

Análisis del Proyecto

1. Requisitos funcionales del sistema.

- Crear ubicaciones.
- Crear categorías.
- Crear productos.
- Crear empleados.
- Crear clientes o proveedores.
- Gestionar compras.
- Gestionar ventas.
- Gestionar aumentos de inventarios.
- Gestionar salidas de inventarios.
- Guardar información de transacciones.
- Ver información de productos.
- Ver productos de una categoría.
- Ver productos de una ubicación.
- Ver información de un empleado.
- Ver información de un cliente o proveedor.
- Autenticar empleado y asignar capacidad de uso según el tipo.

2. Propósitos de clases.

- Categoría: Tendrá las funciones para filtrar productos que pertenecen a esta.
- Cliente_Proveedor: Permitirá reconocer de donde viene una materia prima o a quién le vendo un producto.
- DatabaseConnector: Permite realizar conexiones y consultas a la base de datos.
- Empleado: Permite registrar quien hace las transacciones, también se usará para restringir los menús.
- Gestion: Permite crear las transacciones que se realizan en los inventarios.
- Producto: Es aquella que simula el comportamiento de los distintos productos que se tendrán.
- Ubicacion: Tendrá las funciones para filtrar productos que pertenecen a esta.

3. Propósitos de atributos.

Clase Categoria:

- id: int: Almacena el identificador de la categoría.
- nombre: String: Almacena el nombre de la categoría.

Clase Cliente Proveedor:

- id: int: String: Almacena el identificador del proveedor.
- nombre: String: Almacena el nombre de la empresa.
- tipo: String: Almacena si es cliente/proveedor/la empresa.

Clase DatabaseConnector:

- URL: String: Almacena el url de la base de datos.
- USER: String: Almacena el nombre del usuario.
- PASSWORD: String: Almacena la contraseña del usuario.

Clase Empleado:

- id: int: Almacena el identificador del empleado.
- nombre: String: Almacena el nombre del empleado.
- tipo: String: Almacena el tipo de empleado.
- contrasenia: String: Almacena la contraseña del empleado.
- despedido: boolean: Almacena si el empleado ha sido despedido o no.

Clase Gestion:

- id: int: Almacena el identificador de la gestión.
- tipo: String: Almacena el tipo de gestión.
- fecha: LocalDateTime: Almacena la fecha y hora de la gestión.
- descripcion: String: Almacena la descripción de la gestión.
- identificadores: ArrayList<Integer>: Almacena los identificadores de los productos que se gestionarán.
- cantidades: ArrayList<Float>: Almacena las cantidades de los productos que se gestionarán.
- id_empleado: int: Almacena el identificador del empleado que realizó la gestión.
- id_c_p: int: Almacena el identificador del proveedor o cliente.

Clase Producto:

- id: int Almacena el identificador del producto.
- nombre: String: Almacena el nombre del producto.

- tipo: String: Almacena el tipo de producto.
- cantidad: float: Almacena la cantidad que posee el producto.
- dimension: String: Almacena la dimensión del producto.
- id_ubicacion: int: Almacena la ubicación del producto.
- id_categoria: int: Almacena el identificador de la categoría.
- descontinuado: boolean: Almacena un valor que indica si el producto está descontinuado o no.

Clase Ubicacion:

- id: int: Almacena el identificador de la ubicación.
- nombre: String: Almacena el Nombre de la ubicación.

4. Propósitos de métodos.

Clase Categoria:

- Setters
- Getters
- Existeld(): boolean: Verifica si la categoría existe en la base de datos.
- InsertarFila(): void: Inserta una fila en la tabla CATEGORIAS.
- ModificarFila(): void: Modifica una fila en la tabla CATEGORIAS.
- toString(): String: Información importante de la categoría.
- Informacion(): String: Información detallada de la categoría.
- Categoria(id: int, nombre: String): Constructor del objeto.
- Categoria(id: int): Constructor del objeto.
- Categoria(nombre: String): Constructor del objeto.

Clase Cliente_Proveedor:

- Setters
- Getters
- Existeld(): boolean: Verifica si el cliente o proveedor existe en la base de datos.
- InsertarFila(): void: Inserta una fila en la tabla CLIENTES_PROVEEDORES.
- ModificarFila(): void: Modifica una fila en la tabla CLIENTES_PROVEEDORES.
- Informacion(): String: Información detallada del cliente o proveedor.
- Cliente_Proveedor(nombre: String, tipo: String): Constructor del objeto.
- Cliente_Proveedor(id: int): Constructor del objeto.

Clase DatabaseConnector:

- connect(): Connection: Abre la conexión con la base de datos.
- close(connection: Connection): void: Cierra la conexión de la base de datos.
- executeQuery(query: String): List<Map<String, Object>>: Ejecuta una consulta y devuelve los resultados.
- executeUpdate(sql: String): void: Ejecuta una instrucción la cual permite insertar o modificar filas de alguna tabla en la base de datos.

Clase Empleado:

- Setters
- Getters
- CorroborarContrasenia(String contrasenia2): boolean: Corrobora que una contraseña ingresada sea igual a la registrada.
- Existeld(): boolean: Verifica si el empleado existe en la base de datos.
- InsertarFila(): void: Inserta una fila en la tabla EMPLEADOS.
- ModificarFila(): void: Modifica una fila en la tabla EMPLEADOS.
- Informacion(): String: Información detallada del cliente o proveedor.
- Empleado(nombre: String, tipo: String, contrasenia: String): Constructor del objeto.
- Empleado(id: int): Constructor del objeto.

Clase Gestion:

- InsertarFilaTransaccion(): void: Inserta una o más filas a la tabla TRANSACCIONES.
- InsertarFilaGestion(): void: Inserta una fila a la tabla GESTIONES.
- ToString(ArrayList<Producto> productos): String: Información de la gestión.
- Gestion(id: int, tipo: String, productos: ArrayList<Producto>, cantidades: ArrayList<Float>, empleado: Empleado, id_c_p: int, descripcion: String).

Clase Producto:

- Setters
- Getters
- CorroborarSalida(cantidad_descontada: float): boolean: Corrobora si se puede descontar una cantidad de producto.
- RebajarSalida(cantidad_descontada: float): void: Disminuye la cantidad del producto.
- AumentarExistencias(cantidad_aumentar: float): void: Aumenta la cantidad del producto.
- Existeld(): boolean: Verifica si el producto existe en la base de datos.

- InsertarFila(): void: Inserta una fila en la tabla PRODUCTOS.
- ModificarFila(): void: Modifica una fila en la tabla PRODUCTOS.
- Informacion(): String: Información detallada del producto.
- Producto(nombre: String, tipo: String, cantidad: float, dimension: String, id_ubicacion: int, id_categoria: int): Constructor del objeto.
- Producto(id: int, nombre: String, tipo: String, cantidad: float, dimension: String): Constructor del objeto.
- Producto(id: int): Constructor del objeto.

Clase Ubicacion:

- Setters
- Getters
- Existeld(): boolean: Verifica si la ubicación existe en la base de datos.
- InsertarFila(): void: Inserta una fila en la tabla UBICACIONES.
- ModificarFila(): void: Modifica una fila en la tabla UBICACIONES.
- toString(): String: Información importante de la ubicación.
- Informacion(): String: Información detallada de la ubicación.
- Ubicacion(id: int, nombre: String): Constructor del objeto.
- Ubicacion(id: int): Constructor del objeto.
- Ubicacion(nombre: String): Constructor del objeto.