|  |  |
| --- | --- |
| Livestock & Co Logo. The Logo displays a cow mostly found in dry and hot climate with long curved horns that point upwards. On each side of the cow is a strain of grain. the colors are muted reds and blues on a beige background with yellow undertones.  Final Project  An interactive GUI for small scale farmers and Homesteaders | Abstract  xxxxxxxxxxxxx  Alisha Rhodes  Ivy Tech Community College |

Contents

[What is this GUI about? 2](#_Toc192116035)

[My Motivation 2](#_Toc192116036)

[Pseudocode: 2](#_Toc192116037)

[Login 4](#_Toc192116038)

[Color Theme: 4](#_Toc192116039)

[MySQL Connection: 4](#_Toc192116040)

[Issues: 5](#_Toc192116041)

[General Comments: 5](#_Toc192116042)

[GitHub 5](#_Toc192116043)

# What is this GUI about?

A Livestock Management System to view Schedules, Health Records and important other Farming Operation Reports such as yields to be expected and local market reports.  
  
Target Group:  
Small Scale farmers, Homesteader, Breeders

# My Motivation

As an upcoming livestock farmer myself, it is a tool I could use daily. I have not found yet a Tool that meets the needs for upcoming Homesteads and Farms rather than established Farms and Operations. Small scale Farming Operations are dying all over the country due to corporations, old age and the high cost of production, land and other factors. This takes a huge hit on a lot of communities that rely on the economics of their local farms. When I decided to pursue my Data Analytics degree, I wanted to learn the Market and use my skills to make small scale farming pay the bills again. I think it is possible even in today’s age.

Pseudocode:

Author: Tesha Alisha Rhodes

Final Project: Livestock Management System

# Main Program Flow

BEGIN

# Set up environment for the application

Initialize customtkinter appearance and color theme

Define custom color scheme for UI components

# Define paths for necessary files

Set path for user database file ("users.csv")

Set path for logo image ("logo.png")

Set path for chicken breeds file ("chicken\_breeds.xlsx")

# Function Definitions

# Load an image from a given path

FUNCTION load\_image(path, size):

TRY

Load image and resize it

EXCEPT Exception

Print error message and return None

# Function to check if the provided password matches the stored password

FUNCTION check\_password(stored\_password, provided\_password):

RETURN whether the provided password matches the stored password

# Function to authenticate the user

FUNCTION authenticate\_user(username, password, callback):

TRY

Open users.csv file

Iterate through each row and check if the username and password match

If found, call callback with success and farm\_id

Else, call callback with failure message

EXCEPT FileNotFoundError

Call callback with error message (database not found)

EXCEPT Exception

Call callback with error message

# Create LoginScreen class for user login interface

CLASS LoginScreen:

Initialize the window and UI components

Define create\_widgets to build the login UI with username, password fields, buttons

Define toggle\_password to show/hide password

Define authenticate to check the entered credentials

Define login\_result to handle login success or failure

# Create App class for the main application interface

CLASS App:

Initialize the main window with the farm\_id

Define load\_chicken\_breeds to load chicken breeds from Excel file

Define create\_sidebar to create navigation buttons

Define logout to log the user out and show the login screen again

Define show\_poultry to display poultry management interface

Define create\_poultry\_tab to build UI for poultry management

Define load\_poultry\_data to load poultry data from farm Excel file and display it

Define populate\_chicken\_details to populate chicken details on the UI

Define add\_chicken to add a new chicken to the poultry data

Define update\_chicken to update an existing chicken record

# Main Program Flow

CREATE a LoginScreen object with success callback that creates an App instance upon successful login

Run the login screen loop

END

Imports:

import os

import tkinter as tk

from tkinter import ttk

from PIL import Image, ImageTk

import customtkinter as ctk

import dash

import dash\_core\_components as dcc

from dash import html

# Login

Dataset 1:

Username: Windhof  
Password: QRSTuvw

# Color Theme:

#3A5460 #4F7486 #D78C8E #F1686A #FDDAC0 #FEE6C4

# MySQL Connection:

Edit: 03/06:  
  
I removed the connection to SQL as it is too complicated to open up the Server and ensure that data is available to everyone. I replaced SQL Data with Excel files but kept the “key” approach. Data will still be loaded with primary key in mind.  
  
  
if used, otherwise csv file info here  
Username: teacher

Password: IvyTech2025!

# Issues:

02/18 – Upload of a custom Theme via .json file. The file is not being detected as part of my library.  
 Solution: Awaiting Solution

03/06 – Solved Json file issue with implementing it directly into code, new issues are updating info in excel file and aesthetically pleasing data visualization

# General Comments:

* I built a database in MySQL Workbench for application authentication. I'm uncertain if it will be part of my final project as I haven't set up a localhost or gateway yet. It might be an optional addition, and I'll write code to retrieve data from an Excel file or a text list.
* I tackle issues as they arise, creating solutions on the fly and learning necessary code segments along the way. I debug every code segment meticulously.
* I definitely have to say that I was hoping to get more done, but It is just the introduction class. I am sure as the semesters go by I will add classes to the GUI and make it work better, smoother and more pleasing to the eye.
* The Logo is designed by me.
* Debugging this code was difficult, as the new section was improved another didn’t work out at all. I got it to a point where it is working although it is rough.

GitHub:

https://github.com/alirho45/LivestockCo.git