

**SISTEMA DE GESTIÓN Y
CONTROL DE INVENTARIO, DE
EGRESOS E INGRESOS Y DE
SERVICIOS DE DE UN NEGOCIO
QUE OFRECE UNA AMPLIA
GAMA DE ACCESORIOS PARA
TELÉFONOS.
VERSIÓN BETA**

ÍNDICE

1. Introducción

- 1.1. Propósito
- 1.2. Alcance
- 1.3. Referencias

2. Descripción global

- 2.1. Perspectiva del producto
- 2.2. Funciones del producto
- 2.3. Características del usuario

3. Los requisitos específicos

1. Introducción

1.1. Propósito

El propósito de elaborar este software es atacar los problemas básicos de gestión y control que puede llegar a tener cualquier tipo de negocio, en específico, este momento nos encontramos con un cliente el cual ofrece una amplia variedad de artículos para teléfonos celulares y servicios.

1.2. Alcance

Brindar la administración correcta de los recursos, maximizar progresivamente la ganancia y mantener el control sobre los empleados: supervisión y correcto registro de servicios brindados.

1.3. Referencias

Estudio y valoración de mercado actual, trato con los proveedores, encuestas continuas de calidad y servicios, evaluaciones del rendimiento laboral de los empleados.

2. Descripción global

2.1. Perspectiva del producto

El sistema que engloba nuestro producto es un sistema de administración de empresas: Inventario, control laboral, pagos (nóminas), estudio de mercado, trato con proveedores, presentación y venta de producto, encuestas de calidad y posicionamiento / competencia en el mercado. Por lo tanto nuestro producto toma como planilla el sistema al que pertenece pero nos especializamos en el área de: Inventario, pagos (nóminas), el estudio de mercado, trato con proveedores y competencia de mercado. Dentro de las restricciones que abarca nuestro software son:

1. Tenemos la interfaz de inicio, usuario: administrador, registro/ dar de baja de usuario: trabajador, menú, disponibilidad de inventario, análisis de venta, registro de ventas proveedores, gastos, rendimiento.
2. Usuario: Administrador; interfaz de inicio: control completo a la información, permisos de edición y registro de movimientos.
3. Usuarios: Trabajadores; interfaz de menú: registros de venta, disponibilidad en inventario.
4. Vamos a trabajar desde el hardware de Apple y Android, compatible con la interfaz de usuario: administrador/ trabajador.
5. La interfaz de software a usar son: Android Studio, MySQL WorkBench
6. Por definir
7. Por definir / profundizar.

2.2. Funciones del producto

El producto a grandes rasgos realizará la administración óptima de nuestro cliente (Vendedor diverso de artículos para equipos celulares y servicios relacionados a los mismos) para una actualización tecnológica. Y lo hace a través de cuenta de administrador, cuenta de trabajador, inventario y registro de ventas.

2.3. Características del usuario

El nivel educativo del usuario es educación básica primaria, tiene 2 años con el negocio de celulares y es técnico en reparación de software (desbloques, libera compañías, flexeos, reiniciar teléfonos).

3. Los requisitos específicos

3.1. Pantalla de inicio

3.1.1. Mostrar animación

3.1.1.1. Entrada: Null

3.1.1.2. Salida: Operación exitosa: boolean

3.1.1.3. Proceso: Mostrar en pantalla una animación representativa del negocio.

3.2. Ingresar

3.2.1. Entrada: Usuario: administrador (string)

3.2.2. Salida: Registro: nuevo_administrador (string)

3.2.3. Entrada: Usuario: usernameT(int)

3.2.4. Salida: Worker_profile

3.2.5. Proceso “Ingresar”: If Usuario == administrador (string) dirigir admin_profile. else usernameT(int) dirigir worker_profile.

3.3. Administrador

3.3.1. Admin_Profile

3.3.1.1. Entrada: full_name: string

3.3.1.2. Salida: string(full_name)

3.3.1.3. Entrada: email: string

3.3.1.4. Salida: string (email)

3.3.1.5. Entrada: Password: (string)

- 3.3.1.6. Salida: if admin_profile == (mostrar contraseña) string(password) else string (*****)
- 3.3.1.7. Entrada: cellphone: data_int
- 3.3.1.8. Salida: data_int (cellphone)
- 3.3.1.9. Proceso “Guardar”: El usuario “Administrador” personaliza el perfil, cambia la contraseña y da de alta los medios a los que se le notificarán cualquier movimiento que se lleve a cabo, de igual manera serán los medios para recuperar contraseña en caso de olvidarla y se guardan.

3.3.2. Registro de trabajadores

- 3.3.2.1. Entrada: Nombre de usuario, contraseña del usuario: usernameT (int), passwordT (string)
- 3.3.2.2. Salida: Transacción exitosa : boolean
- 3.3.2.3. Proceso: Dar de alta en una tabla llamada trabajadores de la base de datos una tupla con los datos del trabajador.

3.3.3. Resumen de transacciones

- 3.3.3.1. Entrada: Rango de fechas: string(día, semana, mes)
- 3.3.3.2. Salida: Operación exitosa, botón de regresar:(boolean)
- 3.3.3.3. Proceso: Realizar una consulta a la base de datos sobre cuáles fueron las transacciones realizadas en un determinado día. Mostrar en una tabla ordenada por fecha los atributos: fecha y hora, artículos, cantidad de artículos vendidos, y total, al final de la tabla un total de las transacciones.

3.3.4. Los productos más vendidos

- 3.3.4.1. Entrada: Número de productos, Rango de fechas: int(3,5,10), string(día, semana, mes)
- 3.3.4.2. Salida: Operación exitosa: (boolean)
- 3.3.4.3. Proceso: Realizar una consulta a la base de datos sobre los no_prod más vendidos en un rango de tiempo hasta la fecha, donde no_prod está restringido a ciertos valores y mostrar en pantalla la tabla con el nombre del producto.

3.3.5. Los productos menos vendidos

- 3.3.5.1. Entrada: Número de productos, Rango de fechas: int(3,5,10),string(día, semana, mes)
- 3.3.5.2. Salida: Operación exitosa: boolean
- 3.3.5.3. Proceso: Realizar una consulta a la base de datos sobre los no_prod menos vendidos, donde no_prod está restringido a ciertos valores y mostrar en pantalla la tabla con el nombre del producto.

3.3.6. Los productos que se piden pero no se venden

- 3.3.6.1. Entrada: null
- 3.3.6.2. Salida: Operación exitosa, botón de regresar: boolean
- 3.3.6.3. Proceso: Realizar una consulta a la base de datos en la tabla productos que tengan la característica de que han sido pedidos pero no se venden, mostrar en una tabla los resultados.

3.3.7. Eliminar producto que piden y no se vende

- 3.3.7.1. Entrada: Id del producto: int
- 3.3.7.2. Salida: Operación exitosa: boolean

- 3.3.7.3. Proceso: Eliminar de la base de datos el producto con el id indicado de los productos que se piden pero no se venden.

3.3.8. Promedio de ventas por producto

- 3.3.8.1. Entrada: Id del producto, Rango de tiempo: int, string(día,semana,mes)
- 3.3.8.2. Salida:Operación exitosa: boolean
- 3.3.8.3. Proceso: De las transacciones realizadas en el producto ingresado, agrupar y sumar por mes, día o semana y realizar el promedio de estos valores. Mostrar en una tabla.

3.3.9. Ganancias netas

- 3.3.9.1. Entrada: Rango de tiempo: string (mes, día, semana).
- 3.3.9.2. Salida: Tabla con ganancias netas con las ganancias y las inversiones
- 3.3.9.3. Proceso: Realizar una consulta a la base de datos sobre las transacciones realizadas en un rango de tiempo y calcular inversiones y entradas de dinero. Resumir los resultados obtenidos en una tabla y mostrarla en pantalla.

3.3.10. Razón de ganancias por producto

- 3.3.10.1. Entrada: Rango de tiempo: string (mes, día, semana)
- 3.3.10.2. Salida: Ganancia en porcentaje respecto a la inversión.
- 3.3.10.3. Proceso: Realizar una consulta a la base de datos sobre las transacciones realizadas en un rango de tiempo y calcular la razón de ganancias netas sobre inversiones. Mostrar los resultados resumidos en una tabla.

3.3.11. Registrar Proveedores

- 3.3.11.1. Entrada: registro de proveedores: full_namesP (string), alias_name (string)
- 3.3.11.2. Salida: full_namesP (string), alias_name(string)
- 3.3.11.3. Entrada: dirección : addres_P(string)
- 3.3.11.4. Salida: addres_P(string)
- 3.3.11.5. Entrada: Celular de contacto: contact_cellphone(int)
- 3.3.11.6. Proceso “Guardar”: Se da de alta los proveedores con los que se está trabajando y a su vez los medios por los cuales contactar. Se muestra en pantalla.

3.3.12. Productos que compra con ellos

- 3.3.12.1. Entrada: Nombre del vendedor, nombre del producto, costo del producto: string, string, float
- 3.3.12.2. Salida: Operación exitosa: boolean
- 3.3.12.3. Proceso: Insertar una tupla de valores en una tabla de productos con los datos del producto y relacionarlo con el proveedor .

3.3.13. Dar de alta productos

- 3.3.13.1. Entrada: Nombre del producto, marca, costo: string, string, float
- 3.3.13.2. Salida: Operación exitosa: boolean
- 3.3.13.3. Proceso: Insertar en una tabla productos, los datos del producto que se quiere dar de alta.

3.3.14. Dar de baja de productos

- 3.3.14.1. Entrada: Nombre del producto, o id: int
- 3.3.14.2. Salida: Operación exitosa: boolean

- 3.3.14.3. Proceso: Eliminar de las relaciones que tenga la tupla del producto a eliminar de la tabla productos y después eliminar la tupla misma.

3.3.15. Ingresar nuevos productos al inventario

- 3.3.15.1. Entrada: id del producto, cantidad de nuevos productos, alerta de existencia: int,int,int
- 3.3.15.2. Salida: Operación exitosa: boolean
- 3.3.15.3. Proceso: Realizar una actualización o update a la tabla de productos en la tupla del producto.

3.3.16. Aceptar o rechazar solicitud de alta de productos

- 3.3.16.1. Entrada: Aceptar o rechazar: boolean
- 3.3.16.2. Salida: Operación exitosa: boolean
- 3.3.16.3. Proceso: Realizar un commit o un rollback a los cambios solicitados por el trabajador según se quiera o no dar de alta los cambios. Si la entrada tiene valor verdadero, true o 1 entonces se realizará un commit, si tiene un valor falso, false o 0 entonces se realizará un rollback.

3.3.17. Alerta de existencia de productos

- 3.3.17.1. Entrada: null
- 3.3.17.2. Salida: null
- 3.3.17.3. Proceso: Realizar periódicamente verificaciones de los productos del inventario y buscar aquellos cuya existencia sea menor o igual a la cantidad asigna como alerta por el administrador para avisar que están cerca de agotarse.

3.3.18. Modificar datos de productos

- 3.3.18.1. Entrada: Id del producto: int
- 3.3.18.2. Salida: Operación exitosa:boolean
- 3.3.18.3. Proceso: Por medio de campos de texto solicitar los nuevos datos del producto, siendo capaz de modificar solo aquellos que no pongan en riesgo la integridad de la base de datos, al dar click en modificar debemos realizar un update sobre los datos que obtenemos asegurándonos que todos los datos son congruentes con el modelo del negocio.

3.3.19. Modificar datos de trabajadores

- 3.3.19.1. Entrada: Id trabajador: int
- 3.3.19.2. Salida: Operación exitosa: boolean
- 3.3.19.3. Proceso: Por medio de campos de texto solicitar nuevos datos del trabajador permitiendo modificar sólo aquellos cambios que no pongan en riesgo la integridad de la base de datos y asegurándonos que todos los datos son congruentes.

3.4. Trabajador

3.4.1. Existencia del producto

- 3.4.1.1. Entrada: Id del producto: int
- 3.4.1.2. Salida: Cantidad del producto: int
- 3.4.1.3. Proceso: Realizar una consulta a la base de datos de la cantidad que existe en inventario del producto con cierto id, en caso de que no esté en existencia regresará 0 y deberá existir la opción de agregar a productos que se piden pero no se tienen.

3.4.2. Detalles del producto

- 3.4.2.1. Entrada: Id del producto: int
- 3.4.2.2. Salida: Operación exitosa: boolean
- 3.4.2.3. Proceso: Mostrar de forma amigable y ordenada los datos del producto que le interesan al trabajador como una descripción, imagen, nombre del producto, existencia de este etc.

3.4.3. Ejecutar venta

- 3.4.3.1. Entrada: Id del producto comprado, cantidad del producto comprado, y de la compra: int, int
- 3.4.3.2. Salida: Operación exitosa: boolean
- 3.4.3.3. Proceso: Insertar en la tabla transacciones el id de la transacción y en la tabla transacción-productos, los productos comprados juntos con la cantidad, actualizar la existencia de los productos.

3.4.4. Productos que se piden, pero no se tienen

- 3.4.4.1. Entrada: Nombre del producto: string
- 3.4.4.2. Salida: Operación exitosa: boolean
- 3.4.4.3. Proceso: Insertar en la base de datos un producto que se pide con los datos del producto, en caso de que haya sido pedido previamente se aumentará el número de veces que se han pedido en otro caso, se insertará una nueva tupla en la tabla y se inicializará las veces que se ha pedido con 1.

3.4.5. Agregar producto que se pide pero no se vende

- 3.4.5.1. Entrada: Nombre del producto: string
- 3.4.5.2. Salida: Operación exitosa: boolean

3.4.5.3. Proceso: Buscar en la base de datos en los productos que se piden pero no se venden la existencia de un producto, en caso de que no exista, deberemos insertar una nueva tupla con los datos del producto, en caso de que exista, deberemos aumentar la cantidad de veces que se ha pedido.

3.4.6. Ingresar al inventario nueva mercancia

3.4.6.1. Entrada: Id del producto, cantidad a ingresar: int, int

3.4.6.2. Salida: Operación exitosa: boolean

3.4.6.3. Proceso: Realizar un update a la tabla de productos donde aumente la existencia de los productos necesarios.

3.4.7. Solicitar alta de nuevos productos

3.4.7.1. Entrada: Nombre del producto, marca, costo: string, string, float

3.4.7.2. Salida: Operación exitosa: boolean

3.4.7.3. Proceso: Insertar en una tabla productos, los datos del producto que se quiera solicitar dar de alta al administrador del negocio.

3.5. Tablas de resumen

3.5.1. Ordenamiento por campo

3.5.1.1. Entrada: Tabla, campo a ordenar: Table, string

3.5.1.2. Salida: Tabla ordenada por cierto campo: Table

3.5.1.3. Proceso: Sobre la tabla ordenar las tuplas en orden alfanumérico por determinado campo.

3.5.2. Detalles del producto

3.5.2.1. Entrada: Id del producto: int

3.5.2.2. Salida: Operación exitosa: boolean

3.5.2.3. Proceso: Mostrar de forma ordenada y elegante los detalles del producto con id ingresado.