

**CluckSys**

Creación de un sistema de información para el sector avícola en la empresa *Pollos teda SAS;* empresa de producción y venta de pollo semicriollo.

**Etapa 1: Análisis y Definición**

**1.1.** Descripción del problema:

*La empresa avícola necesita un software para gestionar su inventario y operaciones de producción de pollo semicriollo. El software debe permitir registrar usuarios, gastos, stock de pollos vivos y sacrificados, peso por pollo y menudencias, y generar un reporte final del proceso productivo.*

**1.2.** Objetivos:

* Desarrollar un software que facilite la gestión de inventario y operaciones para la empresa avícola.
* Optimizar el control de gastos, stock y producción de pollo semicriollo.
* Brindar información precisa para la toma de decisiones estratégicas en la empresa.

**1.3.** Funcionalidades:

* Registro de usuarios: El administrador del sistema podrá crear y gestionar usuarios, asignando roles y permisos de acceso.
* Registro de gastos: Los usuarios podrán registrar los gastos asociados a cada etapa del proceso productivo (crianza, engorde, matanza), desglosados por categorías (compras, vacunas, insumos, alimentos, mano de obra).
* Control de stock: Se llevará un registro detallado del stock de pollos vivos y sacrificados, incluyendo el peso individual de cada pollo en el caso de los sacrificados, se podrá registrar su peso y el total de menudencias en stock.
* Generación de reportes: El sistema generará un reporte final del proceso productivo, incluyendo gastos totales, stock de pollos y menudencias, y datos relevantes para la evaluación del rendimiento.

**Etapa 2: Diseño**

**2.1.** Arquitectura del software:

Se propone una arquitectura compuesta por los siguientes módulos:

* Módulo de autenticación: Manejará el registro, autenticación y autorización de usuarios.
* Módulo de gestión de gastos: Permitirá el registro, clasificación y consulta de gastos por categorías y lotes.
* Módulo de control de stock: Facilitará el registro y actualización del stock de pollos vivos y sacrificados, incluyendo peso individual y menudencias.
* Módulo de generación de reportes: Generará reportes personalizados con información relevante del proceso productivo.

Graficos:

Digrama de flujo:

Inicio

1. Usuario inicia sesión:

* El usuario ingresa su nombre de usuario y contraseña.
* El sistema verifica las credenciales en el módulo de autenticación.

1. Autenticación fallida:

* Si las credenciales son incorrectas, el sistema muestra un mensaje de error y regresa al paso 1.

1. Autenticación exitosa:

* El sistema muestra el menú principal al usuario. El usuario selecciona la opción deseada.

1. Módulo de gestión de gastos:

* Si el usuario selecciona la opción de gestión de gastos, el sistema navega al módulo de gestión de gastos.
* El usuario puede registrar, clasificar y consultar gastos por categorías y lotes.

1. Módulo de control de stock:

* Si el usuario selecciona la opción de control de stock, el sistema navega al módulo de control de stock.
* El usuario puede registrar y actualizar el stock de pollos vivos y sacrificados, incluyendo peso individual y menudencias.

1. Módulo de generación de reportes:

* Si el usuario selecciona la opción de generación de reportes, el sistema navega al módulo de generación de reportes.
* El usuario puede generar reportes personalizados con información relevante del proceso productivo.

1. Cerrar sesión:

* El usuario selecciona la opción de cerrar sesión.
* El sistema cierra la sesión del usuario y regresa al paso 1.

Fin

Digrama UML:

Casos de uso:

* UC1: Iniciar sesión
* UC2: Registrar gasto
* UC3: Clasificar gasto
* UC4: Consultar gastos
* UC5: Registrar stock de pollos vivos
* UC6: Actualizar stock de pollos vivos
* UC7: Registrar stock de pollos sacrificados
* UC8: Actualizar stock de pollos sacrificados
* UC9: Generar reporte

Actores:

* Usuario

Clases:

* Módulo de autenticación
* Funciones:
* Registrar usuario
* Autenticar usuario
* Autorizar usuario
* Módulo de gestión de gastos
* Funciones:
* Registrar gasto
* Clasificar gasto
* Consultar gastos
* Módulo de control de stock
* Funciones:
* Registrar stock de pollos vivos
* Actualizar stock de pollos vivos
* Registrar stock de pollos sacrificados
* Actualizar stock de pollos sacrificados
* Módulo de generación de reportes
* Funciones:
* Generar reporte personalizado

Relaciones:

* Usuario interactúa con Módulo de autenticación
* Usuario interactúa con Módulo de gestión de gastos
* Usuario interactúa con Módulo de control de stock
* Usuario interactúa con Módulo de generación de reportes

Mokup 1

**2.2.** Interfaz de usuario:

Se diseñará una interfaz amigable e intuitiva, accesible para usuarios con diferentes niveles de experiencia tecnológica. Se utilizarán gráficos y tablas para una mejor visualización de la información.

**Etapa 3: Implementación**

3.1. Selección de tecnologías:

1. Lenguaje de programación: Se evaluará la utilización de lenguajes como Python junto a el framework de Django para la parte de Back-end, considerando su versatilidad, robustez y amplia comunidad de desarrolladores.
2. Base de datos: Se seleccionará una base de datos relacional como MySQL o PostgreSQL para almacenar la información de manera organizada y segura.
3. Herramientas de desarrollo: Se emplearán herramientas como lenguaje estructutado HTML5, IDEs (Integrated Development Environments) y frameworks para facilitar el proceso de desarrollo y pruebas, inicialmente se utilizará Bopstrap

3.2. Desarrollo del software:

Modelo de integración:

Django se integra perfectamente con HTML5 y Bootstrap, permitiendo generar vistas HTML dinámicas y renderizarlas en el navegador.

PostgreSQL se conecta fácilmente con Django utilizando adaptadores como psycopg2, permitiendo realizar consultas y manipular datos desde la aplicación Python.

Metodologia:

Se seguirá una metodología de desarrollo ágil, dividiendo el proyecto en iteraciones para obtener retroalimentación constante y mejorar el producto final.

Se realizarán pruebas unitarias y de integración para garantizar el correcto funcionamiento del software.

Etapa 4: Pruebas y despliegue

4.1. Pruebas exhaustivas:

Se realizarán pruebas exhaustivas del software para detectar y corregir errores, asegurando su estabilidad y confiabilidad. Se involucrarán usuarios reales para obtener retroalimentación sobre la usabilidad y funcionalidad del sistema.

4.2. Despliegue y mantenimiento:

Se implementará el software en el entorno de producción de la empresa avícola, brindando capacitación a los usuarios y estableciendo un plan de mantenimiento preventivo y correctivo para garantizar su correcto funcionamiento a largo plazo.

Graficos:

**CopyRight**

IDaveCoder - 2024

David Tejada Vargas - Ingeniero de Software.