

SISTEMA DE INFORMACIÓN PARA LA EVALUACIÓN DEL DESEMPEÑO DE
EQUINOS DE LA ESCUELA DE EQUITACIÓN DEL EJÉRCITO NACIONAL DE
COLOMBIA EN COMPETENCIAS, UTILIZANDO ANALÍTICA DE DATOS

MANUAL TÉCNICO

LIZETH JOHANNA FARFÁN MARÍN
YEIMMY LUZENA RICARDO LOZANO

UNIVERSIDAD ANTONIO NARIÑO
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA DE SISTEMAS
BOGOTÁ D.C.
2020

CONTENIDO

	Pág.
INTRODUCCIÓN.....	1
1. DESCRIPCIÓN GENERAL DEL SISTEMA	2
2. CARACTERÍSTICAS DE LOS USUARIOS DEL SISTEMA.....	3
3. REQUISITOS DEL HARDWARE Y DE SOFTWARE.....	3
4. INSTRUCCIONES DE INSTALACIÓN	3
4.1. Descargue de la aplicación AnyDesk.....	3
4.2. INSTALACION DE LA BASE DE DATOS	8
4.3. INSTALACION DE LIBRERIAS (EJECUCION DEL MODELO DE REGRESIÓN)	29
4.4. SUBIDA DEL SERVIDOR.....	30
4.5. DESINTALACIÓN DE LA APLICACIÓN	31
4.5.1. Desinstalación de Librerías	31
4.5.2. Desinstalación del Repositorio	32

TABLA DE FIGURAS

	Pág.
Figura 1. Descargue de AnyDesk	4
Figura 2. Instalación de AnyDesk	4
Figura 3. Iniciar AnyDesk.....	5
Figura 4. Conectar AnyDesk	6
Figura 5. Conectado a AnyDesk.....	7
Figura 6. Inicio aplicación.....	8
Figura 7. Inicio aplicación.....	9
Figura 8. Ejecución del script para crear base de datos	25
Figura 9. Base de datos creada	26
Figura 10. Migración Base datos	27
Figura 11. sqlmigrate Base datos.....	27
Figura 12. migrate Base datos	28
Figura 13. Creación de usuario administrador	29
Figura 14. Instalación de librerías	30
Figura 15. Subida del servidor.....	31
Figura 16. Desinstalación de librerías.....	32
Figura 17. Repositorio de la aplicación	33
Figura 18. Borrar Repositorio.....	33

INTRODUCCIÓN

En el siguiente documento se podrá evidenciar todas las características técnicas de la aplicación web desarrollada para la Escuela de Equitación del Ejército Nacional de Colombia. En este manual se describen los usuarios que la utilizan, los requisitos de software y hardware que se requieren para poder ejecutarla. Por último, los pasos a seguir para acceder a la aplicación mediante acceso remoto.

1. DESCRIPCIÓN GENERAL DEL SISTEMA

Esta aplicación web es un sistema personalizado dirigida a funcionarios de la Escuela de Equitación del Ejército Nacional, ofrece a la ESCEQ diferentes funcionalidades como son: el registro del historial clínico de cada uno de sus equinos almacenamiento de información referente a: datos del equino, exámenes clínicos, antecedentes, vacunas. Adicionalmente cuenta con un módulo que permite realizar el cargue de imágenes del equino al que se está diligenciando su hoja clínica.

Cuenta con un módulo de generación de reportes brindando información de sus equinos y los puntajes que han alcanzado de acuerdo a un plan previo de entrenamiento, mediante técnicas de aprendizaje automático (algoritmos de regresión) al momento de realizarse un registro de un plan de entrenamiento nuevo ingresando información del plan de entrenamiento, esto se relaciona con la preparación que tuvo el equino para participar en la competencia, mediante información como: evaluación nutricional, tiempo de calentamiento, horas de calentamiento, trabajo en potrero, tipo de dieta y suplementos, con cada plan específico y particular el aplicativo genera en tiempo real una predicción correspondiente al puntaje que probablemente obtendría el equino cuando participe en una competencia de salto.

Para acceder a dicha aplicación, se puede hacer desde un computador portátil o de escritorio.

2. CARACTERÍSTICAS DE LOS USUARIOS DEL SISTEMA

Para esta aplicación web el usuario que usará es:

Administrador- Veterinario: es el que se encarga de gestionar toda la información de los usuarios nuevos y existentes, carga de información para entrenamiento del modelo, registro de variables de los equinos, y generación de todos los tipos de reportes. Únicamente se cuenta con el usuario administrador con el acceso a las diferentes funcionalidades de la aplicación.

3. REQUISITOS DEL HARDWARE Y DE SOFTWARE

Los requisitos para utilizar la aplicación web son:

- Acceso a internet.
- Tener un computador, celular o Tablet en casa u oficina.
- Contar con un navegador de internet instalado, puede ser Google Chrome, Firefox, Safari, entre otros.

4. INSTRUCCIONES DE INSTALACIÓN

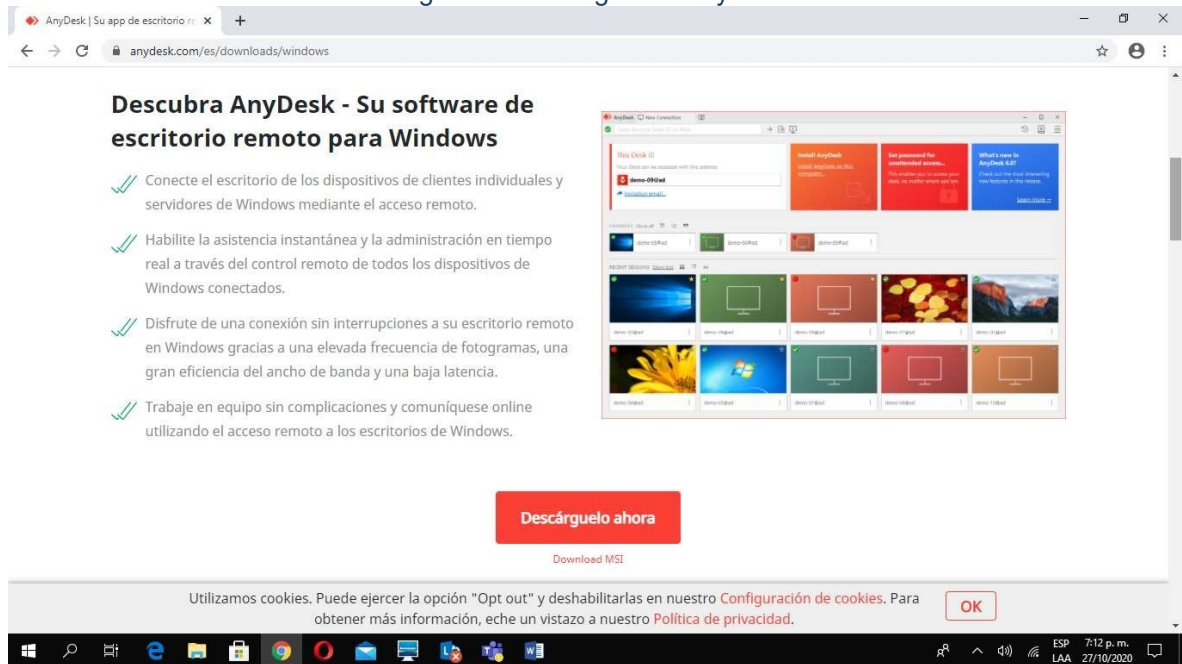
Para que la aplicación corra correctamente en la web, lo primero que se debe hacer es el despliegue de toda la parte del back-end, comenzando por la base de datos, la aplicación back-end y por último la aplicación front-end.

4.1. Descargue de la aplicación AnyDesk

Inicialmente acceder a la página oficial para la descarga de AnyDesk ingresar a la siguiente dirección: <https://anydesk.com/es/downloads/windows>.

Se procede con la descarga desde el botón “Descárguelo ahora”, como se observa en la figura 1.

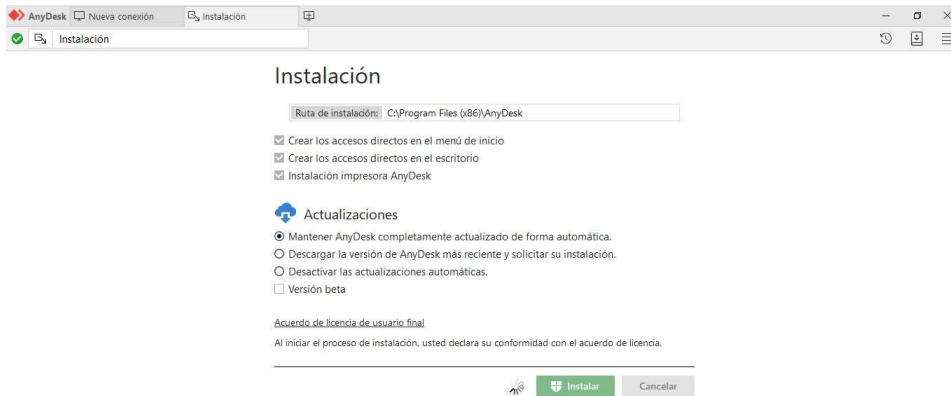
Figura 1. Descargue de AnyDesk.



Fuente: elaboración propia

Una vez descargado se debe ejecutar la aplicación AnyDesk.exe, para iniciar con la instalación. Como se observa en la figura 2, se debe dar click en **instalar**

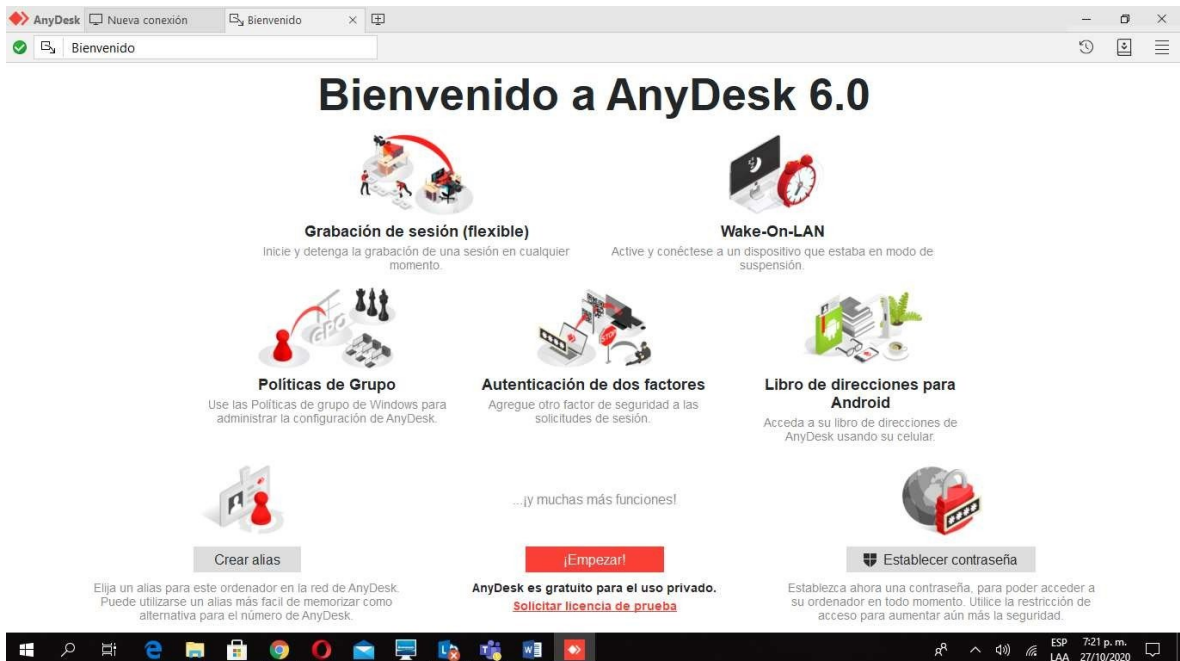
Figura 2. Instalación de AnyDesk.



Fuente: elaboración propia

Se debe iniciar AnyDesk como se observa en la imagen 3, mediante el botón “Empezar”

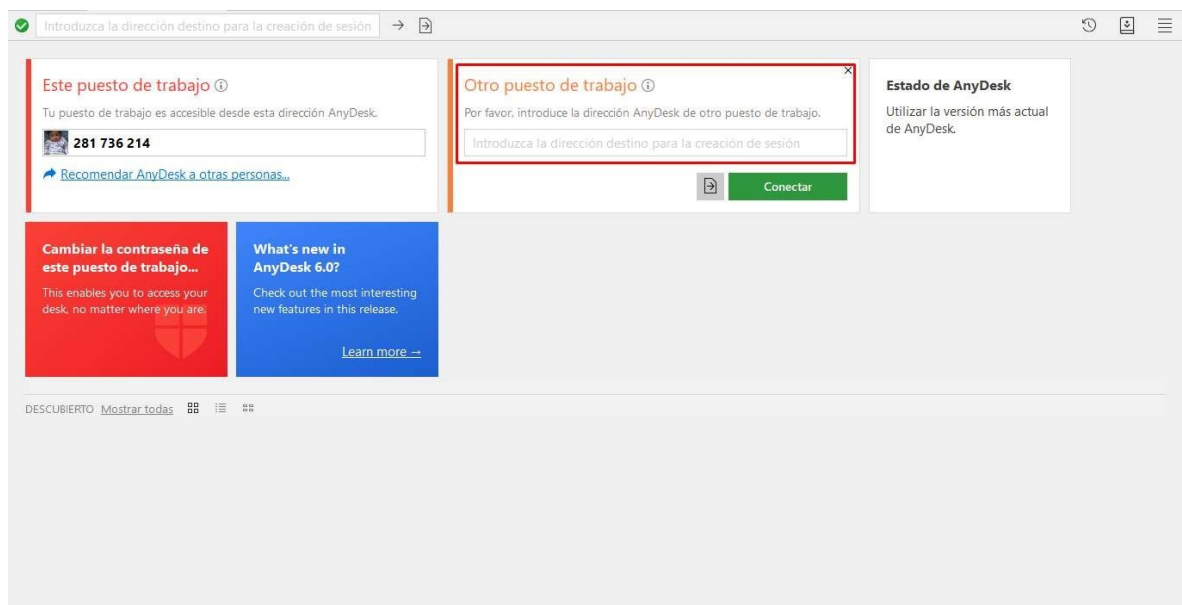
Figura 3. Iniciar AnyDesk.



Fuente: elaboración propia

Una vez hay instalado AnyDesk en su máquina se debe conectar ingresando el siguiente código **281 736 214** o **787 136 252**, y luego iniciar la conexión presionando el botón “Conectar”. Como se observa en la figura 4.

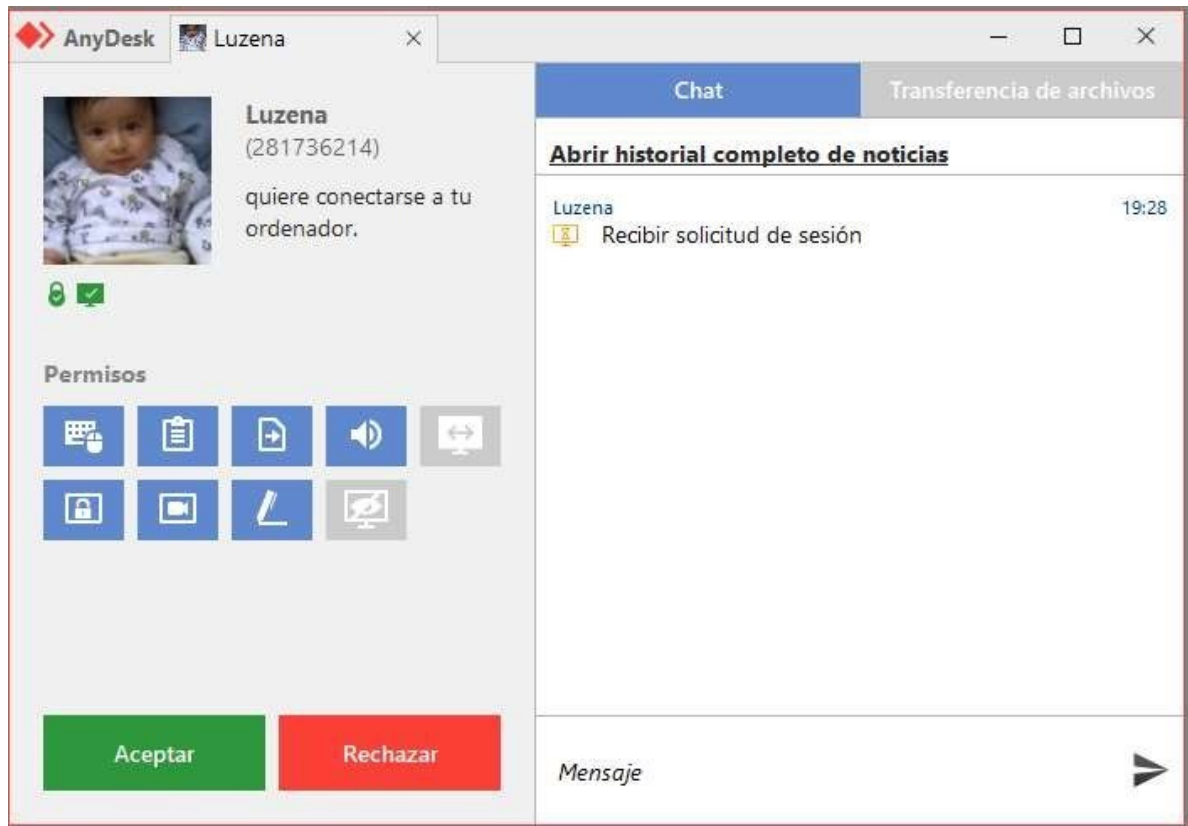
Figura 4. Conectar AnyDesk.



Fuente: elaboración propia

Una vez conectado al equipo asociado al código **281 736 214** o **787 136 252** como se observa en la figura 5. Usted ya puede acceder a la aplicación.

Figura 5. Conectado a AnyDesk.



Fuente: elaboración propia

Al ingresar encontrará el inicio a la aplicación web desarrolla para la Escuela de equitación del Ejército Nacional de Colombia. Ingresar usuario y clave indicados en el manual de usuario.

Figura 6. Inicio aplicación.



The image shows a web application login interface. At the top left is the UAN logo. Below it is the text 'Ingreso al sistema'. There are two input fields: 'Usuario' with the placeholder 'Digite su usuario' and 'Contraseña' with the placeholder 'Digite su contraseña'. Below these fields is a yellow button with the text 'INGRESAR' and a right-pointing arrow.



Fuente: elaboración propia

4.2. INSTALACION DE LA BASE DE DATOS

Se debe conectar por medio de un cliente. Para este caso, con Workbench (cliente gráfico de MySQL). Descargar el cliente siguiendo las instrucciones de la página oficial, en el siguiente enlace: <https://www.mysql.com/products/workbench/>. Luego de instalar el cliente, establecer la conexión con los datos de la base de datos como se muestra en la Figura 7. Colocar los siguientes datos:

User: root

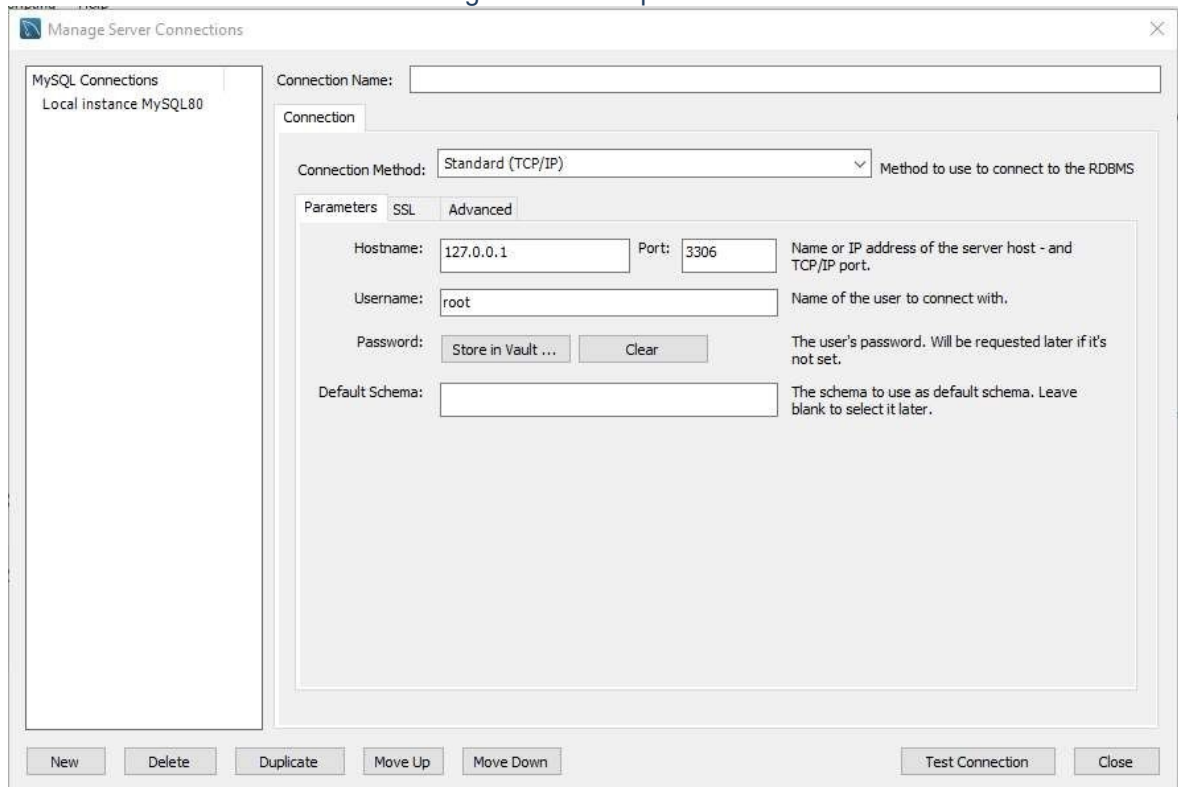
Password: hola1234

Hostname: 127.0.0.1

Puerto: 3306

Dar clic en “OK” y se hace la conexión

Figura 7. Inicio aplicación.



Fuente: elaboración propia

Luego de tener la conexión, crear la base de datos en MySQL. Acceder a la base de datos y realizar los siguientes pasos:

- Debe copiar el siguiente script de creación de los objetos de la base de datos y pegarlo en la consola de workbench. Dar clic en el ícono ejecutar, como se muestra en la Figura 8.

```
-- MySQL Script generated by MySQL Workbench
-- Wed Oct 28 10:39:00 2020
-- 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_ZE
RO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION';
```

```
-----
-- Schema mydb
-----
-----
```

```
-----
-- Schema esceq
-----
-----
```

```
-----
-- Schema esceq
-----
-----
```

```
CREATE SCHEMA IF NOT EXISTS `esceq` DEFAULT CHARACTER SET utf8mb4 COLLATE
utf8mb4_0900_ai_ci ;
USE `esceq` ;
```

```
-----
-- Table `esceq`.`auth_group`
-----
-----
```

```
CREATE TABLE IF NOT EXISTS `esceq`.`auth_group` (
`id` INT NOT NULL AUTO_INCREMENT,
`name` VARCHAR(150) NOT NULL,
PRIMARY KEY (`id`),
UNIQUE INDEX `name` (`name` ASC) VISIBLE)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;
```

```

-----
-- Table `esceq`.`django_content_type`
-----

CREATE TABLE IF NOT EXISTS `esceq`.`django_content_type` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `app_label` VARCHAR(100) NOT NULL,
  `model` VARCHAR(100) NOT NULL,
  PRIMARY KEY (`id`),
  UNIQUE INDEX `django_content_type_app_label_model_76bd3d3b_uniq` (`app_label` ASC,
  `model` ASC) VISIBLE)
ENGINE = InnoDB
AUTO_INCREMENT = 20
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;

-----

-- Table `esceq`.`auth_permission`
-----

CREATE TABLE IF NOT EXISTS `esceq`.`auth_permission` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `name` VARCHAR(255) NOT NULL,
  `content_type_id` INT NOT NULL,
  `codename` VARCHAR(100) NOT NULL,
  PRIMARY KEY (`id`),
  UNIQUE INDEX `auth_permission_content_type_id_codename_01ab375a_uniq` (`content_type_id`
  ASC, `codename` ASC) VISIBLE,
  CONSTRAINT `auth_permission_content_type_id_2f476e4b_fk_django_co`
  FOREIGN KEY (`content_type_id`)
  REFERENCES `esceq`.`django_content_type` (`id`))
ENGINE = InnoDB
AUTO_INCREMENT = 77
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;

```

```
--
-----
-- Table `esceq`.`auth_group_permissions`
-----
CREATE TABLE IF NOT EXISTS `esceq`.`auth_group_permissions` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `group_id` INT NOT NULL,
  `permission_id` INT NOT NULL,
  PRIMARY KEY (`id`),
  UNIQUE INDEX `auth_group_permissions_group_id_permission_id_0cd325b0_uniq` (`group_id`
  ASC, `permission_id` ASC) VISIBLE,
  INDEX `auth_group_permissio_permission_id_84c5c92e_fk_auth_perm` (`permission_id` ASC)
  VISIBLE,
  CONSTRAINT `auth_group_permissio_permission_id_84c5c92e_fk_auth_perm`
  FOREIGN KEY (`permission_id`)
  REFERENCES `esceq`.`auth_permission` (`id`),
  CONSTRAINT `auth_group_permissions_group_id_b120cbf9_fk_auth_group_id`
  FOREIGN KEY (`group_id`)
  REFERENCES `esceq`.`auth_group` (`id`))
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;

-----
-- Table `esceq`.`auth_user`
-----
CREATE TABLE IF NOT EXISTS `esceq`.`auth_user` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `password` VARCHAR(128) NOT NULL,
  `last_login` DATETIME(6) NULL DEFAULT NULL,
  `is_superuser` TINYINT(1) NOT NULL,
  `username` VARCHAR(150) NOT NULL,
  `first_name` VARCHAR(30) NOT NULL,
  `last_name` VARCHAR(150) NOT NULL,
```

```

`email` VARCHAR(254) NOT NULL,
`is_staff` TINYINT(1) NOT NULL,
`is_active` TINYINT(1) NOT NULL,
`date_joined` DATETIME(6) NOT NULL,
PRIMARY KEY (`id`),
UNIQUE INDEX `username` (`username` ASC) VISIBLE)
ENGINE = InnoDB
AUTO_INCREMENT = 2
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;

```

```

-----
-- Table `esceq`.`auth_user_groups`
-----

```

```

CREATE TABLE IF NOT EXISTS `esceq`.`auth_user_groups` (
`id` INT NOT NULL AUTO_INCREMENT,
`user_id` INT NOT NULL,
`group_id` INT NOT NULL,
PRIMARY KEY (`id`),
UNIQUE INDEX `auth_user_groups_user_id_group_id_94350c0c_uniq` (`user_id` ASC, `group_id`
ASC) VISIBLE,
INDEX `auth_user_groups_group_id_97559544_fk_auth_group_id` (`group_id` ASC) VISIBLE,
CONSTRAINT `auth_user_groups_group_id_97559544_fk_auth_group_id`
FOREIGN KEY (`group_id`)
REFERENCES `esceq`.`auth_group` (`id`),
CONSTRAINT `auth_user_groups_user_id_6a12ed8b_fk_auth_user_id`
FOREIGN KEY (`user_id`)
REFERENCES `esceq`.`auth_user` (`id`))
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;
-----

```


-- Table `esceq`.`auth_user_user_permissions`

```

-----
CREATE TABLE IF NOT EXISTS `esceq`.`auth_user_user_permissions` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `user_id` INT NOT NULL,
  `permission_id` INT NOT NULL,
  PRIMARY KEY (`id`),
  UNIQUE INDEX `auth_user_user_permissions_user_id_permission_id_14a6b632_uniq` (`user_id`
  ASC, `permission_id` ASC) VISIBLE,
  INDEX `auth_user_user_permi_permission_id_1fbb5f2c_fk_auth_perm` (`permission_id` ASC)
  VISIBLE,
  CONSTRAINT `auth_user_user_permi_permission_id_1fbb5f2c_fk_auth_perm`
  FOREIGN KEY (`permission_id`)
  REFERENCES `esceq`.`auth_permission` (`id`),
  CONSTRAINT `auth_user_user_permissions_user_id_a95ead1b_fk_auth_user_id`
  FOREIGN KEY (`user_id`)
  REFERENCES `esceq`.`auth_user` (`id`))
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;

```

-- Table `esceq`.`comportamiento`

```

-----
CREATE TABLE IF NOT EXISTS `esceq`.`comportamiento` (
  `id_comp` INT NOT NULL AUTO_INCREMENT,
  `nombre_com` VARCHAR(255) NOT NULL,
  `descripcion_com` VARCHAR(255) NOT NULL,
  PRIMARY KEY (`id_comp`))
ENGINE = InnoDB
AUTO_INCREMENT = 2
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;

```

-- Table `esceq`.`comps_nutricional`

CREATE TABLE IF NOT EXISTS `esceq`.`comps_nutricional` (
`id_comp` INT NOT NULL AUTO_INCREMENT,
`comp_nutricional` VARCHAR(255) NOT NULL,
`tipo_componente` VARCHAR(255) NOT NULL,
`observaciones` VARCHAR(255) NOT NULL,
PRIMARY KEY (`id_comp`))
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;

-- Table `esceq`.`disciplina_deportiva`

CREATE TABLE IF NOT EXISTS `esceq`.`disciplina_deportiva` (
`id_dis_dep` INT NOT NULL AUTO_INCREMENT,
`nombre_dis_dep` VARCHAR(255) NOT NULL,
`descripcion_dis_dep` VARCHAR(255) NOT NULL,
PRIMARY KEY (`id_dis_dep`))
ENGINE = InnoDB
AUTO_INCREMENT = 2
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;

-- Table `esceq`.`equ_disc`

CREATE TABLE IF NOT EXISTS `esceq`.`equ_disc` (
`chip_equ_disc` INT NOT NULL AUTO_INCREMENT,
`exigencia` VARCHAR(255) NOT NULL,

```

`disciplina_id_id` INT NOT NULL,
PRIMARY KEY (`chip_equ_disc`),
INDEX `equ_disc_disciplina_id_id_b0117f59_fk_disciplin` (`disciplina_id_id` ASC) VISIBLE,
CONSTRAINT `equ_disc_disciplina_id_id_b0117f59_fk_disciplin`
FOREIGN KEY (`disciplina_id_id`)
REFERENCES `esceq`.`disciplina_deportiva` (`id_dis_dep`))
ENGINE = InnoDB
AUTO_INCREMENT = 2
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;

```

```

-----
-- Table `esceq`.`raza`

```

```

-----
CREATE TABLE IF NOT EXISTS `esceq`.`raza` (
`id_raza` INT NOT NULL AUTO_INCREMENT,
`nom_raza` VARCHAR(255) NOT NULL,
`des_raza` VARCHAR(255) NOT NULL,
PRIMARY KEY (`id_raza`))
ENGINE = InnoDB
AUTO_INCREMENT = 2
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;

```

```

-----
-- Table `esceq`.`valores_selects_vario`

```

```

-----
CREATE TABLE IF NOT EXISTS `esceq`.`valores_selects_vario` (
`id` INT NOT NULL AUTO_INCREMENT,
`sexo_equino` VARCHAR(255) NOT NULL,
PRIMARY KEY (`id`))
ENGINE = InnoDB
AUTO_INCREMENT = 3

```

DEFAULT CHARACTER SET = utf8mb4

COLLATE = utf8mb4_0900_ai_ci;

insert into valores_selects_vario

values ('Macho');

insert into valores_selects_vario

values ('Hembra');

-- Table `esceq`.`registro_equino`

CREATE TABLE IF NOT EXISTS `esceq`.`registro_equino` (

`chip` INT NOT NULL,

`nom_equino` VARCHAR(255) NOT NULL,

`fec_nacimiento` DATE NOT NULL,

`adv_manejo` LONGTEXT NOT NULL,

`color` VARCHAR(255) NOT NULL,

`comportamiento_id` INT NOT NULL,

`disciplia_id` INT NOT NULL,

`raza_id` INT NOT NULL,

`sexo_id` INT NOT NULL,

PRIMARY KEY (`chip`),

INDEX `registro_equino_raza_id_ff5d2c46_fk_raza_id_raza` (`raza_id` ASC) VISIBLE,

INDEX `registro_equino_sexo_id_0cd798e0_fk_Valores_Selects_Vario_id` (`sexo_id` ASC)
VISIBLE,

INDEX `registro_equino_comportamiento_id_e3524730_fk_comportam` (`comportamiento_id`
ASC) VISIBLE,

INDEX `registro_equino_disciplia_id_2c59f7dd_fk_equ_disc_chip_equ_disc` (`disciplia_id` ASC)
VISIBLE,

CONSTRAINT `registro_equino_comportamiento_id_e3524730_fk_comportam`

FOREIGN KEY (`comportamiento_id`)

REFERENCES `esceq`.`comportamiento` (`id_comp`),

CONSTRAINT `registro_equino_disciplia_id_2c59f7dd_fk_equ_disc_chip_equ_disc`

FOREIGN KEY (`disciplia_id`)

REFERENCES `esceq`.`equ_disc` (`chip_equ_disc`),

CONSTRAINT `registro_equino_raza_id_ff5d2c46_fk_raza_id_raza`

```
FOREIGN KEY (`raza_id`)
REFERENCES `esceq`.`raza` (`id_raza`),
CONSTRAINT `registro_equino_sexo_id_0cd798e0_fk_Valores_Selects_Vario_id`
FOREIGN KEY (`sexo_id`)
REFERENCES `esceq`.`valores_selects_vario` (`id`))
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;
```

```
-----
-- Table `esceq`.`veterinario`
-----
```

```
CREATE TABLE IF NOT EXISTS `esceq`.`veterinario` (
`cedula` INT NOT NULL,
`nombres_veterinario` VARCHAR(255) NOT NULL,
`apellidos_veterinario` VARCHAR(255) NOT NULL,
`fecha_nac_veterinario` DATE NOT NULL,
`correo_veterinario` VARCHAR(254) NOT NULL,
`foto_vet` VARCHAR(100) NULL DEFAULT NULL,
PRIMARY KEY (`cedula`))
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;
```

```
-----
-- Table `esceq`.`historia_clinica`
-----
```

```
CREATE TABLE IF NOT EXISTS `esceq`.`historia_clinica` (
`id_historia` INT NOT NULL AUTO_INCREMENT,
`fecha_apertura` DATE NOT NULL,
`estado` VARCHAR(50) NOT NULL,
`observaciones` LONGTEXT NOT NULL,
`cedula_vet_id` INT NOT NULL,
```

```

`equino_chip_id` INT NOT NULL,
PRIMARY KEY (`id_historia`),
UNIQUE INDEX `equino_chip_id` (`equino_chip_id` ASC) VISIBLE,
INDEX `historia_clinica_cedula_vet_id_69852de0_fk_veterinario_cedula` (`cedula_vet_id` ASC)
VISIBLE,
CONSTRAINT `historia_clinica_cedula_vet_id_69852de0_fk_veterinario_cedula`
FOREIGN KEY (`cedula_vet_id`)
REFERENCES `esceq`.`veterinario` (`cedula`),
CONSTRAINT `historia_clinica_equino_chip_id_6cd82f92_fk_registro_equino_chip`
FOREIGN KEY (`equino_chip_id`)
REFERENCES `esceq`.`registro_equino` (`chip`))
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;

```

```

-----
-- Table `esceq`.`consulta`
-----

```

```

CREATE TABLE IF NOT EXISTS `esceq`.`consulta` (
`id_consulta` INT NOT NULL AUTO_INCREMENT,
`fecha` DATE NOT NULL,
`frec_cardiaca` DOUBLE NOT NULL,
`frec_respiratoria` DOUBLE NOT NULL,
`pulso` VARCHAR(255) NOT NULL,
`temperatura` DOUBLE NOT NULL,
`est_hidratacion` VARCHAR(255) NOT NULL,
`porc_deshidrat` DOUBLE NOT NULL,
`peso` DOUBLE NOT NULL,
`conjuntival` VARCHAR(255) NOT NULL,
`oral` VARCHAR(255) NOT NULL,
`vulva_prepucio` VARCHAR(255) NOT NULL,
`rectal` VARCHAR(255) NOT NULL,
`ojos` LONGTEXT NOT NULL,
`oidos` LONGTEXT NOT NULL,

```

```

`nod_linfaticos` LONGTEXT NOT NULL,
`chip_equino_id` INT NOT NULL,
`historia_clinica_id` INT NOT NULL,
PRIMARY KEY (`id_consulta`),
INDEX `consulta_chip_equino_id_af3b6419_fk_registro_equino_chip` (`chip_equino_id` ASC)
VISIBLE,
INDEX `consulta_historia_clinica_id_8df193ed_fk_historia_` (`historia_clinica_id` ASC) VISIBLE,
CONSTRAINT `consulta_chip_equino_id_af3b6419_fk_registro_equino_chip`
FOREIGN KEY (`chip_equino_id`)
REFERENCES `esceq`.`registro_equino` (`chip`),
CONSTRAINT `consulta_historia_clinica_id_8df193ed_fk_historia_`
FOREIGN KEY (`historia_clinica_id`)
REFERENCES `esceq`.`historia_clinica` (`id_historia`))
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;

```

```

-----
-- Table `esceq`.`dieta`
-----

```

```

CREATE TABLE IF NOT EXISTS `esceq`.`dieta` (
`id_dieta` INT NOT NULL AUTO_INCREMENT,
`nombre_dieta` VARCHAR(255) NOT NULL,
`id_fecha` DATE NOT NULL,
`estado` VARCHAR(255) NOT NULL,
`id_des_conc` INT NOT NULL,
`cant_des_conc` DOUBLE NOT NULL,
`id_alm_con` INT NOT NULL,
`cant_alm_conc` DOUBLE NOT NULL,
`id_com_conc` INT NOT NULL,
`cant_com_conc` DOUBLE NOT NULL,
`id_des_forraje` INT NOT NULL,
`cant_des_forraje` DOUBLE NOT NULL,
`id_alm_forraje` INT NOT NULL,

```

```

`cant_alm_forraje` DOUBLE NOT NULL,
`id_des_suple` INT NOT NULL,
`cant_des_suple` DOUBLE NOT NULL,
`id_alm_suple` INT NOT NULL,
`cant_alm_suple` DOUBLE NOT NULL,
`id_com_suple` INT NOT NULL,
`cant_com_suple` INT NOT NULL,
`hora_des_conc` TIME NOT NULL,
`hora_des_forraje` TIME NOT NULL,
`hora_des_suple` TIME NOT NULL,
`hora_alm_suple` TIME NOT NULL,
`hora_alm_forraje` TIME NOT NULL,
`hora_alm_conc` TIME NOT NULL,
`hora_com_suple` TIME NOT NULL,
`hora_com_forraje` TIME NOT NULL,
`chip_equino_id` INT NOT NULL,
`comps_nutricionales_id_id` INT NOT NULL,
`veterinario_cedula_id` INT NOT NULL,
PRIMARY KEY (`id_dieta`),
UNIQUE INDEX `id_fecha` (`id_fecha` ASC) VISIBLE,
UNIQUE INDEX `chip_equino_id` (`chip_equino_id` ASC) VISIBLE,
INDEX `dieta_comps_nutricionales_26f7998e_fk_comps_nut` (`comps_nutricionales_id_id` ASC)
VISIBLE,
INDEX `dieta_veterinario_cedula_id_e16047cb_fk_veterinario_cedula` (`veterinario_cedula_id`
ASC) VISIBLE,
CONSTRAINT `dieta_chip_equino_id_b78ce672_fk_registro_equino_chip`
FOREIGN KEY (`chip_equino_id`)
REFERENCES `esceq`.`registro_equino` (`chip`),
CONSTRAINT `dieta_comps_nutricionales__26f7998e_fk_comps_nut`
FOREIGN KEY (`comps_nutricionales_id_id`)
REFERENCES `esceq`.`comps_nutricional` (`id_comp`),
CONSTRAINT `dieta_veterinario_cedula_id_e16047cb_fk_veterinario_cedula`
FOREIGN KEY (`veterinario_cedula_id`)
REFERENCES `esceq`.`veterinario` (`cedula`))
ENGINE = InnoDB

```


DEFAULT CHARACTER SET = utf8mb4

COLLATE = utf8mb4_0900_ai_ci;

-- Table `esceq`.`django_admin_log`

```
CREATE TABLE IF NOT EXISTS `esceq`.`django_admin_log` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `action_time` DATETIME(6) NOT NULL,
  `object_id` LONGTEXT NULL DEFAULT NULL,
  `object_repr` VARCHAR(200) NOT NULL,
  `action_flag` SMALLINT UNSIGNED NOT NULL,
  `change_message` LONGTEXT NOT NULL,
  `content_type_id` INT NULL DEFAULT NULL,
  `user_id` INT NOT NULL,
  PRIMARY KEY (`id`),
  INDEX `django_admin_log_content_type_id_c4bce8eb_fk_django_co` (`content_type_id` ASC)
  VISIBLE,
  INDEX `django_admin_log_user_id_c564eba6_fk_auth_user_id` (`user_id` ASC) VISIBLE,
  CONSTRAINT `django_admin_log_content_type_id_c4bce8eb_fk_django_co`
  FOREIGN KEY (`content_type_id`)
  REFERENCES `esceq`.`django_content_type` (`id`),
  CONSTRAINT `django_admin_log_user_id_c564eba6_fk_auth_user_id`
  FOREIGN KEY (`user_id`)
  REFERENCES `esceq`.`auth_user` (`id`))
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;
```

-- Table `esceq`.`django_migrations`

```
CREATE TABLE IF NOT EXISTS `esceq`.`django_migrations` (
```

```

`id` INT NOT NULL AUTO_INCREMENT,
`app` VARCHAR(255) NOT NULL,
`name` VARCHAR(255) NOT NULL,
`applied` DATETIME(6) NOT NULL,
PRIMARY KEY (`id`))
ENGINE = InnoDB
AUTO_INCREMENT = 19
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;

```

```

-----
-- Table `esceq`.`django_session`
-----

```

```

CREATE TABLE IF NOT EXISTS `esceq`.`django_session` (
`session_key` VARCHAR(40) NOT NULL,
`session_data` LONGTEXT NOT NULL,
`expire_date` DATETIME(6) NOT NULL,
PRIMARY KEY (`session_key`),
INDEX `django_session_expire_date_a5c62663` (`expire_date` ASC) VISIBLE)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;

```

```

-----
-- Table `esceq`.`foto_equino`
-----

```

```

CREATE TABLE IF NOT EXISTS `esceq`.`foto_equino` (
`id` INT NOT NULL AUTO_INCREMENT,
`foto` VARCHAR(100) NULL DEFAULT NULL,
`equino_id` INT NOT NULL,
PRIMARY KEY (`id`),
INDEX `foto_equino_equino_id_afa52320_fk_registro_equino_chip` (`equino_id` ASC) VISIBLE,
CONSTRAINT `foto_equino_equino_id_afa52320_fk_registro_equino_chip`

```

```
FOREIGN KEY (`equino_id`)
REFERENCES `esceq`.`registro_equino` (`chip`))
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;
```

```
-----
-- Table `esceq`.`variables`
-----
```

```
CREATE TABLE IF NOT EXISTS `esceq`.`variables` (
`id_var` INT NOT NULL AUTO_INCREMENT,
`fecha_registro` DATE NOT NULL,
`claudicacion` INT NOT NULL,
`grado_claudi` INT NOT NULL,
`presencia_sangre` INT NOT NULL,
`olleres` INT NOT NULL,
`boca` INT NOT NULL,
`heridas_sangrantes` INT NOT NULL,
`evidencia_fat_sud` INT NOT NULL,
`grano` INT NOT NULL,
`forraje` INT NOT NULL,
`suplemento` INT NOT NULL,
`sup_oral` INT NOT NULL,
`sup_intravenoso` INT NOT NULL,
`dieta_veces_dia` INT NOT NULL,
`dieta_horarios` INT NOT NULL,
`calentamiento_previo` INT NOT NULL,
`hora_calentamiento` INT NOT NULL,
`minuto_calentamiento` INT NOT NULL,
`trabajo_cantidad_horas` INT NOT NULL,
`trabajo_cantidad_diaria` INT NOT NULL,
`trabajo_cantidad_semanal` INT NOT NULL,
`trabajo_potrero` INT NOT NULL,
`trabajo_caminador` INT NOT NULL,
```

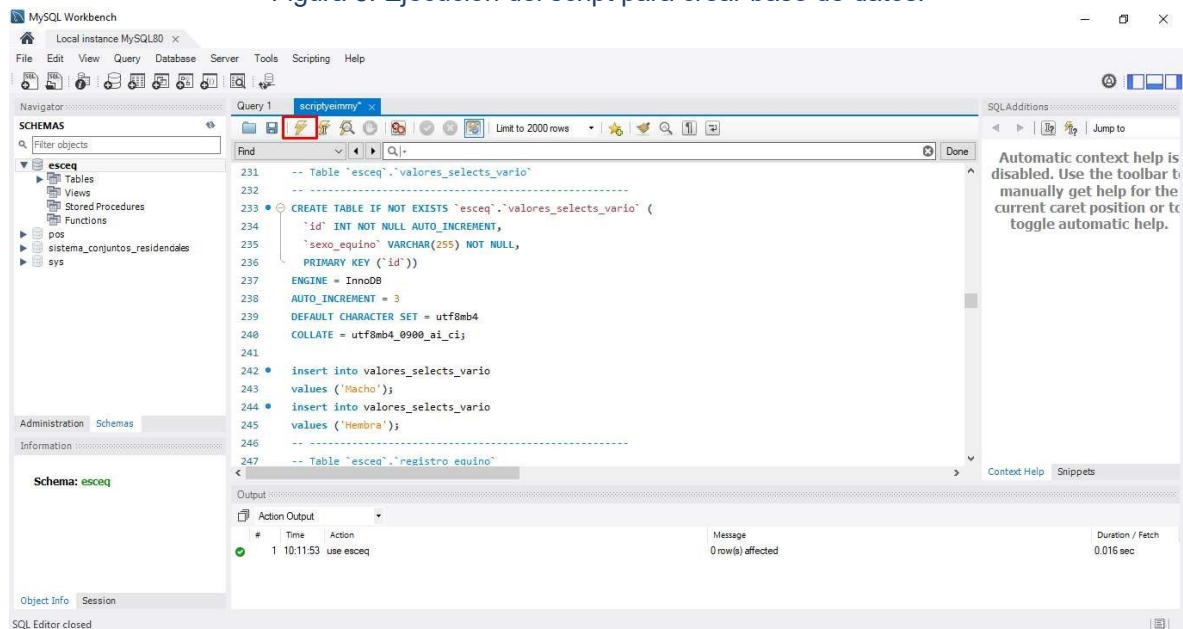
```

`tiempo_pot_cam_horas` INT NOT NULL,
`tiempo_pot_cam_mins` INT NOT NULL,
`puntaje` DOUBLE NOT NULL,
`chip_id` INT NULL DEFAULT NULL,
PRIMARY KEY (`id_var`),
INDEX `variables_chip_id_d1ba94d7_fk_registro_equino_chip` (`chip_id` ASC) VISIBLE,
CONSTRAINT `variables_chip_id_d1ba94d7_fk_registro_equino_chip`
FOREIGN KEY (`chip_id`)
REFERENCES `esceq`.`registro_equino` (`chip`))
ENGINE = InnoDB
AUTO_INCREMENT = 5
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;

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

```

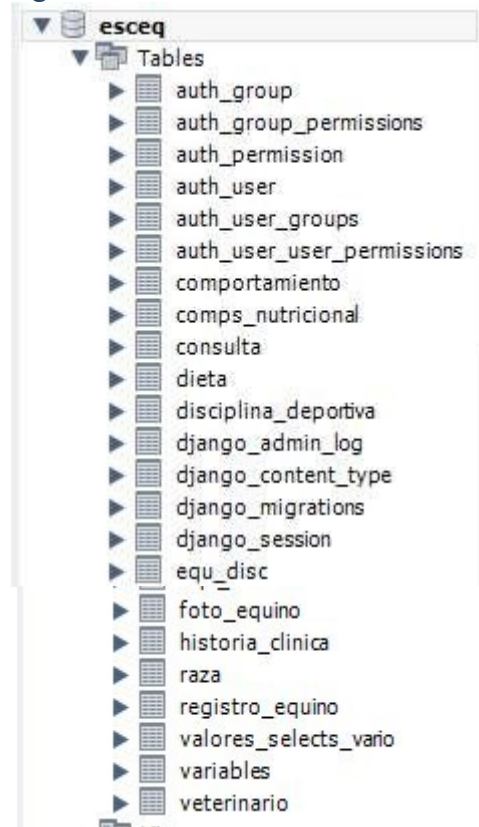
Figura 8. Ejecución del script para crear base de datos.



Fuente: elaboración propia

- Una vez ejecutado el Script, aparecen las tablas que conforman la base de datos, como se observa en la Figura 9.

Figura 9. Base de datos creada.



Fuente: elaboración propia

Otra forma de instalar la base de datos es mediante la consola de comandos de su sistema operativo. A continuación se indica el paso a paso:

Paso 1. Debe ubicarse en la carpeta donde está ubicado el proyecto. Y ejecutar el comando que permite la migración de la base de datos a *sql*: ***python manage.py makemigrations***

En la imagen 10 se ejecuta el comando.

Figura 10. Migración Base datos

```

Simbolo del sistema
Microsoft Windows [Versión 10.0.17134.1304]
(c) 2018 Microsoft Corporation. Todos los derechos reservados.

C:\Users\Luzena>cd..
C:\Users>cd..
C:\>cd C:\Users\Luzena\Repositorios\ESCEQ
C:\Users\Luzena\Repositorios\ESCEQ>python manage.py makemigrations
    
```

Fuente: elaboración propia

Luego digitar en la consola ***python manage.py sqlmigrate 0004*** (Esto permite crear las sentencias SQL de las tablas. Tener en cuenta que el 0004 va a variar según las migraciones que se hagan). Ver figura 11 para la ejecución del comando.

Figura 11. sqlmigrate Base datos

```

Simbolo del sistema
Microsoft Windows [Versión 10.0.17134.1304]
(c) 2018 Microsoft Corporation. Todos los derechos reservados.

C:\Users\Luzena>cd..
C:\Users>cd..
C:\>cd C:\Users\Luzena\Repositorios\ESCEQ
C:\Users\Luzena\Repositorios\ESCEQ>python manage.py sqlmigrate 0004_
    
```

Fuente: elaboración propia

Digitar en el CMD el comando ***python manage.py migrate*** (para terminar la migración de los cambios que se hicieron al modelo). Ver figura 12 de la ejecución del comando.

Figura 12. migrate Base datos

```

Microsoft Windows [Versión 10.0.17134.1304]
(c) 2018 Microsoft Corporation. Todos los derechos reservados.

C:\Users\Luzena>cd .
C:\Users>cd .
C:\>cd C:\Users\Luzena\Repositorios\ESCEQ
C:\Users\Luzena\Repositorios\ESCEQ>python manage.py migrate
  
```

Fuente: elaboración propia

Quando se crea la base de datos el siguiente paso es crear el usuario administrador de la aplicación. Ejecutar en la consola el comando: **python manage.py createsuperuser**

Luego pide cambiar por el nombre de usuario, por defecto le aparecerá el que tiene en su equipo cambiar al usuario: **admin**

Quando solicite la clave ingresar la clave: **hola1234**

En la figura 13 se realiza la creación del usuario administrador quien tendrá acceso total a la aplicación.

Figura 13. Creación de usuario administrador

```

Microsoft Windows [Versión 10.0.17134.1304]
(c) 2018 Microsoft Corporation. Todos los derechos reservados.

C:\Users\Luzena>cd..
C:\Users>cd..
C:\>cd C:\Users\Luzena\Repositorios\ESCEQ
C:\Users\Luzena\Repositorios\ESCEQ>python manage.py createsuperuser
System check identified some issues:

WARNINGS:
registro.Dieta.chip_equino: (fields.W342) Setting unique=True on a ForeignKey has the same effect as using a OneToOneField.
HINT: ForeignKey(unique=True) is usually better served by a OneToOneField.
registro.Historia_Clinica.equino_chip: (fields.W342) Setting unique=True on a ForeignKey has the same effect as using a OneToOneField.
HINT: ForeignKey(unique=True) is usually better served by a OneToOneField.
Nombre de usuario (leave blank to use 'luzena'): admin
Error: That nombre de usuario is already taken.
Nombre de usuario (leave blank to use 'luzena'): admin1
Dirección de correo electrónico: yricardo@uan.edu.co
Password:
Password (again):
Esta contraseña es demasiado común.
Bypass password validation and create user anyway? [y/N]: y
Superuser created successfully.

C:\Users\Luzena\Repositorios\ESCEQ>
  
```

Fuente: elaboración propia

4.3. INSTALACION DE LIBRERIAS (EJECUCION DEL MODELO DE REGRESIÓN)

Esta aplicación cuenta con un módulo en que se empleó un modelo de regresión y es necesario para su correcto funcionamiento instalar las siguientes librerías desde la consola de comandos (cmd).

Las librerías a instalar son las siguientes:

- *pip install django_filters*
- *pip install django-excel*
- *pip install pandas*
- *pip install matplotlib*
- *pip install seaborn*
- *pip install sklearn*

- *pip install django-import-export*

Como se observa en la figura 14. Se debe copiar cada una de las librerías en la consola de comandos para iniciar con la instalación.

Figura 14. Instalación de librerías.

```

Simbolo del sistema - pip install pandas
c:\Users\Luzena> pip install pandas
Requirement already satisfied: pandas in c:\users\luzena\appdata\local\programs\python\python38-32\lib\site-packages (1.1.0)
Requirement already satisfied: pytz>=2017.2 in c:\users\luzena\appdata\local\programs\python\python38-32\lib\site-packages (from pandas) (2017.2)
Requirement already satisfied: numpy>=1.15.4 in c:\users\luzena\appdata\local\programs\python\python38-32\lib\site-packages (from pandas) (1.19.1)
Requirement already satisfied: python-dateutil>=2.7.3 in c:\users\luzena\appdata\local\programs\python\python38-32\lib\site-packages (from pandas) (2.8.1)
Requirement already satisfied: six>=1.5 in c:\users\luzena\appdata\local\programs\python\python38-32\lib\site-packages (from python-dateutil>=2.7.3->pandas) (1.15.0)
  
```

Fuente: elaboración propia

4.4. SUBIDA DEL SERVIDOR

Antes de iniciar la aplicación se debe ejecutar el siguiente comando en la consola para subir el servidor y poder acceder a la aplicación. El comando a ejecutar es:

python manage.py runserver. Si durante la ejecución de la aplicación presenta fallas debe realizarse la subida del servidor nuevamente como se indica en la figura 15. Una vez subido el servidor acceder a la aplicación:

url <http://127.0.0.1:8000/login/>

user: admin

clave: hola1234

Figura 15. Subida del servidor.

```

Simbolo del sistema - python manage.py runserver
Microsoft Windows [Version 10.0.17134.1304]
(c) 2018 Microsoft Corporation. Todos los derechos reservados.

C:\Users\Luzena>cd ..
C:\Users>cd ..
C:\>cd C:\Users\Luzena\Repositorios\ESCEQ
C:\Users\Luzena\Repositorios\ESCEQ>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified some issues:

WARNINGS:
registro.Dieta.chip_equino: (fields.W342) Setting unique=True on a ForeignKey has the same effect as using a OneToOneField.
    HINT: ForeignKey(unique=True) is usually better served by a OneToOneField.
registro.Historia_Clinica.equino_chip: (fields.W342) Setting unique=True on a ForeignKey has the same effect as using a OneToOneField.
    HINT: ForeignKey(unique=True) is usually better served by a OneToOneField.

System check identified 2 issues (0 silenced).
October 30, 2020 - 12:48:33
Django version 3.0.7, using settings 'ESCEQ.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
    
```

Fuente: elaboración propia

4.5. DESINSTALACIÓN DE LA APLICACIÓN

4.5.1. Desinstalación de Librerías

Para iniciar con la desinstalación de la aplicación inicialmente proceda a desinstalar las librerías que fueron instaladas. Ejecutar el siguiente comando en la consola de Windows

pip uninstall. Las librerías a desinstalar son:

- *pip uninstall django_filters*
- *pip uninstall django-excel*
- *pip uninstall pandas*
- *pip uninstall matplotlib*
- *pip uninstall seaborn*

- *pip uninstall sklearn*
- *pip uninstall django-import-export*

Como se observa en la figura 16, se debe copiar cada uno de los comandos en la consola de comandos para iniciar con la desinstalación de las librerías.

Figura 16. Desinstalación de librerías.

```

C:\Users\Luzena> pip uninstall django-import-export
Found existing installation: django-import-export 2.3.0
Uninstalling django-import-export-2.3.0:
  Would remove:
    c:\users\luzena\appdata\local\programs\python\python38-32\lib\site-packages\django_import_export-2.3.0.dist-info\*
    c:\users\luzena\appdata\local\programs\python\python38-32\lib\site-packages\import_export\*
Proceed (y/n)?
  
```

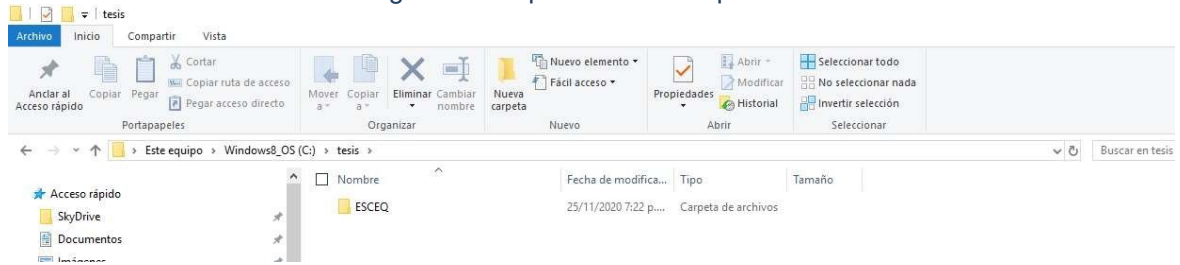
Fuente: elaboración propia

4.5.2. Desinstalación del Repositorio

Ubique el repositorio donde se encuentra la aplicación **C:\tesis\ESCEQ**

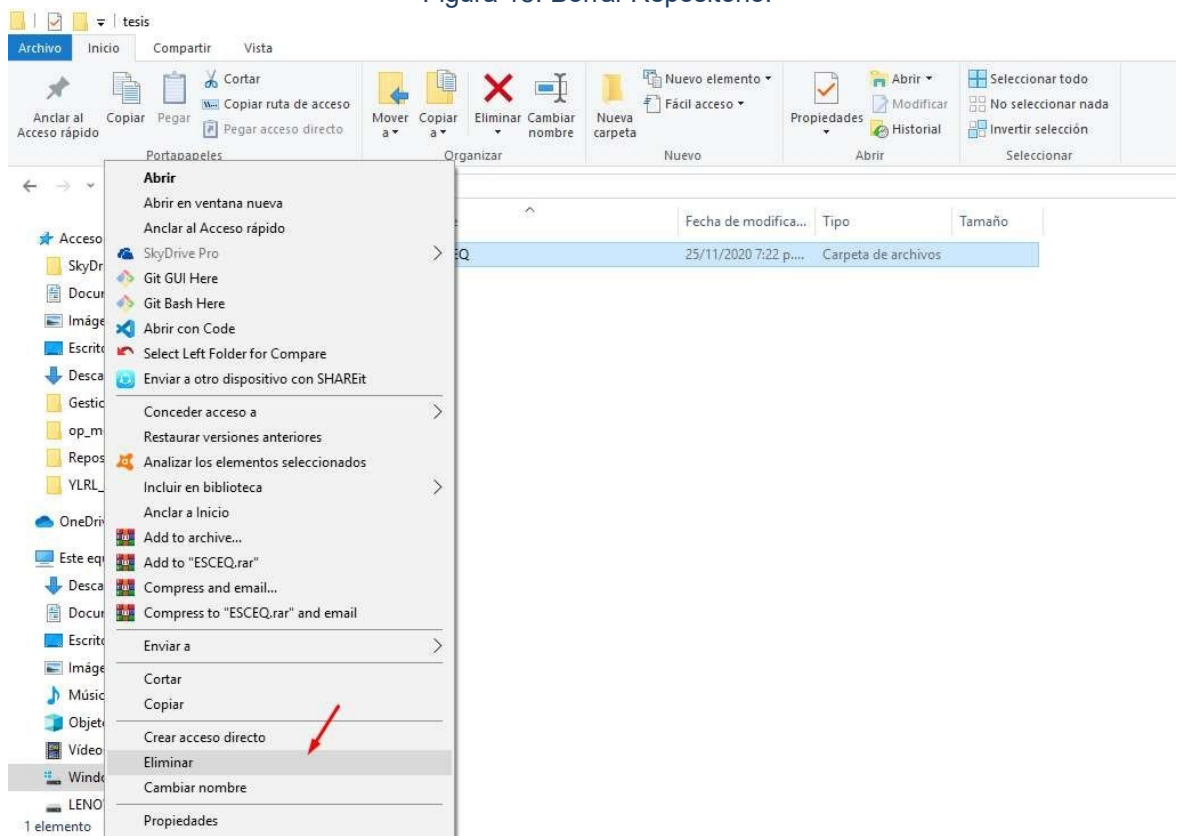
En la figura 17, se presenta el repositorio donde se encuentra la aplicación ubicado en la ruta anteriormente indicada. Se debe borrar el repositorio donde está almacenada la aplicación tal como se indica en la imagen 18.

Figura 17. Repositorio de la aplicación.



Fuente: elaboración propia

Figura 18. Borrar Repositorio.



Fuente: elaboración propia