



Proyecto Final

Estudiantes

Tatiana López Casallas

Juan Esteban Rozo Baquero

Jhon Alexander Rodríguez Aparicio

Docente

Amaury Méndez

Materia

Ingeniería del software II y Big Data

Carrera

Ingeniería Sistemas

Bogotá D.C

202

Explicación del Proyecto - Catastro de Fincas

Introducción

Este proyecto es una aplicación web que permite visualizar y filtrar datos sobre un catastro de fincas. Los datos se extraen de una base de datos MySQL y se muestran en una tabla interactiva donde los usuarios pueden aplicar filtros basados en el tipo de producción, la región, y la cantidad de especímenes específicos en las fincas.

Código 1: Interfaz y Extracción de Datos

El primer código se encarga de conectar con la base de datos MySQL y extraer los datos de la tabla 'catastro_fincas'. Luego, estos datos se codifican en formato JSON para ser utilizados en la interfaz web mediante JavaScript.

1. Se importa el archivo de conexión a la base de datos ('conexion.php') y se crea una instancia de la clase 'Conexiondb'. A través de esta instancia, se establece una conexión con la base de datos usando la función 'crearconexion()'.
2. Luego, se realiza una consulta SQL a la tabla 'catastro_fincas' para obtener todos los registros. Los resultados de la consulta se almacenan en un array asociativo usando la función 'fetch_all(MYSQLI_ASSOC)'.
3. Los datos obtenidos se convierten a formato JSON utilizando 'json_encode()' y se almacenan en la variable JavaScript 'datosOriginales'.
4. Finalmente, se utiliza Bootstrap para diseñar la interfaz del usuario. En esta interfaz, se incluyen formularios desplegables para que el usuario pueda seleccionar los filtros deseados (tipo de alimento, región, categoría de ganado, y rango de cantidades) y botones para buscar y restablecer los filtros.

Código 2: Filtrado y Visualización de Datos

El segundo código, escrito en JavaScript y JQuery, se encarga de manejar la lógica del filtrado y la visualización de los datos en la tabla.

1. Se define la variable 'datosFiltrados' para almacenar los datos que serán filtrados y mostrados en la tabla. Inicialmente, esta variable toma el valor de 'datosOriginales', lo que significa que todos los registros están disponibles para ser mostrados.
2. Al cargar la página, se ejecuta la función 'desplegarDatos()', que inserta los datos en la tabla HTML correspondiente.
3. El botón 'reset' restaura los filtros a sus valores predeterminados y muestra todos los datos nuevamente.

4. El botón 'buscar' aplica los filtros seleccionados por el usuario. Los datos se filtran basándose en los valores seleccionados en los formularios desplegables. Por ejemplo, si se selecciona una región, solo se mostrarán las fincas de esa región. Si se selecciona un tipo de alimento o una categoría de ganado, también se aplican esos filtros.

5. La función 'desplegarDatos()' se encarga de construir dinámicamente las filas de la tabla HTML utilizando los datos filtrados. Cada finca se muestra con su nombre, tipo de alimento, región, coordenadas geográficas (latitud y longitud), y cantidades de cada categoría de ganado.

Código 3: Conexión a la Base de Datos

El tercer código define la clase 'Conexiondb', que se utiliza para establecer la conexión con la base de datos MySQL.

1. La función 'crearconexion()' define los parámetros de conexión: 'host' (servidor), 'user' (usuario de la base de datos), 'pass' (contraseña), y 'bbd' (nombre de la base de datos).
2. Se crea una instancia de la clase 'mysqli' utilizando los parámetros definidos, lo que permite conectarse a la base de datos especificada.
3. Si la conexión falla, se imprime un mensaje de error indicando el problema.
4. Si la conexión es exitosa, se devuelve el objeto 'mysqli' para ser utilizado en las consultas a la base de datos.

Conclusión

Este proyecto combina PHP para manejar la conexión y extracción de datos desde una base de datos MySQL, y JavaScript/JQuery para la manipulación y visualización de esos datos en una interfaz web interactiva. La estructura modular del código facilita el mantenimiento y la ampliación del proyecto, permitiendo agregar nuevas funcionalidades o modificar las existentes de manera sencilla.