



Universidad Católica de Honduras

“Nuestra Señora Reina De La Paz”

Campus Jesús Sacramentado

Informe de Makeupbar System

Asignatura:

Desarrollo De Software

Catedrático:

Ing. Carlos Flores

Integrantes:

Gustavo Elías Meza 1001199900273

Dina Lean Guerrero 1201200100768

Yovany Hernández García 1306199800185

Wilmer Josué García 1006199801227

Cesar Ariel Mejía 1201199900871

2020

Índice

Índice	2
Introducción.....	3
Objetivos.....	4
Manual	5
Diccionario de datos	9
Tabla Inventario.....	9
Tabla Producto.....	9
Tabla Empleado.....	9
Tabla Producto General	10
Tabla Maquillaje.....	10
Tabla Envío	11
Tabla Compra	11
Tabla Cliente.....	12
Tabla Instagram	12
Tabla Servicio Entrega	13
Tabla Proveedor.....	13
Tabla Detalle De Venta	14
Tabla Venta.....	14
Tabla Factura	15
Constraints.....	16
Triggers.....	18
Conclusiones.....	19

Introducción

Makeupbar es una empresa de ventas de maquillaje y productos de maquillaje en relación con lo la misma.

Hemos desarrollado un sistema para optimizar procesos que se realizan en la empresa con ciertos grados de dificultad analizando y buscando aquellos detalles por mejorar de esta manera esperando agradar la vista de las personas encargadas de su uso.

En el presente mostraremos un poco sobre las ventanas que contendrá el sistema del mismo modo detallando ciertos puntos de relevancia que serán de gran ayuda para el correcto uso del sistema.

El sistema está enfocado en la entrada de los empleados al sistema para que puedan realizar sus tareas de una forma más personalizada y sencilla.

Objetivos

General

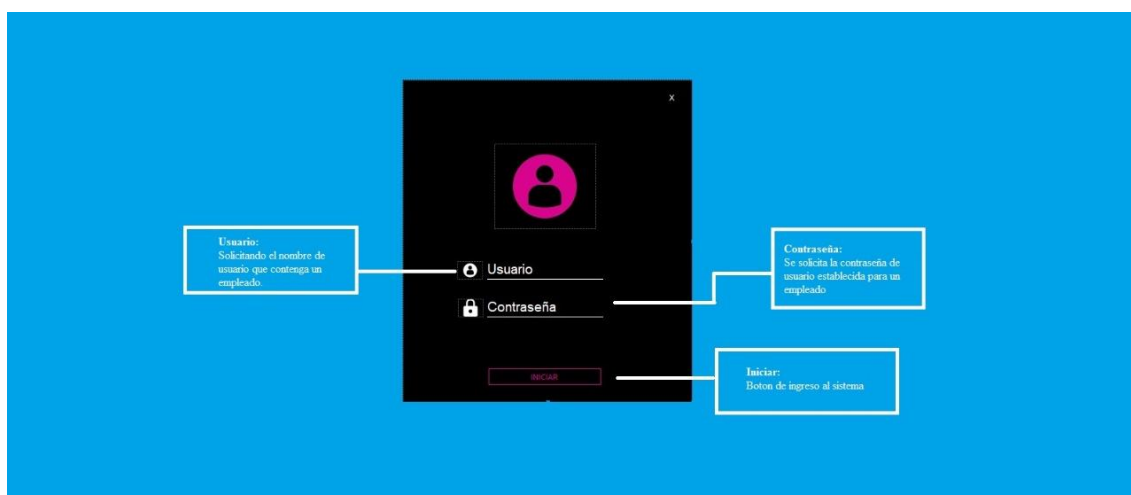
Desarrollar un sistema que cumpla con las necesidades solicitadas por los empleados y dueño de la empresa

Específicos

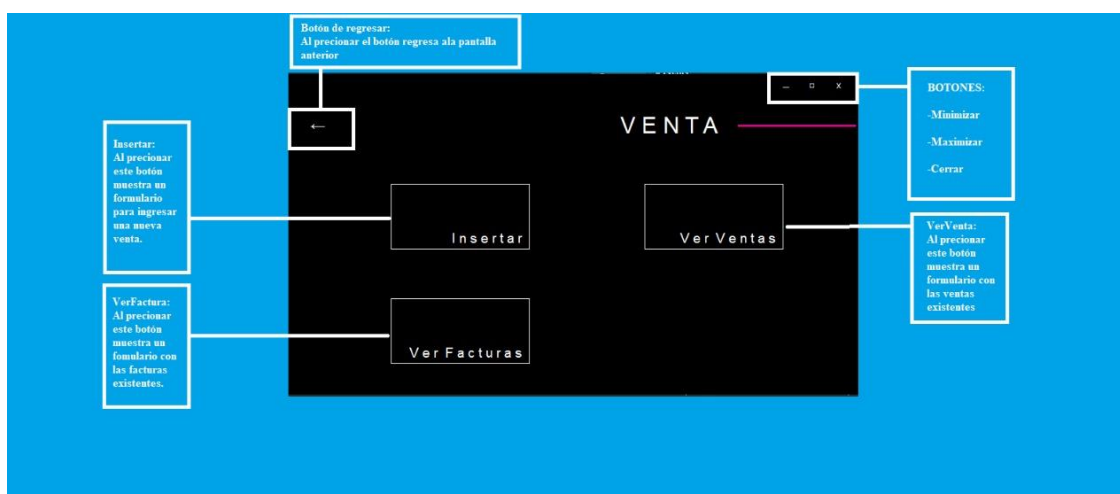
- Generar una interfaz grafica minimalista y que cumpla con lo requerido por la empresa.
- Funcionamiento sencillo en sus operaciones, eliminando la complejidad de su software actual.

Manual


Ingreso del empleado al sistema.



Módulos principales del programa para las distintas operaciones.



Visualización de datos



The screenshot shows a window titled "Datos Cliente" with a table containing the following data:

IdCliente	Nombre	Correo	Telefono
0000-0000-0000	Default	default@corre.com	0000-0000

A callout box on the right states: "Muestra los datos del cliente que ha sido registrado previamente".

Inserción de datos



The screenshot shows a window titled "Insertar Cliente" with a form for entering client data. The form includes the following fields:

- IdCliente
- Cumpleaños (date picker: martes 14 de abril de 2020)
- Nombre
- Ciudad
- Correo
- Tono de Base
- Teléfono
- Tono de Polvo
- Perfil Instagram
- Tipo de Cutie

A callout box on the right states: "Formulario de ingreso de datos. Ingrese todos los datos del cliente. Perfil instagram puede quedar vacío, no todos poseen uno.".

At the bottom of the form is a button labeled "Insertar". A callout box below the button states: "Botón Insertar; Al tener los campos llenos puede presionar el botón para que sean añadidos a la base de datos.".

Modificación de un registro.

The screenshot shows a form titled "ModificarEmpleado" with a blue header. The form contains a list of employees on the left, with the first one highlighted in red. To the right, there are input fields for "Identidad", "Usuario", "Contraseña", and "Rol". Below these fields are two buttons: "Eliminar" and "Modificar".

Selected Item
Aquí seleccionamos el registro que queremos editar sus datos.

Carga de Datos
Se cargan todos los registros dentro del registro seleccionado.

Elimina el ítem seleccionado.

Apertura al Modo Modificación
Coloca en rojo los campos que se pueden modificar y avilita la opción guardar.

Funcionamiento de la venta

The screenshot shows a form titled "InsertarVenta" with a blue header. The form contains a "Venta" section with "Cliente" and "Empleado" fields, and a "Total Venta" field. Below this is a "Detalles" section with "Venta", "Precio", "Factura", "Cantidad", "Producto", and "Descuento" fields. At the bottom, there are three buttons: "Limpiar", "Insertar", and "Cancelar".

Limpiar:
Iniciamos todos los campos a su valor por defecto.

Insertar:
Inserta el producto a la venta.

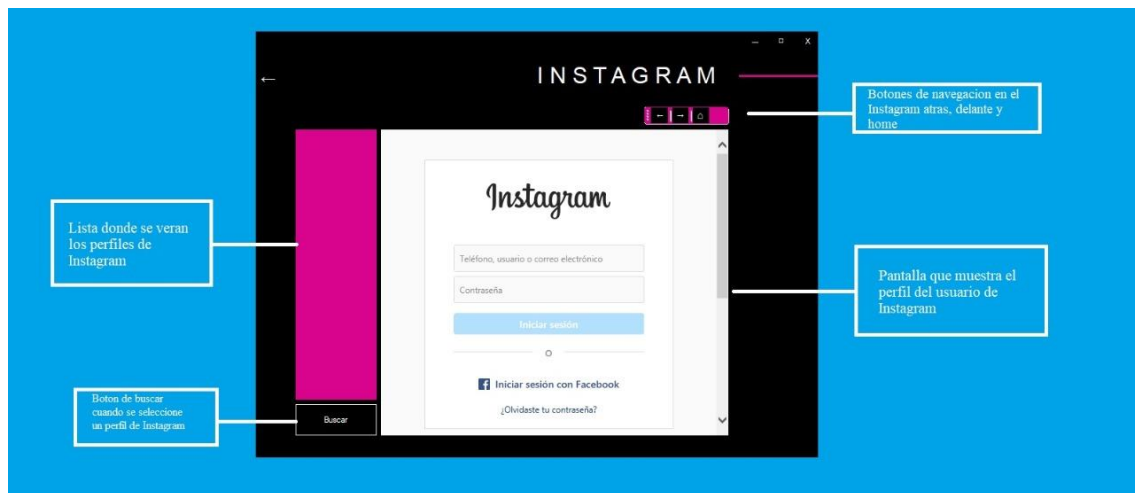
Tabla:
Muestra los productos de la compra actual.

Total Venta:
Muestra el total de la venta.

Venta:
Genera la venta y la factura que serán usadas.

Cancelar:
Cancela el ingreso de la venta.

Navegador web integrado para la conectividad a Instagram.



Diccionario de datos

Tabla Inventario

Esta tabla es tipo historial y es creada a partir de triggers desde la tabla de maquillaje o producto general.

Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idInventario	INT	✓	✓					✓		
precio	FLOAT								NULL	
idCodigoDeBarra	INT								NULL	
Cantidad	INT								NULL	

Tabla Producto

Tabla tipo historial creada a partir de triggers desde producto y maquillaje.

Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idProducto	INT	✓	✓					✓		
CodigoDeBarra	INT								NULL	
NombreDelProducto	VARCHAR(50)								NULL	
Precio	FLOAT								NULL	
Cantidad	INT								NULL	

Tabla Empleado

Se realiza un registro de cada uno del empleado o usuario que tendrán acceso al sistema, cabe destacar que se dejó un usuario por defecto.

Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idEmpleado	VARCHAR(15)	✓	✓							
Usuario	VARCHAR(50)		✓							
Contraseña	VARCHAR(50)		✓							
Rol	VARCHAR(50)		✓							

Tabla Producto General

producto general de la base de datos de makeupbar, teniendo como llave primaria IdCodigoDeBarra que es auto incremental que empieza desde 2001000000, utiliza una llave foránea que es idProveedor. Ningún campo puede estar vacío o ser nulo.

Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idCodigoDeBarra	INT	✓	✓					✓		
NombreProducto	VARCHAR(50)								NULL	
Marca	VARCHAR(50)								NULL	
PrecioUnitario	VARCHAR(50)								NULL	
Cantidad	VARCHAR(50)								NULL	
Descripcion	VARCHAR(100)								NULL	
idProveedor	INT								NULL	

Tabla Maquillaje

teniendo como llave primaria IdCodigoDeBarra que es auto incremental que empieza desde 1001000000, utiliza una llave foránea que es idProveedor. Ningún campo puede estar vacío o ser nulo.

Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idCodigoDeBarra	INT	✓	✓					✓		
NombreDelProducto	VARCHAR(60)								NULL	
Marca	VARCHAR(50)								NULL	
TonoNumero	VARCHAR(50)								NULL	
FechaDeExpiracion	DATE								NULL	
PrecioUnitario	VARCHAR(50)								NULL	
Cantidad	VARCHAR(50)								NULL	
Descripcion	VARCHAR(100)								NULL	
idProveedor	INT								NULL	

Tabla Envío

Base de datos de makeupbar, teniendo como llave primaria IdEnvio que es auto incremental, utiliza una llave foránea que es idServicoDeEntrega. Ningún campo puede estar vacío o ser nulo.

Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idEnvio	INT	✓	✓					✓		
idCliente	VARCHAR(15)		✓							
Direccion	VARCHAR(50)								NULL	
Telefono	VARCHAR(9)								NULL	
idServicioDeEntrega	INT		✓							

Tabla Compra

Se encarga de la creación de la tabla compra dentro de la base de datos makeupbar, teniendo como llave primaria IdCompra que es auto incremental, utiliza una llave foránea que es IdProveedor. Ningún campo puede estar vacío o ser nulo.

Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
IdCompra	INT	✓	✓					✓		
IDProveedor	INT		✓							
NombreDelProducto	VARCHAR(50)		✓							
Cantidad	INT		✓							
Costo	DOUBLE		✓							
Descripcion	VARCHAR(100)		✓							

Tabla Cliente

Crea una tabla de nombre cliente, la llave primaria es IdCliente que no es auto incremental, los campos que no pueden ser nulos son IdCliente y Nombre, se hace una inserción de un cliente por defecto.

Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
IdCliente	VARCHAR(15)	✓	✓							
Nombre	VARCHAR(50)		✓							
Correo	VARCHAR(50)								NULL	
Telefono	VARCHAR(9)								NULL	
PerfilInstagram	VARCHAR(30)								NULL	
Cumpleaños	DATETIME								NULL	
Ciudad	VARCHAR(50)								NULL	
TonoDeBase	VARCHAR(40)								NULL	
TonoDePolvo	VARCHAR(40)								NULL	
TipoDeCuties	VARCHAR(40)								NULL	

Tabla Instagram

Crea la tabla Instagram, llave primaria es IdInstagram que es autoincremental, en caso de no tener sé colocara null, posee id del Cliente y su respectivo usuario, además del link con el que se pueda acceder a su perfil mediante el navegador integrado en el sistema.

Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
IdInstagram	INT	✓	✓					✓		
IdCliente	VARCHAR(15)		✓							
Usuario	VARCHAR(50)								NULL	
URL	VARCHAR(100)								NULL	

Tabla Servicio Entrega

La tabla Servicio de Entrega se encarga de manejar datos relevantes con respecto a la misma teniendo como llave principal el id de la entrega siendo este auto incremental, además posee campos que muestran el nombre de la empresa, la persona que se contrató, el teléfono y su correo.

Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
IdServicioEntrega	INT	✓	✓					✓		
nombreEmpresa	VARCHAR(50)		✓							
nombreContrato	VARCHAR(50)		✓							
telefono	VARCHAR(9)		✓							
correo	VARCHAR(30)								NULL	

Tabla Proveedor

La tabla proveedor es de suma importancia ya que los artículos se obtienen por medio de un proveedor que abastece los inventarios de la empresa, en se manejaran datos de este como vienen siendo el nombre de la empresa que se asocia el proveedor, nombre del proveedor, su teléfono, su correo y su descripción. De esta manera se busca mantener un control de los proveedores y de esa forma facilitar su accesibilidad a ciertas operaciones.

Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
IdProveedor	INT	✓	✓					✓		
nombreEmpresa	VARCHAR(50)		✓							
nombreDelContrato	VARCHAR(50)		✓							
telefonoContacto	VARCHAR(9)		✓							
correo	VARCHAR(30)		✓							
descripcion	VARCHAR(50)		✓							

Tabla Detalle De Venta

Se encarga de la creación de la tabla DetalleDeVenta dentro de la base de datos makeupbar, teniendo como llave primaria idDetalleDeVenta que es auto incremental, utiliza como llaves foráneas idVenta, idFactura, idProducto. Ningún campo puede estar vacío o ser nulo.

Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idDetalleDeVenta	INT	✓	✓					✓		
idVenta	INT								NULL	
idFactura	INT								NULL	
idProducto	INT								NULL	
precio	DOUBLE								NULL	
Cantidad	INT								NULL	
Descuento	INT								NULL	
Total	DOUBLE								NULL	

Tabla Venta

Se encarga de la creación de la tabla Venta dentro de la base de datos makeupbar, teniendo como llave primaria idVenta que es auto incremental, utiliza como llaves foráneas idCliente y idEmpleado, el campo Fecha es autogenerable. Ningún campo puede estar vacío o ser nulo.

Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idVenta	INT	✓	✓					✓		
idCliente	VARCHAR(15)								NULL	
idEmpleado	VARCHAR(15)								NULL	
Fecha	DATETIME								CURRENT_TIMESTAMP()	

Tabla Factura

Se encarga de la creación de la tabla Factura dentro de la base de datos makeupbar, teniendo como llave primaria IdFactura que es auto incremental, utiliza como llaves foráneas idCliente y idEmpleado, el campo FechaActual es autogenerable Ningún campo puede estar vacío o ser nulo.

Column name	Data Type	PK	NN	UQ	BIN	UN	ZE	AI	Default	Comment
IdFactura	INT	✓	✓					✓		
FechaActual	DATETIME								CURRENT_TIMESTAMP()	
IdEmpleado	VARCHAR(15)		✓							
idCliente	VARCHAR(15)		✓							

Constraints

```
/* Restricciones para que los campos no esten vacíos en la tabla Cliente */
ALTER TABLE cliente
ADD CONSTRAINT CK_IdCliente_noVacío CHECK (LENGTH(TRIM(IdCliente)) > 0),
ADD CONSTRAINT CK_Nombre_noVacío CHECK (LENGTH(TRIM(Nombre)) > 0),
ADD CONSTRAINT CK_Correo_noVacío CHECK (LENGTH(TRIM(Correo)) > 0),
ADD CONSTRAINT CK_Telefono_noVacío CHECK ((LENGTH(TRIM(Telefono)) > 0)AND(LENGTH(TRIM(Telefono)) < 10)),
ADD CONSTRAINT CK_PerfilInstagram_noVacío CHECK (LENGTH(TRIM(PerfilInstagram)) > 0),
ADD CONSTRAINT CK_Ciudad_noVacío CHECK (LENGTH(TRIM(Ciudad)) > 0),
ADD CONSTRAINT CK_TonoDeBase_noVacío CHECK (LENGTH(TRIM(TonoDeBase)) > 0),
ADD CONSTRAINT CK_TonoDePolvo_noVacío CHECK (LENGTH(TRIM(TonoDePolvo)) > 0),
ADD CONSTRAINT CK_TipoDeCuties_noVacío CHECK (LENGTH(TRIM(TipoDeCuties)) > 0);

/* Máscara para validar el correo*/
ALTER TABLE cliente
ADD CONSTRAINT CK_validar_Formato_correo CHECK (Correo LIKE '%@%');

/* Restricción unique para que el id del cliente sea único */
ALTER TABLE cliente
ADD CONSTRAINT UQ_IdClienteUnico UNIQUE (IdCliente);
```

El primer segmento corresponde a los Constraints que verifican los campos y que ninguno de ellos este vacío al momento de insertar datos.

El siguiente constraint corresponde a la validación del correo electrónico que debe contener un @ para ser válido.

El último constraint corresponde al IdCliente que debe ser único debido a que no debe repetirse el Id, debe ser único para cada cliente que se registre.

```
/* Restricciones para que los campos no esten vacíos en la tabla Instagram */
ALTER TABLE instagram
ADD CONSTRAINT CK_IdCliente_instagram_noVacío CHECK ((LENGTH(TRIM(IdCliente)) > 0)),
ADD CONSTRAINT CK_Usuario_noVacío CHECK ((LENGTH(TRIM(Usuario)) > 0)),
ADD CONSTRAINT CK_URL_noVacío CHECK (LENGTH(TRIM(URL)) > 0);

/* Restricciones para que los campos no esten vacíos en la tabla Compra */
ALTER TABLE compra
ADD CONSTRAINT CK_NombreDelProducto_noVacío CHECK (LENGTH(TRIM(NombreDelProducto)) > 0),
ADD CONSTRAINT CK_Cantidad_noVacío CHECK ((Cantidad) > 0),
ADD CONSTRAINT CK_Costo_noVacío CHECK ((Costo) > 0),
ADD CONSTRAINT CK_Descripcion_noVacío CHECK (LENGTH(TRIM(Descripcion)) > 0);

/* Restricción unique para que el nombre del producto sea único */
ALTER TABLE compra
ADD CONSTRAINT UQ_NombredelProducto UNIQUE (NombreDelProducto);
```

El primer segmento corresponde a los Constraints en la tabla Instagram, de los cuales ninguno puede estar vacío al momento de ingresar datos en la tabla cliente.

El segundo segmento corresponde a la tabla compra de los cuales los campos no pueden estar vacíos, los que contienen números no pueden ser 0.

El último constraint corresponde al nombre del producto el cuál es único y no podrá repetirse.

Triggers

```
USE `makeupbar`$$
CREATE DEFINER = CURRENT_USER TRIGGER `makeupbar`.`cliente_AFTER_INSERT` AFTER INSERT ON `cliente` FOR EACH ROW
BEGIN
    insert into instagram(IdCliente, Usuario, URL)
    VALUES (new.IdCliente, new.PerfilInstagram, concat('www.instagram.com/', new.PerfilInstagram));
END$$

USE `makeupbar`$$
CREATE DEFINER = CURRENT_USER TRIGGER `makeupbar`.`cliente_AFTER_DELETE` AFTER DELETE ON `cliente` FOR EACH ROW
BEGIN
    DELETE FROM instagram
    WHERE instagram.IdCliente= old.IdCliente;
END$$
```

Se aplican dos triggers en la tabla cliente, ambos destinados a la tabla Instagram. El primero se crea a partir de la inserción de datos, para el URL se concatena el link de Instagram con el nombre del usuario, así se crea el link con acceso al usuario. En el navegador es más visible el enlace.

El segundo trigger tiene como funcionalidad eliminar el perfil de Instagram a partir del cliente que se ha eliminado.

Conclusiones

- El diseño y desarrollo del proyecto llamado Makeupbar System cumplió con lo requerido por la empresa, exceptuando el correo electrónico al llegar el cumpleaños de un cliente.
- Se agrego una de las operaciones base de la empresa, y es la comunicación del cliente por medio de la red social de Instagram, para tener un mejor control de los usuarios que visitan regularmente la empresa.
- El sistema presento las diversas ventajas para el usuario, una interfaz amigable con la que se puede interactuar de manera fácil para el usuario.