

Documentación Técnica del Proyecto Web: Sistema de Gestión Ovino-Caprino

Resumen del Proyecto:

Este sistema web fue diseñado para facilitar la gestión de stock ganadero por parte de productores ovino-caprinos, y proporcionar al administrador acceso completo a la base de datos para su análisis. Desarrollado con PHP, MySQL y HTML/CSS/JavaScript, y ejecutado sobre un servidor Apache configurado en una instancia de Google Cloud con Ubuntu, este sistema incorpora un login seguro con roles diferenciados.

1. Inicio y Planificación del Proyecto

Este proyecto nació como parte de una colaboración con una empresa enfocada en el incentivo de la cría ovina y caprina. El objetivo inicial fue construir un sistema centralizado donde los productores pudieran actualizar mensualmente su inventario, y la administración accediera a reportes históricos y actuales. Se definieron los requerimientos funcionales, incluyendo la segmentación de roles, paneles diferenciados y base de datos estructurada con codificación ``utf8mb4_spanish_ci``.

2. Arquitectura General

El sistema sigue una arquitectura cliente-servidor basada en tecnologías de software libre. La interfaz está diseñada para ser ligera y clara, enfocándose en accesibilidad para usuarios con conocimientos informáticos básicos. La capa de negocio está compuesta por scripts PHP que gestionan validaciones, conexiones, autenticación y CRUD de datos.

3. Estructura de Archivos y Carpetas

La estructura del sistema sigue una organización clara:

- ``/admin/``: Panel exclusivo para el administrador, con acceso total a la base de datos.

- `/productor/`: Panel donde los productores pueden registrar o modificar su stock.
- `/includes/`: Archivos PHP comunes como conexión a base de datos (`db.php`) y verificación de login (`auth.php`).
- `/css/`, `/js/`: Archivos estáticos para la presentación.
- `index.php`: Página de inicio del sistema y portal de acceso.

4. Sistema de Login y Roles

El sistema de autenticación está basado en formularios PHP conectados con la base de datos MySQL. Se utilizan sesiones (`$_SESSION`) para mantener al usuario autenticado. Una vez que se valida el usuario, se verifica su rol y se redirige al panel correspondiente (`/admin/dashboard.php` o `/productor/panel.php`). Las contraseñas están hasheadas con `password_hash()` usando el algoritmo `bcrypt`.

5. Base de Datos

La base de datos `ganado_db` está codificada en `utf8mb4_spanish_ci` para soportar caracteres especiales. Sus tablas clave incluyen:

- `usuarios`: ID, nombre, email, password, rol (admin o productor)
- `stock`: ID, id_usuario, mes, año, cantidad_ovinos, cantidad_caprinos, observaciones
- `registros`: Logs automáticos de cambios en la base de datos para trazabilidad

6. Registro y Edición de Stock

Cada productor, una vez autenticado, puede acceder a un formulario con validaciones (tanto del lado del cliente con JavaScript como del servidor con PHP) donde registra o edita su stock mensual. La clave compuesta `(id_usuario, mes, año)` garantiza la unicidad de registros por período. Si existe un registro previo, se muestra el botón "Editar"; si no, "Guardar".

7. Panel de Administración

El panel del administrador permite:

- Ver y buscar todos los registros de stock por productor
- Exportar datos a Excel o CSV para análisis externo
- Crear, editar o eliminar usuarios productores
- Ver logs de actividad

8. Seguridad y Buenas Prácticas

El sistema fue construido respetando buenas prácticas de seguridad como:

- Prepared Statements para evitar SQL Injection
- Validación del lado cliente y servidor
- Hashing de contraseñas con `password_hash()`
- Uso de sesiones seguras (`session_start()`) y `session_regenerate_id()`
- Control de acceso en cada página con funciones `require_login()` y verificación de rol

9. Hospedaje y Configuración del Servidor

El servidor está montado en Google Cloud con Ubuntu 22.04 LTS. Se utilizó:

- Apache2 como servidor web
- PHP 8.x
- MySQL como gestor de base de datos
- No-IP para DNS dinámico, permitiendo acceso desde IP cambiante

El servidor acepta conexiones SSH desde VS Code usando claves públicas. Se hizo hardening básico desactivando el login por contraseña.

10. Futuras Mejoras

Se planifica incluir:

- Registro por geolocalización del productor
- Reportes gráficos (Chart.js o Google Charts)
- Copias de seguridad automáticas en la nube
- Soporte multilingüe
- Panel móvil responsive

Este documento es una base sólida para el mantenimiento, extensión y comprensión del sistema por nuevos desarrolladores o administradores del proyecto.