2db`.`parqueadero` (  
  `id_parqueadero` INT NOT NULL,  
  `tipo_parqueadero` VARCHAR(45) NULL,  
  `estado` TINYINT NULL,  
  `observaciones` VARCHAR(45) NULL,  
  `residentes_id_cedula` INT NOT NULL,  
  `apartamento_id_apartamento` INT NOT NULL,  
  `apartamento_propietario_id_cedula_propietario` INT NOT NULL,  
  `apartamento_propietario_residentes_id_cedula_residente` INT NOT NULL,  
  PRIMARY KEY (`id_parqueadero`, `residentes_id_cedula`,  
  `apartamento_id_apartamento`,  
  `apartamento_propietario_id_cedula_propietario`,  
  `apartamento_propietario_residentes_id_cedula_residente`),  
  UNIQUE INDEX `id_parqueadero_UNIQUE` (`id_parqueadero` ASC) VISIBLE,
```



```
INDEX `fk_parqueadero_residentes_idx` (`residentes_id_cedula` ASC)
VISIBLE,
INDEX `fk_parqueadero_apartamento1_idx` (`apartamento_id_apartamento` ASC,
`apartamento_propietario_id_cedula_propietario` ASC,
`apartamento_propietario_residentes_id_cedula_residente` ASC) VISIBLE,
CONSTRAINT `fk_parqueadero_residentes`
    FOREIGN KEY (`residentes_id_cedula`)
    REFERENCES `prados2db`.`residentes` (`id_cedula_residente`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
CONSTRAINT `fk_parqueadero_apartamento1`
    FOREIGN KEY (`apartamento_id_apartamento` ,
`apartamento_propietario_id_cedula_propietario` ,
`apartamento_propietario_residentes_id_cedula_residente`)
    REFERENCES `prados2db`.`apartamento` (`id_apartamento` ,
`propietario_id_cedula_propietario` ,
`propietario_residentes_id_cedula_residente`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

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




CONCLUSION

En síntesis, los softwares de prototipado desempeñan un papel fundamental en el desarrollo exitoso de un proyecto de software, actuando como pilares que sostienen el proceso y visión del esquema planteado, desde la concepción hasta la entrega final del producto. Independientemente del tipo o formato, estas herramientas son indispensables para comunicar, planificar, diseñar, implementar, probar y mantener el software de manera efectiva.



REFERENCIAS

<https://www.pixartprinting.es/blog/figma-que-es/>

<https://cei.es/que-es-figma/>

<https://ebac.mx/blog/que-es-figma>