

MANUAL TÉCNICO MI MUEBLERÍA
por: ERICK DANIEL MORALES XICARA
Carné: 201930699

Manual Técnico

El software puede ser utilizado en cualquier sistema operativo ya que se utilizo java, también se puede ejecutar en la mayoría de computadoras que tengan java 7 o superior. Se utilizo el JDK11 de java, y el IDE NetBeans Apache 12.3. A su vez se utilizo Tomcat 9 que se utiliza como contenedor web para el soporte de servlets y JSPs.

El software consiste en el manejo de una mueblería , en la cual presenta varias interfaces de acuerdo al tipo de usuario que lo utilice, donde se pueden realizar distintas acciones siempre dependiendo del tipo de usuario que es, así como crear nuevos muebles que se fabricaran, realizar ventas y devoluciones de las mismas entre muchas funciones mas, en este caso se utilizaron lenguajes como Java, Html y CSS para la implementación web del software

El software a su vez contiene varias clases y manejadores que permiten iniciar y ejecutar el sistema antes mencionado.

PAQUETE WEB

Este paquete contiene todas las implementaciones web que permiten las interfaces graficas con la cual interactuara el usuario.

Administración: este subpaquete contiene todas las interfaces graficas necesarias para el área de administración.

- *creación_usuario*: este jsp permite la creación de un nuevo usuario en el sistema.
- *creación_mueble*: este jsp permite la creación de un nuevo mueble que se prora producir en la fabrica.
- *ensamble_pieza*: este jsp permite asignarle que piezas son necesarias para en ensamble de mueble.
- *i_administración*: este jsp se realizo como un menú que permitirá al usuario administrador navegar por sus funciones.
- *mod_usuario*: este jsp permite la modificacion de algun usuario, la modificacion es cancelar intermitentemente algun usuario.
- *rep_dev*: este jsp presenta el reporte de devoluciones en un rango de fechas.
- *rep_mas_vendido*: este jsp presenta un reporte del usuario con mas ventas en un rango de fechas.
- *rep_mueble_mas*: este jsp presenta un reporte del mueble mas vendido en un rango de fechas.
- *rep_mueble_menos*: este jsp presenta un reporte del mueble menos vendido en un rango de fechas.
- *rep_venta*: este jsp presenta un reporte de las ventas realizadas en un rango de fechas.

Venta: este subpaquete contiene todas las interfaces graficas necesarias para el usuario vendedor.

- *buscador_final:* este jsp presenta una consulta de los muebles disponibles.
- *consulta_compra_intervalo:* este jsp presenta una consulta de algún cliente en un rango de fechas.
- *consulta_dia:* este jsp presenta una consulta de todas las compras del dia.
- *consulta_factura:* este jsp presenta una consulta sobre alguna factura en especifico.
- *dev_intervalo:* este jsp presenta una consulta sobre las devoluciones de un cliente en un intervalo de tiempo.
- *i_venta:* este jsp se utiliza como menú para la movilización del usuario vendedor por el software.
- *prueba_buscador:* este jsp se utilizo para probar los buscadores.
- *realizar_venta:* este jsp permite al vendedor iniciar una venta, donde verificara si el cliente ya esta registrado.
- *reintegración_mueble:* este jsp permite al vendedor reintegrar un mueble a la empresa.
- *venta_cliente:* este jsp permite al vendedor realizar una venta de un mueble por medio del código.
- *venta_sin_cliente:* este jsp permite al vendedor registrar al nuevo cliente antes de realizar una venta.

Fabrica: este subpaquete contiene todas las interfaces graficas necesarias para el usuario de fabrica.

- *consulta_fechas:* este jsp muestra todos los muebles creados.
- *consulta_fechas_A:* este jsp muestra todos los muebles creados de forma ascendente por medio de la fecha de ensamble.

- consulta_pieza: este jsp muestra todas las piezas disponibles dentro de la fabrica.
- consulta_pieza_A: este jsp muestra todas las piezas dentro de la fabrica de forma ascendente basándose en en la cantidad de cada una.
- consulta_pieza_D: este jsp muestra todas las piezas de la fabrica de forma descendente basándose en la cantidad de cada una.
- consulta_fechas_D: este jsp muestra todos los muebles creados de forma descendente por medio de la fecha de ensamble.
- editar_p: este jsp permite editar alguna pieza.
- eliminar_p: este jsp permite eliminar alguna pieza.
- ensamblar_m: este jsp permite ensamblar un mueble nuevo que sera y estara disponible para la venta.
- i_fabrica: este jsp presenta un menu que sera utilizado por el usuario fabricante para movilizarse por el software.
- mod_piezas: este jsp presenta un listado de todas las piezas en donde se puede buscar una pieza para luego modificar.
- n_pieza: este jsp permite el registro de una nueva pieza.

Inicio: este subpaquete contiene las interfaces para el inicio de sesión y la carga de los archivos como la presentación si hay errores.

Imagen: este subpaquete contiene las imágenes utilizadas para la interfaces grafica.

PAQUETE CLASES

Este paquete contiene todas las clases utilizadas para el manejo de las entidades en la base de datos.

Cliente: solamente contiene metodo constructor y ademas de sus getters y setters.

Devolución: solamente contiene metodo constructor y ademas de sus getters y setters.

Ensamble_Pieza: solamente contiene metodo constructor y ademas de sus getters y setters.

Estado Mueble Ensamblado: solamente contiene metodo constructor y ademas de sus getters y setters.

Factura: solamente contiene metodo constructor y ademas de sus getters y setters.

Mueble: solamente contiene metodo constructor y ademas de sus getters y setters.

Mueble Ensamblado: contiene metodo constructor y ademas de sus getters y setters.

- cod_mueble: genera un codigo indentificador aleatorio con un numero entre 10 y 26 caracteres.

Pieza: solamente contiene metodo constructor y ademas de sus getters y setters.

Tipo_Usuario: solamente contiene metodo constructor y ademas de sus getters y setters.

Usuario: solamente contiene metodo constructor y ademas de sus getters y setters.

PAQUETE CONEXIÓN

Contiene una clase que permitirá la conexión con la base de datos.

Conexion_Sql: es la encargada de realizar la conexión con la base de datos. Aparte de su constructor contiene:

- *conectar:* es el encargado de conectar con la base de datos.
- *Desconectar:* permite desconectar la conexión con la base de datos.

PAQUETE MANEJADORES

Aqui se encuentran todas las clases que tendran como objetivo llevar la logica de las clases que representan las entidades en la base de datos.

Las clases contenidas son:

- Manejador_Carga_Datos
- Manejador_Cliente
- Manejador_Devolucion
- Manejador_Ensamble_Pieza
- Manejador_Factura

- Manejador_Mueble
- Manejador_Mueble_Ensamblado
- Manejador_Pieza
- Manejador_Usuario

Las clases estan constituidas con los siguientes metodos (recalcando que cada manejador implementa el metodo de acuerdo a su necesidad, utilizando sobre escritura de metodos)

- *añadir*: Permite añadir una tupla a la base de datos.
- *select_all*: Permite obtener todas las tuplas de la entidad.
- *Select*: permite obtener tuplas de la entidad dependiendo del valor que se necesite.
- *Update*: permite actualizar (luego de haber modificado) alguna tupla.
- *Delete*: permite eliminar tuplas del sistema.
- *select_especial*: permite obtener registros de una entidad por medio de un campo en concreto.

PAQUETE SERVLET

Este paquete contiene los servlets necesarios para el Inicio/Fin de sesion y la validacion para el inicio.

Cerrar_Sesion: solamente se sobrescribio el metodo doGet, permitiendo cerrar la sesion del usuario.

Inicio_Servlet: solamente se sobrescribio el emtodo doGet que es utilizado para la redireccion.

Validacion: solamente se sobrescribio el metodo doGet que es utilizado para la validacion del usuario y la redireccion del mismo.

PAQUETE SERVLET ADMIN

Contiene los servlets utilizados por el usuario de tipo administrador.

Carga_Datos: se sobrescribio el metodo doPost para poder subir el archivo de carga.

Creacion_Usuario: se sobrescribio el metodo doPost que permite la verificacion y creacion de un nuevo usuario en el sistema.

Ensamble_Pieza: se sobrescribio el metodo doPost para la verificacion y la asignacion de piezas a algun mueble.

Modificacion_Usuario: se sobrescribio el emtodo doPost para la modificacion de los usuarios del sistema.

N_Mueble: se sobrescribio el metodo doPost para la creacion y verificacion de un nuevo mueble que podra ser ensamblado en la fabrica.

PAQUETE SERVLET FABRICA

Contiene los servlets que seran necesarios para el usuario de fabrica.

C_Pieza:

- *doPost:* se sobrescribio el metodo para la creacion y verificacion de una nueva pieza.
- *Response:* este metodo es el encargarlo de mostrarle al usuario una respuesta a la solicitud realizada.

D_Pieza:

- *doPost:* se sobrescribio el metodo para la eliminacion de una pieza.
- *Response:* este metodo es el encargarlo de mostrarle al usuario una respuesta a la solicitud realizada.

Ensamblar:

- *doPost:* se sobrescribio el metodo para la creacion y/o verificacion para ensamblar un nuevo mueble.
- *Response:* este metodo es el encargarlo de mostrarle al usuario una respuesta a la solicitud realizada.

M_Pieza:

- *doPost:* se sobrescribio el metodo para la modificacion de una nueva pieza.
- *Response:* este metodo es el encargarlo de mostrarle al usuario una respuesta a la solicitud realizada.

PAQUETE SERVLET VENTA

Contiene todos los servlets que serán necesarios para las opciones del usuario.

Nueva_Venta:

- *doPost:* se sobrescribio el metodo para verificar que el cliente al cual se le esta vendiendo ya esta registrado en el sistema.
- *Response:* este metodo es el encargarlo de mostrarle al usuario una respuesta a la solicitud realizada.

Nuevo_Cliente:

- *doPost:* se sobrescribio el metodo para la verificacion y creacion de un nuevo cliente.
- *Response:* este metodo es el encargarlo de mostrarle al usuario una respuesta a la solicitud realizada.

Reintegro_Venta:

- *doPost:* se sobrescribio el metodo para verificacion y registro de una devolucion.
- *Response:* este metodo es el encargarlo de mostrarle al usuario una respuesta a la solicitud realizada.

Venta_Final:

- *doPost:* se sobrescribio el metodo para la verificacion y creacion de una nueva venta.
- *Response:* este metodo es el encargarlo de mostrarle al usuario una respuesta a la solicitud realizada.