

# BASE DE DATOS I

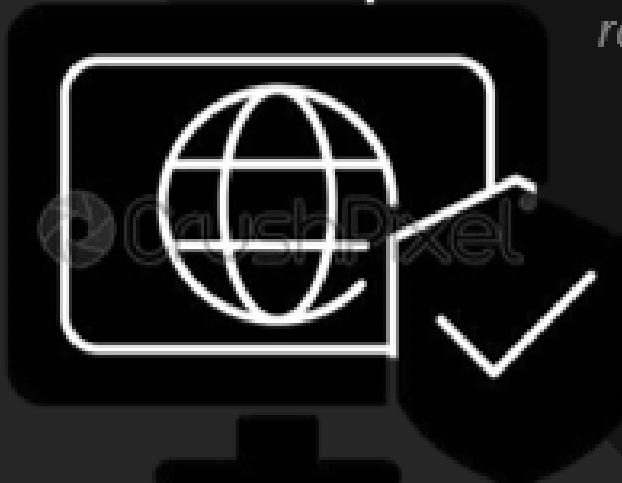


## PROYECTO FINAL

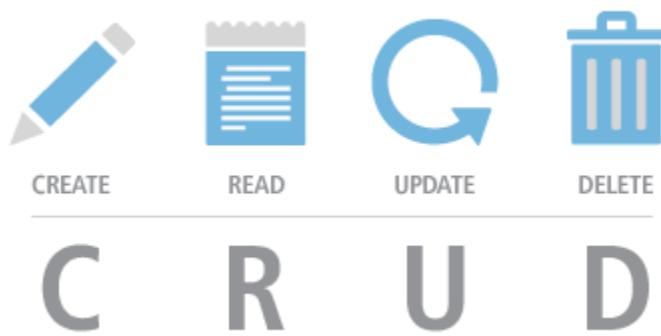
2021



*SQL es un lenguaje de computación para trabajar con conjuntos de datos y las relaciones entre ellos*



# PROYECTO FINAL BASE DE DATOS



## Facultad:

- Ingeniería

## Estudiantes:

- Martin Josue Mamani Pilco
- Mijael Jhonatan Rojas Arias
- Laura Nathalia Quispe Pacari

## Docente:

- Ing. William Roddy Barra Paredes

**Fecha:** 22/Noviembre/2021

**Nombre del equipo:** 7

## 1. Introducción

El objetivo del proyecto es sistematizar los registros de los pacientes de un consultorio dental para agilizar los registros de estos y acceder a ellos con mayor facilidad.

## 2. Diseño de la base de datos

### 2.1. Contexto de la Base de Datos

Dada la situación de registrar los datos de los pacientes, el nombre más adecuado para la base de datos es Reg\_pacientes.

### 2.2. Análisis y definición de las tablas

Descripción de las tablas

Nombre de las tablas	Descripción
usuario	Almacena los usuarios logueados al software
datos_paciente	Almacena los datos de cada paciente registrado por el Doctor al software .
antecedentes_patologicos	Esta tabla relaciona con la tabla datos_paciente. Almacena los antecedentes patológicos (Enfermedades que puede haber padecido el paciente) del paciente registrado.

<b>Observaciones_patologicas</b>	Esta tabla se relaciona con la tabla antecedentes_patologicos. Son las observaciones de los antecedentes patológicos del paciente registrado.
<b>Antecedentes_no_patologicos</b>	Esta tabla relaciona con la tabla datos_paciente. Almacena los antecedentes no patológicos (Datos relacionados con su medio) del paciente registrado.
<b>Observaciones</b>	Esta tabla relaciona con la tabla datos_paciente. Almacena las observaciones del paciente registrado por el Doctor.
<b>Datos_paciente-doctor</b>	Esta tabla relaciona con la tabla datos_paciente. Almacena todos los datos personales entre el paciente y el Doctor

## 2.3 Diseño de la base de datos

### 2.3.1. Código SQL de las tablas

Usuario	<pre> create table usuario (     id_usuario int primary key identity,     usuario varchar (50),     contra varchar (50),     telefono int,     correo varchar (50),     nombres varchar(50),     apellidos varchar (50) );  insert into usuario(usuario, contra, telefono, correo, nombres, apellidos) values ('MijaelR', 'mijaelr2003', 60121247, 'mijaelrojas67@gmail.com', 'Mijael Jhonatan', 'Rojas Arias'),        ('GroberR', 'grober2000', 70560909, 'groberrojaschoque@gmail.com', 'Grober', 'Rojas Choque'),        ('MassielR', 'massielr2009', 63114605, 'massielrojas2009@gmail.com', 'Massiel Yelena', 'Rojas Arias'); </pre>
Datos_paciente	<pre> create table datos_paciente(     ci_pacientes varchar(20) primary key ,     nombres varchar(50) ,     apellidos varchar(50) ,     edad int ,     genero varchar (1),     ocupacion varchar (50),     lugar_nacimiento varchar(70),     domicilio varchar(50),     motivo_consulta varchar (200),     telefono int,     fecha_nacimiento varchar (20),     nombre_tutor varchar(50), );  insert into datos_paciente (ci_pacientes, nombres, apellidos, edad, genero, ocupacion, lugar_nacimiento, domicilio, motivo_consulta, telefono, fecha_nacimiento, nombre_tutor) values ('1324657 LP', 'Jhamil Max', 'Rojas Rojas', 20, 'M', 'Programador', 'Nuestra Señora de La Paz', '2 de Abril', 'Consulta', 1324678, '12/02/2000', 'Luis Fernando Rojas') </pre>

Antecedentes_patologicos	<pre> create table antecedentes_patologico (     id_antecedente_patologico int primary key identity ,     alergias varchar (2),     hemorragias varchar(2),     alteracion_en_la_cicatrizacion varchar(2),     traumatismos_craneo_faciales varchar(2),     experiencias_previas_de_exodoncia varchar (2),     paciente_aparentemente_sano varchar(2),     ci_pacientes varchar(20) ,     foreign key (ci_pacientes) references datos_paciente(ci_pacientes), );  insert into antecedentes_patologico (alergias, hemorragias, alteracion_en_la_cicatrizacion, traumatismos_craneo_faciales, experiencias_previas_de_exodoncia, paciente_aparentemente_sano, ci_pacientes) values ('No','No', 'No', 'No', 'No', 'Si', '1324657 LP') </pre>
Observaciones_patologicos	<pre> create table observaciones_patologicas (     id_observaciones_patologicas int primary key identity ,     traumatismos_craneo_faciales_obs varchar(50),     experiencias_previas_de_exodoncia_obs varchar (50),     paciente_aparentemente_sano_obs varchar (50),     id_antecedentes_patologicos int,     foreign key (id_antecedentes_patologicos) references antecedentes_patologico (id_antecedente_patologico) );  insert into observaciones_patologicas (traumatismos_craneo_faciales_obs, experiencias_previas_de_exodoncia_obs, paciente_aparentemente_sano_obs, id_antecedentes_patologicos) values ('No padece de nada', 'No padece de nada', 'No padece de nada', 1) </pre>

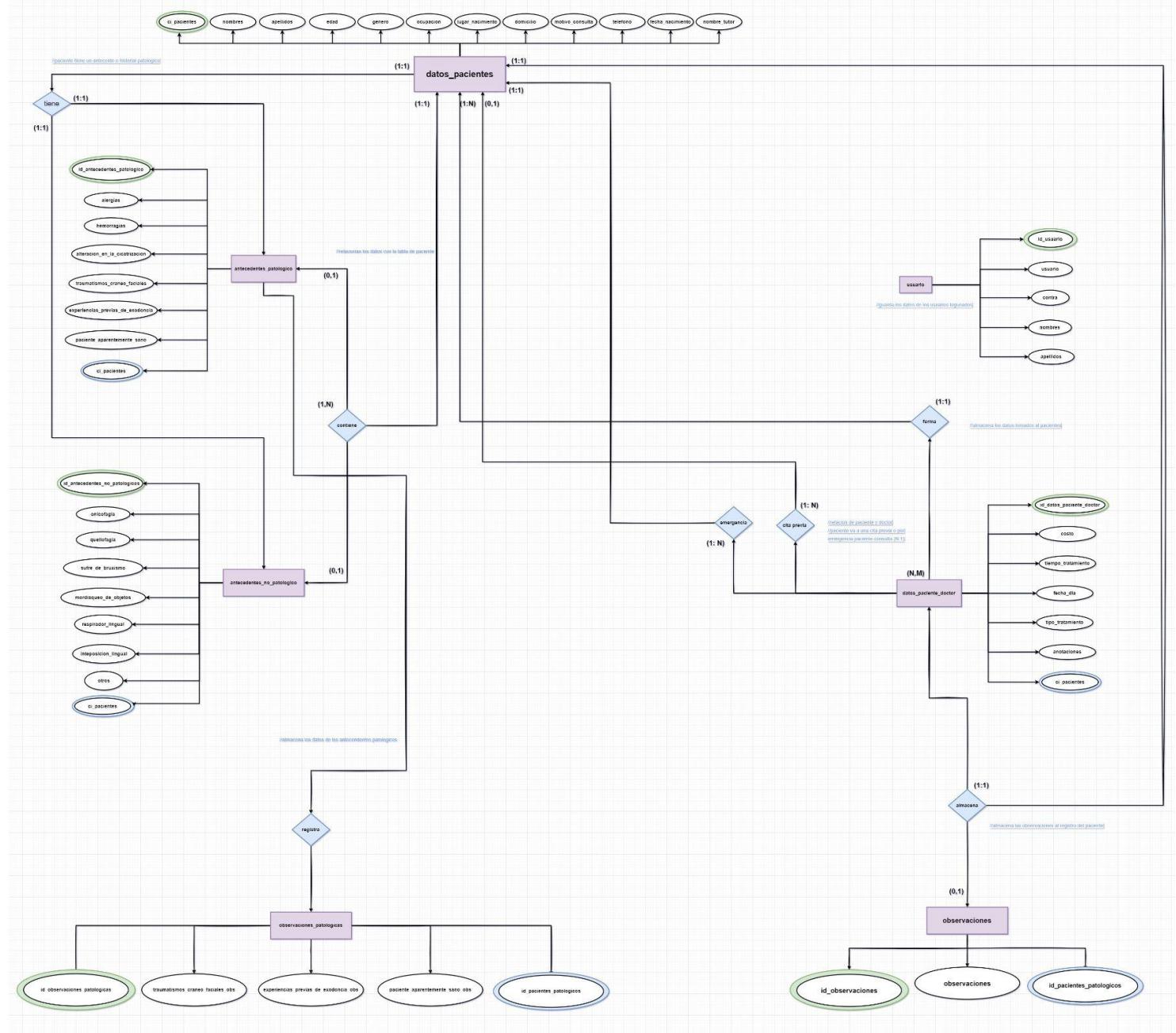
Antecedentes_no_patologicas	<pre> create table antecedentes_no_patologicos (     id_antecedentes_nopatologicos int primary key identity,     onicofagia varchar (2),     queilofagia varchar (2),     sufre_de_bruxismo varchar (2),     mordisqueo_de_objetos varchar (2),     respirador_bucal varchar (2),     inteposicion_lingual varchar (2),     otros varchar (50),     ci_pacientes varchar(20) ,     foreign key (ci_pacientes) references datos_paciente(ci_pacientes), );  insert into antecedentes_no_patologicos (onicofagia, queilofagia, sufre_de_bruxismo, mordisqueo_de_objetos, respirador_bucal, inteposicion_lingual, otros, ci_pacientes) values ('No', 'No', 'No', 'No', 'No', 'No', 'NO', '1324657 LP') </pre>
Observaciones	<pre> create table observaciones (     id_observaciones int primary key identity,     observaciones varchar (500),     ci_pacientes varchar(20) ,     foreign key (ci_pacientes) references datos_paciente(ci_pacientes), );  insert into observaciones (observaciones, ci_pacientes) values ('No hay obsaervaciones', '1324657 LP') </pre>

Datos\_paciente\_doctor

```
create table datos_paciente_doctor (
    id_datos_paciente_doctor int primary key identity ,
    costo varchar (50) ,
    tiempo_tratamiento varchar (50),
    fecha_dia varchar (20) ,
    tipo_tratamiento varchar (200),
    anotaciones varchar (200),
    ci_pacientes varchar (20) ,
    foreign key (ci_pacientes) references datos_paciente (ci_pacientes)
);

insert into datos_paciente_doctor (costo, tiempo_tratamiento, fecha_dia,
tipo_tratamiento, anotaciones, ci_pacientes)
values ('100 bs', '1 semana', '06/11/2021', 'Consulta', 'Varias cosas que
hacer', '1324657 LP')
```

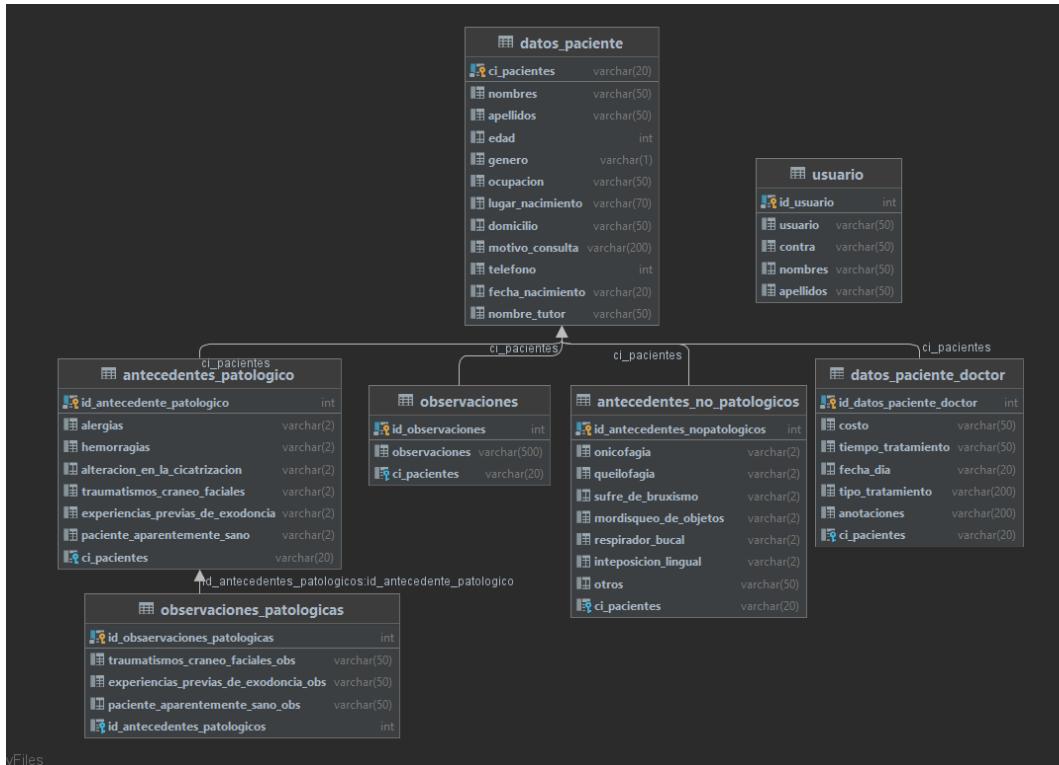
### 2.3.2. Modelo entidad-relación de la Base de Datos



Url para mejor visualización:

[https://drive.google.com/file/d/1wRlOQ4ZgMQ4x4KrRyz10i1Gug6mup\\_uC/view?usp=sharing](https://drive.google.com/file/d/1wRlOQ4ZgMQ4x4KrRyz10i1Gug6mup_uC/view?usp=sharing)

### 2.3.3. Modelo lógico de la Base de Datos



### 2.4. Búsquedas, funciones, vistas, etc.

#### 3. Usabilidad

##### 3.1. Imágenes acerca del uso del sistema

-Este es el login en el cual el usuario ingresa con su nombre de usuario y contraseña



-Así es como se ve el programa una vez ingresamos. Esta pantalla nos permite ver consultas, hacer consultas específicas, agregar registros, borrar registros y modificar los registros ya existentes



-Aquí podemos observar la consulta general de los registros de los pacientes que hay

	ci_pacientes	nombres	apellidos	edad	genero	ocupacion
>	1324657 LP	Jhamil Max	Rojas Rojas	17	M	Programador
	1324658 LP	Alex Joel	Callisaya Tomez	18	M	Estudiante
*						

-Aquí vemos las consultas específicas según las coloquemos

Form2

### Registro de datos de pacientes

ci_pacientes	nombres	apellidos	edad	genero	ocupacion
1324658 LP	Alex Joel	Callisaya Torrez	18	M	Estudiante
*					

**Consulta**

Alex Joel

**Borrar**

**CI**

**Agregar**

**CI**  **Nombres**  **Apellidos**  **Edad**  **Genero**

**Ocupacion**  **Lu.Nacimiento**  **Domicilio**  **Telefono**

**Motivo Consulta**  **Fe.Nacimiento**

**Nombre\_tutor**

**Actualizar**

**Cambiar el/a**  a

**si su:**  es  **Salir**

-Aquí agregamos un registro completo de un paciente

Form2

### Registro de datos de pacientes

ci_pacientes	nombres	apellidos	edad	genero	ocupacion
1324657 LP	Jhamil Max	Rojas Rojas	17	M	Programador
1324658 LP	Alex Joel	Callisaya Torrez	18	M	Estudiante
*					

**Consulta**

**Borrar**

**CI**

**Agregar**

**CI**  1234789 LP **Nombres** Martin Mamani **Apellidos** Mamani Pilco **Edad** 18 **Genero** M

**Ocupacion** Estudiante **Lu.Nacimiento** Señora de La Paz **Domicilio** 16 de julio **Telefono** 12345678

**Motivo Consulta** Limpieza **Fe.Nacimiento** 10/01/2000

**Nombre\_tutor** -

**Actualizar**

**Cambiar el/a**  a

**si su:**  es  **Salir**

-Aquí se ve el mensaje de que se añadió un nuevo registro



-Aquí ya podemos ver el registro añadido en nuestra base de datos

The screenshot shows a Windows application window titled "Form2". The main title of the form is "Registro de datos de pacientes". On the left side, there are several buttons: "Consulta", "Borrar", "CI", and "Agregar" (which is highlighted with a blue border). Below these are input fields for "CI", "Nombres", "Apellidos", "Edad", "Genero", "Ocupacion", "Lu.Nacimiento", "Domicilio", "Telefono", "Motivo Consulta", "Fe.Nacimiento", "Nombre\_tutor", and "Actualizar". At the bottom left is a "Cambiar el/la" field followed by "a" and another field. At the bottom right is a "Salir" button. A table is displayed in the center, showing patient data with columns: ci\_pacientes, nombres, apellidos, edad, genero, and ocupacion. The first row of the table is highlighted in blue.

ci_pacientes	nombres	apellidos	edad	genero	ocupacion
1234789 LP	Martin Mamani	Mamani Pilco	18	M	Estudiante
1324657 LP	Jhamil Max	Rojas Rojas	17	M	Programador
1324658 LP	Alex Joel	Callisaya Torrez	18	M	Estudiante
*					

-Aquí podemos observar cómo se modifica un registro

Form2

### Registro de datos de pacientes

	ci_pacientes	nombres	apellidos	edad	genero	ocupacion
>	1234789 LP	Martin Mamani	Mamani Pilco	18	M	Estudiante
	1324657 LP	Jhamil Max	Rojas Rojas	17	M	Programador
*	1324658 LP	Alex Joel	Callisaya Torrez	18	M	Estudiante

CI   
Agregar  Consulta  Borrar

Ocupacion  Lu.Nacimiento   
Motivo Consulta   
Nombre\_tutor   
Actualizar   
Cambiar el/la  a   
si su:  es  Genero   
Telefono   
Salir

Se actualizo el registro correctamente  Aceptar

-Y ya se modificó el registro corrigiendo el error

Form2

### Registro de datos de pacientes

	ci_pacientes	nombres	apellidos	edad	genero	ocupacion
>	1234789 LP	Martin Josue	Mamani Pilco	18	M	Estudiante
	1324657 LP	Jhamil Max	Rojas Rojas	17	M	Programador
*	1324658 LP	Alex Joel	Callisaya Torrez	18	M	Estudiante

CI   
Agregar  Consulta  Borrar

Ocupacion  Lu.Nacimiento  Domicilio  Telefono   
Motivo Consulta  Fe.Nacimiento   
Nombre\_tutor   
Actualizar   
Cambiar el/la  a   
si su:  es  Genero   
Telefono   
Salir

-Y por último ahora vemos como se borra un registro existente

Form2

### Registro de datos de pacientes

ci_pacientes	nombres	apellidos	edad	genero	ocupacion
1234789 LP	Martin Josue	Mamani Pilco	18	M	Estudiante
1324657 LP	Jhamil Max	Rojas Rojas	17	M	Programador
1324658 LP	Alex Joel	Callisaya Tomez	18	M	Estudiante
*					

CI  
1234789 LP

Borrar

CI Nombres Apellidos  
Ocupacion Lu.Nacimiento Domicilio Telefono  
Motivo Consulta  
Nombre\_tutor  
Actualizar  
Cambiar el/la a  
si su: es

Genero  
Acepta

Se elimino el registro correctamente

Salir

-Finalmente vemos como se eliminó el registro

Form2

### Registro de datos de pacientes

ci_pacientes	nombres	apellidos	edad	genero	ocupacion
1324657 LP	Jhamil Max	Rojas Rojas	17	M	Programador
1324658 LP	Alex Joel	Callisaya Tomez	18	M	Estudiante
*					

CI  
Agregar  
CI Nombres Apellidos Edad Genero  
Ocupacion Lu.Nacimiento Domicilio Telefono  
Motivo Consulta Fe.Nacimiento  
Nombre\_tutor  
Actualizar  
Cambiar el/la a  
si su: es

Salir

### 3.2. Video acerca del uso del sistema

URL del video:

[https://drive.google.com/drive/folders/1HpG52hG\\_TX7cICTwjdqT1V-ElOcOwosp?usp=sharing](https://drive.google.com/drive/folders/1HpG52hG_TX7cICTwjdqT1V-ElOcOwosp?usp=sharing)

### 4. Conclusión

Este proyecto resultó ser muy educativo, además de que pudimos ver lo importante en la toma de registros pertinentes para administrar una base de datos pequeña. Esta experiencia nos ayudó a comprender mejor el manejo de una base de datos así como comprender el manejo en conjunto de una base de datos con Windows forms de Visual Studio.