

Manual de mantenimiento técnico del sistema de gestión de Glam makeup store

1 Tabla de contenido

2	Introducción.....	6
2.1	Propósito del manual:	6
2.2	Descripción general del sistema:	6
2.3	Objetivos del sistema:	6
2.4	Uso de GitHub:	6
3	Requisitos para el sistema.	7
3.1	Hardware:	7
3.2	Software:	7
3.3	Librerías:	7
3.4	Control de versiones:	7
4	Arquitectura del sistema.....	7
5	Instalaciones para el sistema	7
5.1	Preparación del entorno:.....	8
5.2	Configuración Inicial:.....	8
5.3	Ejecución del Sistema:.....	8
6	Estructura del código.....	8
6.1	Login	8
6.1.1	Descripción de la Clase	8
6.1.2	Propósito.....	8
6.1.3	Métodos Clave	8
6.2	AdminSoporteManager	9
6.2.1	Descripción de la Clase:	9
6.2.2	Propósito:.....	9
6.2.3	Métodos Clave	9
6.2.4	Interacción con Otros Componentes:.....	9
6.3	Aplicacion.....	10
6.3.1	Descripción de la Clase	10

6.3.2	Propósito.....	10
6.3.3	Métodos Clave	10
6.3.4	Interacción con Otros Componentes.....	10
6.4	GestionDatos	10
6.4.1	Descripción de la Clase	10
6.4.2	Propósito.....	10
6.4.3	Métodos Clave	10
6.4.4	Interacción con Otros Componentes.....	10
7	Interfaz del usuario.....	10
7.1	Descripción de pantallas:	10
7.2	Elementos:	11
7.3	Flujo de navegación:	11
7.3.1	Pantalla de login	11
7.3.2	Ventana principal:	11
7.3.3	Admin:.....	11
7.3.4	Soporte:	11
7.3.5	Caja:	12
8	Funcionalidades del sistema	12
8.1	Módulo de gestión de clientes:.....	12
8.1.1	Crear cliente:.....	12
8.1.2	Modificar cliente:	12
8.1.3	Eliminar cliente:.....	13
8.1.4	Ver lista de clientes:.....	13
8.2	Módulo de gestión de ventas:	13
8.2.1	Registrar nueva venta:	13
8.2.2	Modificar ventas:.....	13
8.2.3	Eliminar ventas:	13
8.2.4	Añadir y eliminar métodos de pago:.....	13
8.3	Módulo de gestión del inventario de productos:	14
8.3.1	Crear producto:	14
8.3.2	Modificar producto:	14

8.3.3	Descontinuar producto:.....	14
8.3.4	Reabastecer stock:	14
8.4	Módulo de gestión del catálogo de servicios:.....	14
8.4.1	Crear nuevo servicio:	15
8.4.2	Modificar servicio:	15
8.4.3	Eliminar servicio:.....	15
8.5	Módulo de generación de reportes:.....	15
8.5.1	Reportes diarios:.....	15
8.5.2	Reportes personalizados:	15
9	Data.py y clase gestión de datos	16
9.1	Inicialización de la Clase (GestionDatos):.....	16
9.2	Creación de DataFrames	17
9.3	Carga de DataFrames:.....	17
9.4	Guardado de DataFrames:	17
9.5	Ajuste de Columnas en Excel	17
9.6	Verificaciones del Sistema:	17
9.6.1	“verificar_admin_existente”: Comprueba si existe al menos un usuario administrador en el sistema.....	17
9.6.2	“verificar_stock_no_negativo”: Asegura que las unidades actuales de los productos no sean negativas.....	17
9.6.3	“realizar_verificaciones”: Realiza ambas verificaciones anteriores y proporciona advertencias o errores según corresponda.	18
9.7	Gestión de Clientes.....	18
9.7.1	“agregar_producto”: Añade un nuevo producto al inventario.	18
9.7.2	“buscar_producto”: Busca un producto por su referencia.	18
9.7.3	“verificar_disponibilidad”: Verifica si un producto está disponible en el inventario.....	18
9.7.4	“actualizar_producto”: Actualiza la información de un producto.....	18
9.8	Gestión de Ventas.....	18
9.8.1	“agregar_venta_servicio”: Añade una nueva venta de servicio.....	18
9.8.2	“agregar_venta_producto”: Añade una nueva venta de producto....	18

9.8.3	“actualizar_venta”: Actualiza una venta existente (producto o servicio).....	18
9.8.4	“eliminar_venta”: Elimina una venta existente (producto o servicio). 18	
9.9	Gestión de Servicios	18
9.9.1	“agregar_servicio”: Añade un nuevo servicio.	19
9.9.2	“actualizar_servicio”: Actualiza la información de un servicio.	19
9.9.3	“buscar_servicio”: Busca un servicio por su ID.	19
9.9.4	“eliminar_servicio”: Elimina un servicio por su ID.....	19
9.10	Gestión de Reservas de Servicios	19
9.10.1	“reservar_servicio”: Añade una nueva reserva de servicio	19
9.10.2	“cancelar_reserva_servicio”: Cancela una reserva de servicio.	19
9.10.3	“buscar_reserva_por_servicio”: Busca reservas de servicio por ID de servicio. 19	
9.11	Gestión de Usuarios.....	19
9.11.1	“agregar_usuario”: Añade un nuevo usuario.....	19
9.11.2	“buscar_usuario”: Busca un usuario por su nombre de usuario.....	19
9.11.3	“modificar_usuario”: Modifica la información de un usuario.	19
9.12	Gestión de Facturas.....	19
9.12.1	“conversion_factura”: Genera una factura basada en las ventas de productos y servicios de un cliente específico en una fecha determinada.	20
10	Autenticación y autorización	20
10.1	Autenticación.....	20
10.1.1	Ingreso de credenciales:	20
10.1.2	Verificación de credenciales:	20
10.1.3	Acceso a la ventana principal:	20
10.2	Autorización	20
10.2.1	Determinación de rol:.....	20
10.2.2	Acceso a las funciones	20
10.2.3	Restimación de funciones	21
10.3	Flujo de Autenticación y Autorización.....	21
10.3.1	Pantalla de Login:	21

10.3.2	Redirección a la Ventana Principal:	21
10.3.3	Acceso a Funcionalidades:	21
11	Validaciones.....	22
12	Conexiones	22
13	Manejo de errores	22

2 Introducción

2.1 Propósito del manual:

Este manual proporciona una guía detallada para el uso y mantenimiento del sistema de Glam Makeup Store, incluyendo la instalación, configuración y administración del sistema.

2.2 Descripción general del sistema:

El sistema de Glam Makeup Store es una aplicación para la gestión de una tienda de maquillaje, que incluye funcionalidades para gestionar clientes, ventas, inventario de productos, catálogo de servicios, administración de usuarios y generación de reportes.

2.3 Objetivos del sistema:

- ❖ Optimizar la gestión de inventario y ventas.
- ❖ Facilitar la administración de clientes y servicios.
- ❖ Proporcionar una interfaz de usuario intuitiva para facilitar las operaciones diarias.
- ❖ Generar reportes detallados para análisis y toma de decisiones.

2.4 Uso de GitHub:

GitHub se utiliza para el control de versiones del código fuente del sistema. Esto permite la colaboración entre desarrolladores, seguimiento de cambios y gestión de versiones del proyecto. El repositorio de GitHub incluye el código fuente, documentación y versiones anteriores del sistema.

3 Requisitos para el sistema.

3.1 Hardware:

- ❖ Procesador: Intel Core i3 o superior
- ❖ Memoria RAM: 4 GB o más
- ❖ Espacio en Disco: 500 MB para la aplicación y datos

3.2 Software:

- ❖ Sistema Operativo: Windows 10 o superior, Linux
- ❖ Python 3.8 o superior
- ❖ Git para control de versiones

3.3 Librerías:

Las siguientes librerías se usaron en el desarrollo del programa:

- ❖ PyQt5 – Versión 5.15.10
- ❖ Pandas – Versión 2.2.0
- ❖ Openpyxl – Versión 3.1.2
- ❖ DateTime – Versión 5.5

3.4 Control de versiones:

- ❖ GitHub:

El repositorio del proyecto se encuentra en GitHub y se utiliza para el control de versiones. Es necesario tener instalado Git para clonar el repositorio y gestionar las versiones del código.

4 Arquitectura del sistema

5 Instalaciones para el sistema

Para instalar el sistema, siga estos pasos:

5.1 Preparación del entorno:

- ❖ Asegúrese de tener instalado Python 3.x y PyQt5.
- ❖ Instale las dependencias requeridas utilizando “pip install -r requirements.txt”.

5.2 Configuración Inicial:

- ❖ Descargue y descomprima el proyecto en el directorio deseado.
- ❖ Configure la base de datos según las especificaciones del archivo “config.py”.

5.3 Ejecución del Sistema:

- ❖ Navegue al directorio raíz del proyecto.
- ❖ Ejecute el archivo principal del sistema utilizando el comando “python main.py”.

6 Estructura del código

Se proporcionará una descripción de la estructura del código, incluyendo la organización de archivos y carpetas, y una explicación de los componentes principales del sistema:

6.1 Login

6.1.1 Descripción de la Clase

La clase se llama “Login”

6.1.2 Propósito

La clase “Login” maneja la interfaz de inicio de sesión, la validación de credenciales, y la redirección de usuarios según su rol.

6.1.3 Métodos Clave

- “__init__” (self, app): Inicializa la clase y configura la interfaz de usuario.
- “limpiarCampo” (self): Limpia los campos de usuario y contraseña.

- “checkFields” (self): Verifica los campos de entrada y autentica al usuario.
- “showMessage(self, message, is_error=True): Muestra mensajes de error o éxito en la interfaz.
- “showErrorMessage” (self, message): Muestra un cuadro de mensaje de error.
- “authenticate_user” (self, username, password): Autentica las credenciales del usuario.
- “openAdminSupportWindow” (self, user_role: str): Abre la ventana de soporte o administración según el rol del usuario.
- “openCajaWindow” (self): Abre la ventana de la caja.
- “abrir_manual” (self): Abre el manual de usuario en PDF.

6.2 AdminSoporteManager

6.2.1 Descripción de la Clase:

La clase se llama “AdminSoporteManager”.

6.2.2 Propósito:

Este componente carga estilos CSS y gestiona las tareas administrativas y de soporte.

6.2.3 Métodos Clave

- ❖ “__init__” (self, parent, user_role): Inicializa la clase con el rol de usuario específico.
- ❖ “leer_estilos” (self, app, estilos): Carga y aplica los estilos CSS a la aplicación.
- ❖ “run” (self): Ejecuta la ventana administrativa.

6.2.4 Interacción con Otros Componentes:

- ❖ Carga los estilos y configura la interfaz según el rol del usuario.
- ❖ Interactúa con GestionDatos para obtener y manipular datos necesarios para las tareas administrativas y de soporte.

6.3 Aplicacion

6.3.1 Descripción de la Clase

La clase se llama “Aplicacion”.

6.3.2 Propósito

Gestiona la interfaz principal de la caja registradora, facilitando la gestión de ventas y transacciones.

6.3.3 Métodos Clave

- ❖ “__init__” (self, parent): Inicializa y configura la interfaz de la caja.
- ❖ “show” (self): Muestra la ventana de la aplicación de caja.

6.3.4 Interacción con Otros Componentes

Interactúa con “GestionDatos” para realizar operaciones de ventas y gestionar transacciones.

6.4 GestionDatos

6.4.1 Descripción de la Clase

6.4.2 Propósito

6.4.3 Métodos Clave

6.4.4 Interacción con Otros Componentes

7 Interfaz del usuario

El sistema registra el ID del usuario al iniciar sesión para rastrear las acciones realizadas por cada perfil y asegurar el correcto seguimiento de los datos y modificaciones.

7.1 Descripción de pantallas:

Pantalla inicial donde el usuario ingresa sus credenciales para acceder al sistema.

7.2 Elementos:

- ❖ Campo para ingresar el nombre de usuario.
- ❖ Campo para ingresar la contraseña.
- ❖ Botones de "Cancelar" y "Confirmar" para manejar la autenticación.
- ❖ Botón de "Cerrar programa" para salir de la aplicación.
- ❖ Mensaje de error si las credenciales son incorrectas ("Usuario/contraseña incorrectos, ingrese de nuevo").

7.3 Flujo de navegación:

7.3.1 Pantalla de login

- ❖ El usuario ingresa sus credenciales (nombre de usuario y contraseña).
- ❖ Tras autenticarse correctamente, el sistema redirige a la Ventana Principal correspondiente al rol del usuario (Admin, Soporte, Caja).

7.3.2 Ventana principal:

El usuario puede navegar a través de las secciones disponibles según su rol.

7.3.3 Admin:

El rol admin tiene acceso a las siguientes funcionalidades en el programa:

- ❖ Gestionar catálogo de servicios.
- ❖ Gestionar inventario de productos.
- ❖ Gestionar ventas.
- ❖ Realizar reportes personalizados.
- ❖ Gestionar clientes.

7.3.4 Soporte:

El rol soporte tiene acceso a las siguientes funcionalidades en el programa:

- ❖ Gestionar catálogo de servicios.
- ❖ Gestionar inventario de productos.
- ❖ Gestionar ventas.
- ❖ Realizar reportes personalizados.
- ❖ Gestionar clientes.
- ❖ Administrar usuarios.

7.3.5 Caja:

El rol caja tiene acceso a las siguientes funcionalidades en el programa:

- ❖ Realizar ventas.
- ❖ Realizar reportes diarios.
- ❖ Añadir nuevos clientes.

8 Funcionalidades del sistema

Esta sección describe las principales funcionalidades del sistema de Glam Makeup Store, detallando las operaciones que pueden realizarse en cada módulo.

8.1 Módulo de gestión de clientes:

El módulo de gestión de clientes cuenta con las siguientes funciones:

8.1.1 Crear cliente:

- ❖ Permite añadir nuevos clientes al sistema.
- ❖ Los datos requeridos incluyen nombre, cédula, teléfono, y dirección.
- ❖ Proceso: El usuario ingresa la información del cliente en un formulario, y al guardar, se crea un nuevo registro en el DataFrame correspondiente.

8.1.2 Modificar cliente:

- ❖ Permite actualizar la información de un cliente existente.
- ❖ Proceso: El usuario selecciona un cliente de la lista, realiza los cambios necesarios en el formulario y guarda los cambios.

8.1.3 Eliminar cliente:

- ❖ Permite eliminar un cliente del sistema.
- ❖ Proceso: El usuario selecciona un cliente de la lista y confirma la eliminación. El registro se esconde del DataFrame.

8.1.4 Ver lista de clientes:

- ❖ Permite visualizar todos los clientes registrados en el sistema.
- ❖ Proceso: El sistema muestra una tabla con los datos de todos los clientes almacenados en el DataFrame.

8.2 Módulo de gestión de ventas:

El módulo de gestión de ventas cuenta con las siguientes funciones:

8.2.1 Registrar nueva venta:

- ❖ Permite registrar una nueva venta en el sistema.
- ❖ Los datos requeridos incluyen producto/servicio vendido, cantidad, precio, y método de pago.
- ❖ Proceso: El usuario ingresa la información de la venta en un formulario, y al guardar, se crea un nuevo registro en el DataFrame de ventas.

8.2.2 Modificar ventas:

- ❖ Permite actualizar la información de una venta existente.
- ❖ Proceso: El usuario selecciona una venta de la lista, realiza los cambios necesarios en el formulario y guarda los cambios.

8.2.3 Eliminar ventas:

- ❖ Permite eliminar una venta del sistema.
- ❖ Proceso: El usuario selecciona una venta de la lista y confirma la eliminación. El registro se esconde del DataFrame.

8.2.4 Añadir y eliminar métodos de pago:

- ❖ Permite gestionar los métodos de pago aceptados en la tienda.

- ❖ Proceso: El usuario puede añadir nuevos métodos de pago o eliminar los existentes desde la configuración del sistema.

8.3 Módulo de gestión del inventario de productos:

El módulo de gestión del inventario de productos cuenta con las siguientes funciones:

8.3.1 Crear producto:

- ❖ Permite añadir nuevos productos al inventario.
- ❖ Los datos requeridos incluyen nombre del producto, descripción, precio, y cantidad en stock.
- ❖ Proceso: El usuario ingresa la información del producto en un formulario, y al guardar, se crea un nuevo registro en el DataFrame de productos.

8.3.2 Modificar producto:

- ❖ Permite actualizar la información de un producto existente.
- ❖ Proceso: El usuario selecciona un producto de la lista, realiza los cambios necesarios en el formulario y guarda los cambios.

8.3.3 Descontinuar producto:

- ❖ Permite descontinuar un producto del inventario.
- ❖ Proceso: El usuario selecciona un producto de la lista y confirma la descontinuación. El registro sale descontinuado del DataFrame.

8.3.4 Reabastecer stock:

- ❖ Permite aumentar la cantidad de un producto en el inventario.
- ❖ Proceso: El usuario selecciona un producto y especifica la cantidad a añadir al stock actual.

8.4 Módulo de gestión del catálogo de servicios:

El módulo de gestión del catálogo de servicios cuenta con las siguientes funciones:

8.4.1 Crear nuevo servicio:

- ❖ Permite añadir nuevos servicios al catálogo de la tienda.
- ❖ Los datos requeridos incluyen nombre del servicio, descripción, duración y precio.
- ❖ Proceso: El usuario ingresa la información del servicio en un formulario, y al guardar, se crea un nuevo registro en el DataFrame de servicios.

8.4.2 Modificar servicio:

- ❖ Permite actualizar la información de un servicio existente.
- ❖ Proceso: El usuario selecciona un servicio de la lista, realiza los cambios necesarios en el formulario y guarda los cambios.

8.4.3 Eliminar servicio:

- ❖ Permite eliminar un servicio del catálogo.
- ❖ Proceso: El usuario selecciona un servicio de la lista y confirma la eliminación. El registro se esconde del DataFrame.

8.5 Módulo de generación de reportes:

El módulo de generación de reportes cuenta con las siguientes funciones:

8.5.1 Reportes diarios:

- ❖ Permite generar reportes diarios de ventas, inventario, y servicios de manera automática interactuando con el reporte de su respectiva tabla.

8.5.2 Reportes personalizados:

El apartado de reportes personalizados se divide en 3 tipos de reportes, cada uno basados en distintos filtros y criterios definidos por el usuario

8.5.2.1 Reportes de inventario

- ❖ Permite generar reportes del estado del inventario.

- ❖ Proceso: El usuario selecciona el rango de fechas y genera el reporte, que se muestra en una tabla o se guarda en un archivo PDF.

8.5.2.2 *Reportes de servicios*

- ❖ Permite generar reportes de los servicios realizados en un periodo específico.
- ❖ Proceso: El usuario selecciona el rango de fechas y genera el reporte, que se muestra en una tabla o se guarda en un archivo PDF.

8.5.2.3 *Reportes de ventas*

- ❖ Permite generar reportes de ventas en un periodo específico.
- ❖ Proceso: El usuario selecciona el rango de fechas y genera el reporte, que se muestra en una tabla o se guarda en un archivo PDF.

9 Data.py y clase gestión de datos

El archivo de Data.py es donde se encuentran todas las funciones referentes a los dataframes, aquí se encuentra la clase Gestión de datos desde donde se inicializa la base de datos y con sus métodos, se manipulan las diferentes tablas que tiene

9.1 Inicialización de la Clase (GestionDatos):

- ❖ Descripción:
Esta clase es responsable de gestionar los datos de la tienda de maquillaje, incluyendo clientes, productos, ventas, servicios, usuarios, roles y más.
- ❖ Método constructor:
“__init__”: Inicializa la clase, establece el nombre del archivo Excel, crea y carga los dataframes, y guarda los dataframes en el archivo Excel.

9.2 Creación de DataFrames

❖ Método:

“crear_dataframes”: Inicializa las columnas y crea los dataframes vacíos para clientes, productos, métodos de pago, ventas de productos y servicios, servicios, reservas de servicios, facturas, usuarios y roles. Además, añade datos predeterminados para métodos de pago y roles, y crea un usuario de soporte por defecto.

9.3 Carga de DataFrames:

❖ Método:

“cargar_dataframes”: Intenta cargar los dataframes desde un archivo Excel existente. Si el archivo no se encuentra, se crearán nuevos dataframes.

9.4 Guardado de DataFrames:

❖ Método:

“guardar_dataframes”: Guarda todos los dataframes en un archivo Excel utilizando ExcelWriter y ajusta las columnas para mejorar la legibilidad.

9.5 Ajuste de Columnas en Excel

❖ Método:

“ajustar_columnas_excel”: Ajusta el ancho de las columnas en el archivo Excel basado en la longitud máxima de los valores en cada columna.

9.6 Verificaciones del Sistema:

❖ Método:

9.6.1 “verificar_admin_existente”: Comprueba si existe al menos un usuario administrador en el sistema.

9.6.2 “verificar_stock_no_negativo”: Asegura que las unidades actuales de los productos no sean negativas.

9.6.3 “realizar_verificaciones”: Realiza ambas verificaciones anteriores y proporciona advertencias o errores según corresponda.

9.7 Gestión de Clientes

❖ Métodos

9.7.1 “agregar_producto”: Añade un nuevo producto al inventario.

9.7.2 “buscar_producto”: Busca un producto por su referencia.

9.7.3 “verificar_disponibilidad”: Verifica si un producto está disponible en el inventario.

9.7.4 “actualizar_producto”: Actualiza la información de un producto

9.8 Gestión de Ventas

❖ Métodos

9.8.1 “agregar_venta_servicio”: Añade una nueva venta de servicio.

9.8.2 “agregar_venta_producto”: Añade una nueva venta de producto.

9.8.3 “actualizar_venta”: Actualiza una venta existente (producto o servicio).

9.8.4 “eliminar_venta”: Elimina una venta existente (producto o servicio).

9.9 Gestión de Servicios

❖ Métodos

9.9.1 “agregar_servicio”: Añade un nuevo servicio.

9.9.2 “actualizar_servicio”: Actualiza la información de un servicio.

9.9.3 “buscar_servicio”: Busca un servicio por su ID.

9.9.4 “eliminar_servicio”: Elimina un servicio por su ID.

9.10 Gestión de Reservas de Servicios

❖ Métodos

9.10.1 “reservar_servicio”: Añade una nueva reserva de servicio

9.10.2 “cancelar_reserva_servicio”: Cancela una reserva de servicio.

9.10.3 “buscar_reserva_por_servicio”: Busca reservas de servicio por ID de servicio.

9.11 Gestión de Usuarios

❖ Métodos

9.11.1 “agregar_usuario”: Añade un nuevo usuario.

9.11.2 “buscar_usuario”: Busca un usuario por su nombre de usuario.

9.11.3 “modificar_usuario”: Modifica la información de un usuario.

9.12 Gestión de Facturas

❖ Métodos

9.12.1 “conversion_factura”: Genera una factura basada en las ventas de productos y servicios de un cliente específico en una fecha determinada.

10 Autenticación y autorización

10.1 Autenticación

10.1.1 Ingreso de credenciales:

El usuario accede a la pantalla de Login e ingresa su nombre de usuario y contraseña en los campos correspondientes.

10.1.2 Verificación de credenciales:

El sistema compara las credenciales ingresadas con los datos almacenados en la base de datos (o un archivo Excel en este caso).

- ❖ Código de ejemplo para la verificación (Tomar foto del código de verificación)

10.1.3 Acceso a la ventana principal:

- ❖ Si las credenciales son correctas, el sistema redirige al usuario a la Ventana Principal correspondiente a su rol.
- ❖ Si las credenciales son incorrectas, el sistema muestra un mensaje de error indicando que las credenciales no son válidas y solicita volver a ingresarlas.

10.2 Autorización

10.2.1 Determinación de rol:

Una vez autenticado, el sistema determina el rol del usuario a partir de la base de datos.

- ❖ Código de ejemplo para la autorización (Tomar foto del código de verificación)

10.2.2 Acceso a las funciones

Dependiendo del rol del usuario, se le otorga acceso a diferentes funcionalidades dentro del sistema:

- ❖ Admin:
Acceso completo a la gestión de inventarios, ventas, clientes, reportes y usuarios.
- ❖ Soporte
Acceso a la gestión de clientes, ventas y reportes, así como a la administración de usuarios.
- ❖ Caja
Acceso a la realización de ventas y generación de reportes diarios.

10.2.3 Restricción de funciones

Cada sección de la aplicación verifica el rol del usuario antes de permitir el acceso a sus funcionalidades específicas

10.3 Flujo de Autenticación y Autorización

Con este flujo, se asegura que solo los usuarios autorizados puedan acceder a las funcionalidades específicas de acuerdo con su rol, manteniendo la seguridad y el orden en el sistema de la tienda de maquillaje.

10.3.1 Pantalla de Login:

- ❖ Usuario ingresa credenciales.
- ❖ Sistema verifica las credenciales.
- ❖ Si son correctas, determina el rol del usuario.

10.3.2 Redirección a la Ventana Principal:

- ❖ Dependiendo del rol, el usuario es redirigido a la Ventana Principal correspondiente.
- ❖ El sistema muestra las funcionalidades adecuadas según el rol del usuario.

10.3.3 Acceso a Funcionalidades:

- ❖ Usuario navega y accede a las funcionalidades disponibles para su rol.

- ❖ Sistema controla el acceso y muestra solo las opciones permitidas para cada rol.

11 Validaciones

Las validaciones del código se encuentran en la carpeta API en el archivo validaciones.py, estas validaciones se exportan al resto del código, algunas de estas validaciones incluyen

- ❖ Validaciones de nombre
- ❖ Validaciones de cedula
- ❖ Validaciones de teléfono
- ❖ Validaciones de código de barras
- ❖ Validaciones de precio
- ❖ Validaciones de usuario
- ❖ Validaciones de contraseña
- ❖ Validaciones de stock
- ❖ Validaciones de cantidad comprada

12 Conexiones

El sistema se inicia desde el archivo main.py, y desde ahí se conecta al login.py, el login.py conecta con las ventanas de caja, soporte y admin, y a su vez conecta con las validaciones correspondientes

13 Manejo de errores