

# Final – Agustin Alvarez

## Descripción de la temática de la base de datos:

Mi idea fue hacer una base de datos relacionada a una agencia de seguros que opera desde internet, en la cual para poder ser cliente primero deberían registrarse en el sitio web, rellenar alguna información personal adicional y luego poder elegir entre un seguro de vivienda o vehículo. Deberán aclarar la información sobre el producto a asegurar para poder dar cotización sobre el mismo.

## Objetivos:

Diseñar una base de datos relación que sea fácil de manipular, sin saltarnos el protocolo de seguridad para manipular datos sensibles sin riesgos de fugas con la ayuda de las Functions, Stored Procedures y los Triggers para monitorear las tablas a medida que los datos ingresan y con las creaciones de vistas para facilitar la información al usuario.

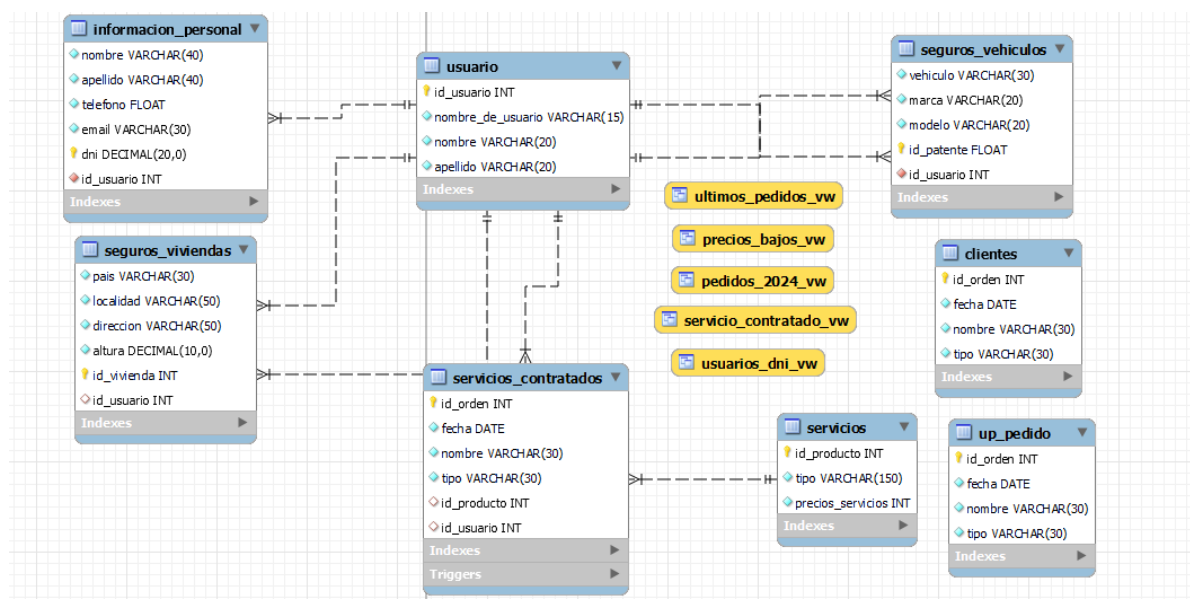
## Situación problemática:

Agencia aseguradora viene a solucionar el problema de los formularios tediosos llenos de campos por llenar. Buscando la forma de hacer el procedimiento mas simple para el usuario sin tantas vueltas.

## Modelo de Negocios:

Operando por el bienestar y tranquilidad de los usuarios ante los peligros inminentes de la vida, Agencia Aseguradora permite asegurar autos, motos y viviendas de forma que ante cualquier situación de perdida podamos darle al usuario una financiación adecuada al servicio contratado.

## Diagrama de Entidad - Relación (EDR):



## Listado de tablas que componen la base de datos:

USUARIO: Tabla que registran los usuarios.

Tablas	Campos	Detalles del Campo	PK	FK	Tipo De Dato
	Id_usuario	Identificador de cada usuario	PK		INT
Usuario	Nombre_de_usuario	Nombre del usuario			VARCHAR(15)
	Nombre	Nombre real del usuario			VARCHAR(20)
	Apellido	Apellido del usuario			VARCHAR(20)

SEGUROS VEHICULOS: Tabla que registra los distintos tipos de vehículos para asegurar.

Tablas	Campos	Detalles del Campo	PK	FK	Tipo De Dato
	Vehículo	Tipo de vehículo			VARCHAR(30)
Seguros Vehículos	Marca	Marca del vehículo			VARCHAR(20)
	Modelo	Modelo del vehículo			VARCHAR(20)
	Id_patente	Numero de patente	PK		FLOAT(10)
	Id_usuario	Identificador del usuario		FK	INT

SEGUROS VIVIENDAS: Tabla que registra las distintas viviendas para asegurar.

Tablas	Campos	Detalles del Campo	PK	FK	Tipo De Dato
	País	País en el que se encuentra la propiedad			VARCHAR(30)
	Localidad	Localidad en la que se encuentra la propiedad			VARCHAR(50)
Seguros Viviendas	Dirección	Dirección exacta de la propiedad			VARCHAR(50)
	Altura	Numero en el que se encuentra la propiedad			NUMERIC(10,0)
	Id_vivienda	Identificador de la vivienda	PK		INT
	Id_usuario	Identificador del usuario		FK	INT

INFORMACION PERSONAL: Tabla que registra la información personal de cada usuario registrado.

Tablas	Campos	Detalles del Campo	PK	FK	Tipo De Dato
	Nombre	Nombre real del usuario			VARCHAR(40)
	Apellido	Apellido del usuario			VARCHAR(40)
<b>Información Personal</b>	Teléfono	Teléfono del usuario			FLOAT
	Email	Dirección de correo del usuario			VARCHAR(30)
	Dni	Dni del usuario	PK		NUMERIC(20)
	Id_usuario	Identificador del usuario		FK	INT

CLIENTES: Tabla que registra a los clientes a medida que contratan servicios.

Tablas	Campos	Detalles del Campo	PK	FK	Tipo De Dato
	Id_orden	Identificador del pedido realizado	PK		INT
<b>Clientes</b>	Fecha	Fecha del pedido			DATE
	Nombre	Nombre del cliente			VARCHAR(30)
	Tipo	Tipo de servicio contratado			VARCHAR(30)

SERVICIOS: Tabla que registra las distintas cotizaciones de los servicios ofrecidos.

Tablas	Campos	Detalles del Campo	PK	FK	Tipo De Dato
	Id_producto	Identificador del servicio a vender	PK		INT
<b>Servicios</b>	Tipo	Tipo de servicio a vender			VARCHAR(150)
	Precios_servicios	Precio del servicio a vender			INT

SERVICIOS CONTRATADOS: Tabla que registra los servicios que se van contratando.

Tablas	Campos	Detalles del Campo	PK	FK	Tipo De Dato
	Id_orden	Identificador de cada orden solicitada	PK		INT
	Fecha	Fecha a la que se contrato el servicio			Date
<b>Servicios Contratados</b>	Nombre	Nombre del usuario que contrato el servicio			VARCHAR(30)
	Tipo	Tipo de servicio contratado			VARCHAR(30)
	Id_producto	Identificador del producto		FK	INT
	Id_usuario	Identificador del usuario		FK	INT

UP PEDIDO: Tabla que guarda la información antes del cambio. (explicado en triggers)

Tablas	Campos	Detalles del Campo	Pk	Fk	Tipo De Dato
	Id_orden	Identificador de la orden a modificar	PK		INT
<b>Up Pedido</b>	Fecha	Fecha que se solicitó el servicio			DATE
	Nombre	Nombre del usuario que contrato el servicio			VARCHAR(30)
	Tipo	Tipo del servicio contratado			VARCHAR(30)

## LISTADO DE VISTAS:

Nombre de las vistas	Descripción	Objetivo
<b>servicio_contratado_vw</b>	Muestra los servicios que contrato cada usuario.	Facilitar la información al personal que la solicite.
<b>usuarios_dni_vw</b>	Encuentra a los usuarios cuyo DNI sea mayor a 44M.	Encontrar a los usuarios de una determinada edad en adelante.
<b>precios_bajos_vw</b>	Muestra el precio mas bajo de cada servicio.	Facilitar el acceso al precio mas barato de cada servicio.
<b>pedidos_2024_vw</b>	Encuentra y muestra los pedidos realizados en el año 2024.	Facilitar el seguimiento al personal sobre los pedidos realizados este año.
<b>ultimos_pedidos_vw</b>	Encuentra y muestra los últimos 8 pedidos realizados con éxito.	Facilitar el seguimiento de los últimos pedidos realizados.

## LISTADO DE FUNCIONES:

Funciones	Descripción	Objetivo
<b>precio_servicio_venta_final_fn</b>	En esta función se debe de ingresar los valores para obtener el precio de determinado servicio.	Averiguar el precio de venta de un servicio.
<b>calcular_iva_venta_fn</b>	Averigua el IVA de cada venta en diferentes porcentajes de este.	Calcular el IVA que debe tener cada venta.
<b>calcular_total_venta_fn</b>	Realiza los cálculos para encontrar el total de una venta.	Calcular precio final + IVA de un servicio.

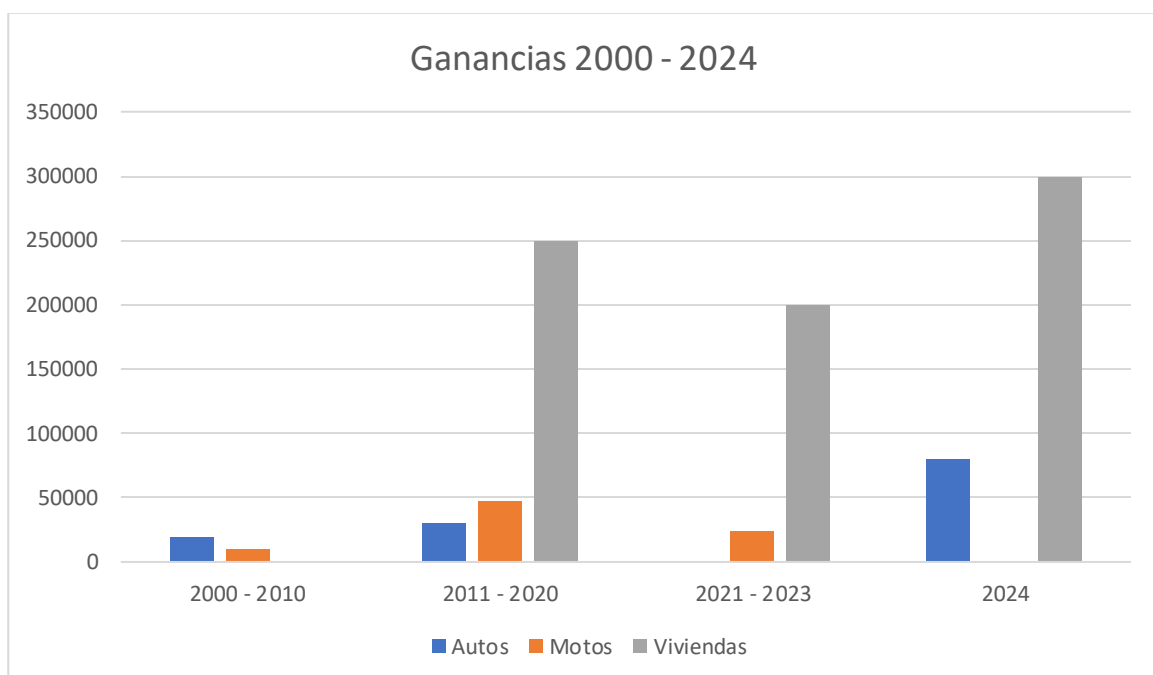
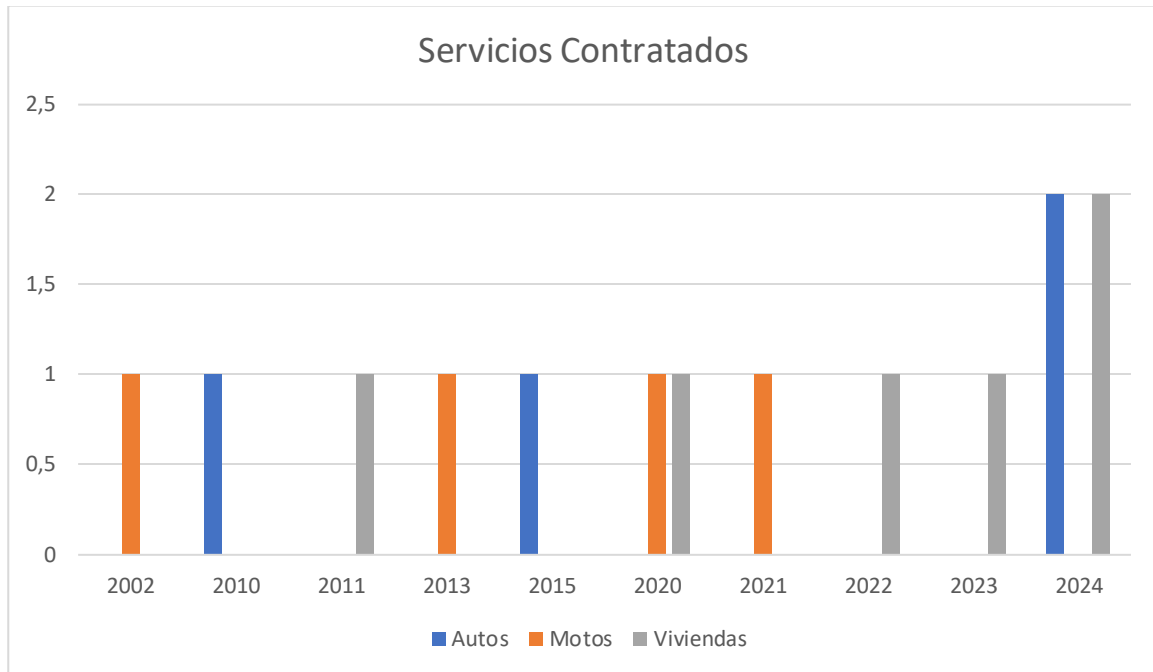
### LISTADO DE STORED PROCEDURES:

Stored Procedures	Descripción	Objetivo
ordenar_tablas_sp	Ordenar las tablas seleccionadas de forma ascendente o descendente.	Facilitar el orden de la información al personal.
insertar_nuevo_usuario_sp	Ingresa usuarios nuevos a la DB.	Agilizar el ingreso de nuevos usuarios
servicios_contratados_sp	Ingresa los nuevos servicios contratados.	Facilitar la demanda de pedidos al personal.

### LISTADO DE TRIGGERS:

Triggers	Descripcion
clientes_nuevos_tr	Este trigger tiene la tarea de que, a cada servicio vendido, almacenar en una nueva tabla a los clientes. Monitorea la tabla "servicios_contratados", luego de que se realice un INSERT y registra todo en la tabla "clientes"
up_pedido_tr	Este trigger tiene que funcionar a medida que los pedidos se modifiquen, ya sea que hubo un error de selección o algo, guarde la información previa a el cambio. Monitorea la tabla "servicios_contratados", luego de que se realice un UPDATE registra todo en la tabla "up_pedido"

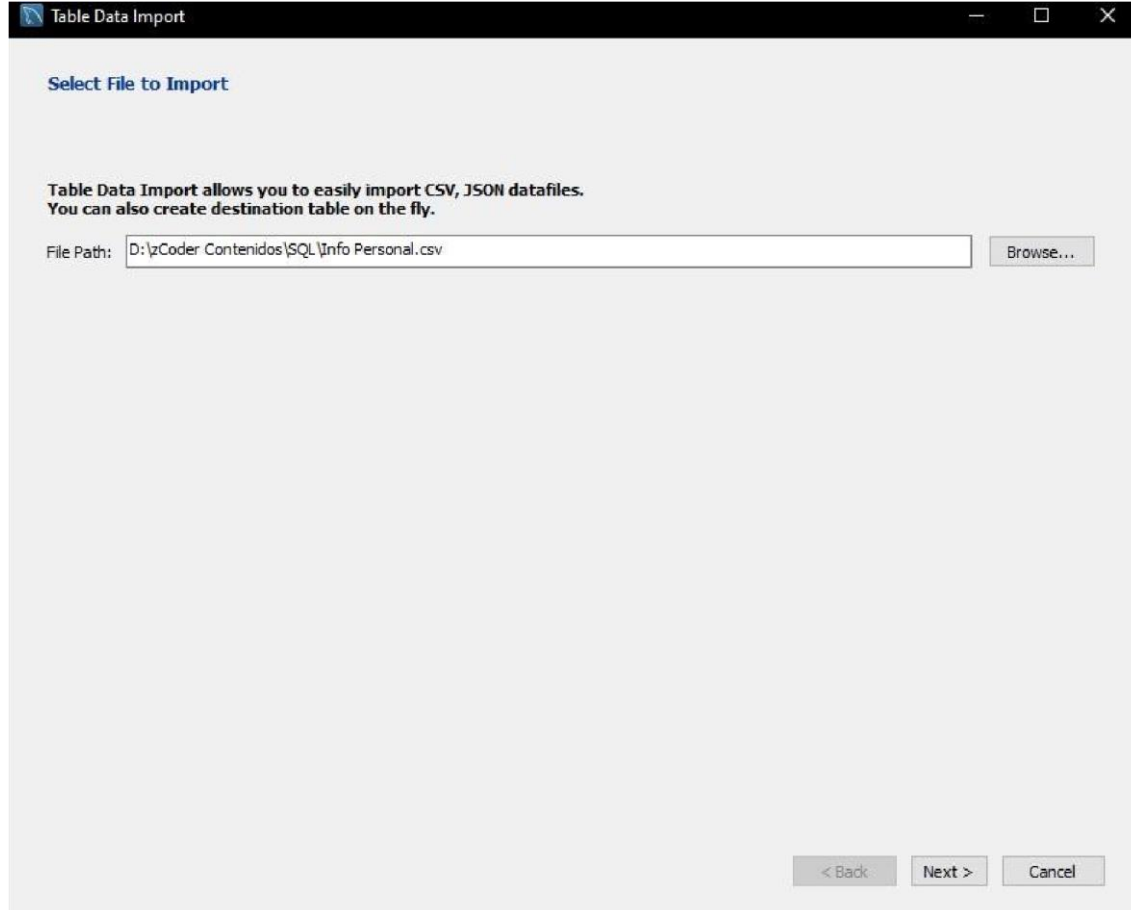
## REPORTES:



## INSERCIÓN DE DATOS EN FORMATO CSV:

Para las tablas usuario, seguros\_viviendas e información\_personal importe los datos de las siguientes tablas.

- Primero cree la tabla:



```

CREATE TABLE IF NOT EXISTS informacion_personal (
  nombre VARCHAR(40) NOT NULL,
  apellido VARCHAR(40) NOT NULL,
  telefono FLOAT(20) NOT NULL,
  email VARCHAR(30) NOT NULL,
  dni NUMERIC(20) NOT NULL,
  id_usuario INT AUTO_INCREMENT,
  PRIMARY KEY (id_usuario, dni),
  FOREIGN KEY (id_usuario) REFERENCES usuario(id_usuario));

```

- Luego inserte los datos yendo al dibujo de la carpeta que dice "Import records from an external file":




- Una vez adentro busque el documento .CSV para poder hacer la inserción:
- Luego seleccione la tabla a la que quería que se importen los datos:



**Table Data Import**

### Configure Import Settings

Detected file format: csv 

Encoding: utf-8

Columns:

<input checked="" type="checkbox"/> Source Column	Dest Column
<input checked="" type="checkbox"/> nombre	<span>nombre</span>
<input checked="" type="checkbox"/> apellido	<span>apellido</span>
<input checked="" type="checkbox"/> telefono	<span>telefono</span>
<input checked="" type="checkbox"/> email	<span>email</span>
<input checked="" type="checkbox"/> dni	<span>dni</span>

nombre	apellido	telefono	email	dni
Dagny	Vawton	943036183	dvawton0@...	850220765
Maure	Guilliatt	36562657	mguilliatt1...	695399943
Maryanna	Neeves	840391457	mneeves2...	635002238
Murdoch	Lexa	62025359	mlexa3@dii...	491248950
Carline	Ricioppo	659265671	cricioppo4...	859807773

< Back   Next >   Cancel

### Select destination table and additional options.

☒ Use existing table: agencia\_aseguradora.informacion\_personal

☐ Create new table: agencia\_asegurador . Info Personal

☐ Truncate table before import

- Puse siguiente y corroboré que estén todos los datos en orden:

- Pulse Next hasta que se importaron los datos:

File D:\zCoder Contenidos\SQL\Info Personal.csv was imported in 0.533 s  
 Table agencia\_aseguradora.informacion\_personal has been used  
 10 records imported

- Finalizado el proceso fui a revisar que este toda la información en la tabla de mi db:

89 ● `SELECT * FROM informacion_personal;`

90

nombre	apellido	telefono	email	dni	id_usuario
Dagny	Vawton	943036000	dvawton0@networksolutions.com	850220765	1
Maure	Guilliatt	36562700	mguilliatt1@tripod.com	695399943	2
Maryanna	Neeves	840391000	mneeves2@discuz.net	635002238	3
Murdoch	Lexa	62025400	mlexa3@diigo.com	491248950	4
Carline	Ricioppo	659266000	cricioppo4@comcast.net	859807773	5
Leticia	Shrubshall	123471000	lshrubshall5@1688.com	98896917	6
Bren	Luchelli	171838000	bluchelli6@opensource.org	357657411	7
Julie	Oxteby	828692000	joxteby7@gizmodo.com	297418034	8
Wilton	Merton	731660000	wmerton8@amazon.de	206114829	9
Fallon	Fairholme	679161000	ffairholme9@addtoany.com	233336126	10
NULL	NULL	NULL	NULL	NULL	NULL

Y listo, este procedimiento lo repetí en las tablas “usuario” y “seguros\_viviendas”.

### HERRAMIENTAS UTILIZADAS:

- MySQL Workbench 8.0 CE
- EXCEL(IMPORTAR DATOS Y TRANSFORMARLOS A .CSV)
- WORD (CREACION DE TABLAS Y REPORTE)
- MOCKAROO(CREACION DE DATOS)
- GITHUBG(CONTROLAR LAS DIFERENTES VERSIONES DEL PROYECTO)

### CONCLUSION:

- Se logro desarrollar de manera eficiente la base de datos para manejar los diferentes servicios de seguros. Utilizando tablas, funciones, vistas, store procedures, triggers.
- Se pudo satisfacer la parte del manejo de los datos de la empresa.
- La base de datos mejora la gestión de la empresa, ya que se automatizan los pedidos de manera que no necesitan un operario haciendo uno por uno.

- Este sistema optimiza las operaciones diarias de la empresa y proporciona una base escalable y la adaptabilidad a las necesidades futuras.