



Yasmedical

DISTRIBUIDORA DE INSUMOS MEDICOS

ELABORADO POR:
ALEJANDRO PADILLA

INDICE

Introducción	2
Objetivo	3
Situación problemática	4
Modelo de negocio, Diagrama ER	5
Descripción De Tablas	6
descripción De Objetos De Base De Datos	7
Tecnologías Aplicadas, Enlace Del Proyecto	8
Anexos.....	9

INTRODUCCION

Con los años, los modelos de negocios han evolucionado significativamente, con el crecimiento de este se han implementado nuevas herramientas para llegar a cabo una buena gestión y organización de manera más optima. Con la llegada de las bases de datos relaciones y no relacionales las empresas han podido almacenar de manera eficiente toda la información que esta genera a diario y crece proporcional o desproporcionalmente a medida que va evolucionando e incorporando más productos, servicios, etc. Este es el caso de Yasmedical, que ha consigo implementar una base de datos con MySQL en su sistema para almacenar de manera óptima y eficiente la información de sus proveedores, ventas y clientes, para tener un mejor control de este y, por lo tanto, una mejor recopilación de datos para resúmenes.

OBJETIVO

El objetivo de implementar esta funcionalidad es resguardar los datos importantes de manera segura y optima a largo plazo, para consultarlos constantemente y mediante el buen análisis de estos datos de los distintos tipos de departamentos dentro de la empresa (como lo pueden ser el departamento de ventas, departamento de recursos humanos, contabilidad y almacenes) crear resúmenes concretos periódicamente, y, por lo tanto, una mejor toma de decisiones sobre el futuro de la empresa.

SITUACION PROBLEMÁTICA

En este apartado mencionaremos los puntos clave que se tomaron en cuenta al momento de implementar este nuevo sistema, con el crecimiento de la empresa desde su fundación, se han inaugurado nuevos departamentos, nuevos productos y servicios. La generación diaria de datos ha crecido drásticamente los últimos años y hemos optado por incorporar dicha mejora para seguir avanzando acorde al crecimiento interno y completar todos los desafíos, resultado de cada una de las decisiones que se toman para ampliar cada vez más el negocio. Con esto, logramos tener un mayor control de inventario, mejor categorización de proveedores según el producto suministrado, como también identificar el tipo de producto más demandado, una mejor asignación de comisiones por cada factura que generen los vendedores y una estructuración de datos de ventas más sólida.

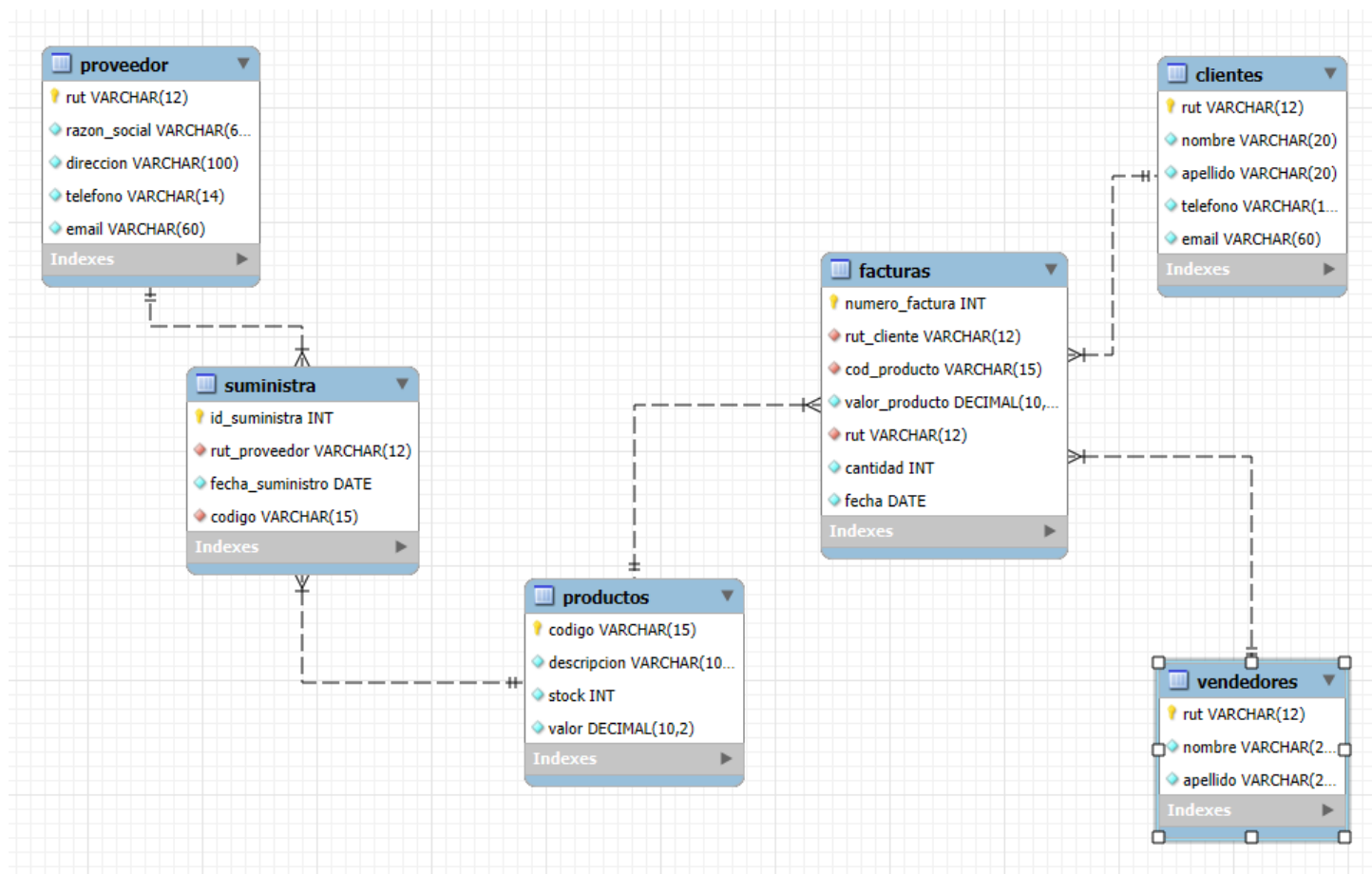
MODELO DE NEGOCIO

Este modelo de negocio consiste en un sistema de recepción, organización y distribución de insumos médicos a entidades públicas y privadas del sector salud.

La empresa cuenta con una cartera de **proveedores** que suministran la bodega principal donde son almacenados los **productos**, estos pueden ser suministrados por múltiples proveedores, los cuales serán gestionados por los **vendedores** quienes se encargarán de realizar las **ventas** a todos los clientes con su respectiva **factura**, cada vendedor tendrá asignado uno o más clientes para respetando así los bonos y comisiones por cada venta realizada y factura generada.

Con este esquema se pretende posicionarnos en uno de los mayores distribuidores de insumos médicos de Chile, con la posibilidad próximamente de exportar nuestros productos a distintos países.

DIAGRAMA ENTIDAD-RELACION



LISTADO DE TABLAS

Proveedor						
Desc	Información básica de los proveedores					
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT
PK	RUT	VARCHAR	12		X	
	RAZON_SOCIAL	VARCHAR	60	X		
	DIRECCION	VARCHAR	100	X		
	TELEFONO	VARCHAR	14	X		
	EMAIL	VARCHAR	60	X		

Productos						
Desc	Información básica de los productos					
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT
PK	CODIGO	VARCHAR	15		X	
	DESCRIPCION	VARCHAR	100	X		
	STOCK	INT		X		
	VALOR	DECIMAL	10, 2	X		

Vendedores						
Desc	Información básica de los vendedores					
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT
PK	RUT	VARCHAR	12		X	
	NOMBRE	VARCHAR	20	X		
	APELLIDO	VARCHAR	20	X		

Facturas						
Desc	Información básica de Las facturas generadas por vendedores a clientes					
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT
PK	N°_FACTURA	INT			X	AUTO_INCREMENT
FK	RUT_CLIENTE	VARCHAR	12	X		
FK	COD_PRODUCTO	VARCHAR	15	X		
	VALOR_PRODUCTO	DECIMAL	8. 2	X		
	RUT_VENDEDOR	VARCHAR	12	X		
	CANTIDAD	INT		X		
	FECHA	DATE		X		

Clientes						
Desc	Información básica de los clientes					
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT
PK	RUT	VARCHAR	12		X	
	NOMBRE	VARCHAR	20	X		
	APELLIDO	VARCHAR	20	X		
	TELEFONO	VARCHAR	14	X		
	EMAIL	VARCHAR	60	X		

Suministra						
Desc	Código y fecha de suministro de insumos					
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT
pk	ID_SUMINISTRA	INT				AUTO_INCREMENT
FK	RUT_PROVEEDOR	VARCHAR	12		x	
	FECHA_SUMINISTRO	DATE		x		
	CODIGO	VARCHAR	15			

DESCRIPCION DE OBJETOS DE BASE DE DATOS

Vistas.

1. **insumos_recibidos_2023 AS:** Vista para visualizar todos los insumos recibidos durante el año 2023.
2. **vista_productos_rango_precio:** vista de productos por rango de precio.
3. **total_ventas_vendedores:** vista para ver cantidad de ventas por vendedores en orden descendente.
4. **ProductosMasVendidos:** vista para visualizar los productos más vendidos desde su inicio
5. **ProveedoresMasSuministros:** vista de proveedores con mayor cantidad de suministros.
6. **ProductosVendidosPorMes:** vista de los productos más vendidos en un periodo de 12 meses, se puede especificar el año usando "WHERE Año = 2024"
7. **TotalFacturacionAnual:** vista para mostrar el total de facturación de la empresa año tras año desde su inicio.

Funciones.

1. **CalcularIVA:** función para calcular el IVA de una factura.
2. **ObtenerNombreCliente:** función para obtener el nombre de un cliente ingresando su Rut.

Stored Procedures.

1. **UpdateStock:** procedimiento para actualizar stock de productos, sumando la cantidad nueva con la cantidad actual.
2. **AddBill:** procedimiento para generar facturas, tomando el precio correspondiente del producto de su respectiva tabla.

Triggers.

1. **after_products_update:** trigger para auditar los cambios efectuados en la tabla "Productos", este trigger cuenta con tu tabla independiente para dar seguimiento a los cambios.
2. **before_facturas_insert:** trigger para evitar generar facturas si el producto requerido no cuenta con suficiente stock, enviando un mensaje al usuario con la siguiente descripción: "No hay suficiente stock para completar la factura."

TECNOLOGIAS APLICADAS

Para la implementación de este sistema de bases de datos en este proyecto se utilizó principalmente **MySQL**, creado por **Oracle**, una vez creada esta base, las tablas correspondientes a los departamentos, las inserciones de datos a cada uno de ellos y los demás objetos (vistas, funciones, procedimientos almacenados y triggers) son utilizadas por el lenguaje de programación **Python** con las siguientes librerías:

1. **mysql.connector** : controlador para conectar con la base de datos e ingresar consultas.
2. **Pandas**: para transformar las tablas obtenidas de MySQL en dataframes.
3. **Dotenv**: esta librería busca automáticamente el archivo **.env** con la dirección del host, usuario, contraseña y nombre de la base de datos, de lo contrario le pedirá que ingrese los datos manualmente a través de la consola.
4. **Dash**: framework para crear aplicaciones de datos, este a su vez depende de otras librerías preinstaladas como **flask**.
5. **Plotly Express**: viene instalado con **dash**, es una librería para crear gráficos interactivos de forma rápida y sencilla.
6. **Webbrowser**: se utiliza para buscar el navegador predeterminado en el sistema y renderizar todos los datos transformados en gráficos previamente.

Es sumamente requerido tener **Python** instalado, preferiblemente a partir de la versión 3.11. Para lanzar esta aplicación en Windows se utiliza el script **run.ps1** y/o **run.bat** adjunto en el proyecto, en caso de ser Linux se deja el script **Run.sh**. Si no tiene instalado **MySQL** en su ordenador no hay problema, se dejan adjuntas en formato **CSV** la base de datos como alternativa.

ENLACE DEL PROYECTO

<https://github.com/Alerjop3d/YasmedicalDashboard.git>

ANEXOS

Dashboard Yasmedical

