



UNIVERSIDAD DEL ISTMO

FACULTAD DE INGENIERÍA

BASES DE DATOS

DOCUMENTACION TECNICA

JAVIER ALEJANDRO LÓPEZ GUZMAN

Guatemala, 29 de octubre de 2021

# Contenido

Contenido.....	2
Tablas.....	3
Funciones .....	13
Procedimientos almacenados .....	14
Vistas.....	19
Triggers .....	21
Diagrama de arquitectura.....	23
Vista, modelo, controlador.....	24

# Tablas

Nombre: **centrosdevacunacion**

Función: Almacenar información de los centros de vacunación disponible

Contenido:

Campos	Tipo de Dato	Descripción	Llaves
Numero	Int(11)	Identificador del centro de vacunación	Primaria
Nombre	Varchar(255)	Nombre del centro de vacunación	
Departamento	Varchar(255)	Departamento donde se encuentra el centro de vacunación	
Dirección	Varchar(255)	Dirección donde se encuentra el centro de vacunación	

Nombre: **centrovacunacione**

Función: Almacenar la vacuna escogida por el usuario.

Contenido:

Campos	Tipo de Dato	Descripción	Llaves
<b>dpi</b>	bigInt()	Identificador del usuario	Primaria Llave foranea(usuarios/dpi)
<b>centroVacunacionE</b>	Varchar(255)	Centro de vacunación elegido por el usuario	

Nombre: **fechaVacunacion**

Función: Almacena las fechas programadas para la vacunación del usuario.

Contenido:

Campos	Tipo de Dato	Descripción	Llaves
<b>dpi</b>	bigInt()	Identificador del usuario	Primaria Llave foranea(usuarios/dpi)
<b>primerDosis</b>	date	Fecha de vacunación al usuario primer dosis	
<b>primerDosis</b>	date	Fecha de vacunación al	

		usuario segunda dosis	
--	--	--------------------------	--

Nombre: **noticias**

Función: Almacena noticias que se presentaran en el portal para todos los usuarios.

Contenido:

Campos	Tipo de Dato	Descripción	Llaves
numero	Int(11)	Identificador de la noticia	Primaria
titulo	Varchar(255)	Titulo de la noticia a presentar	
descripcion	text	Contenido de toda la noticia	
imagen	Varchar(90)	Path donde se guarda la imagen a mostrar	

Nombre: **token**

Función: Almacena tokens únicos de los usuarios para poder descargar un PDF con su proceso de vacunación

Contenido:

Campos	Tipo de Dato	Descripción	Llaves
<b>dpi</b>	bigInt()	Identificador del usuario	Primaria Llave foranea(usuarios/dpi)
<b>token</b>	bigInt()	Token único por usuario para visualizar su proceso de vacunación	

Nombre: **usuarios**

Función: Almacena toda la información importante y necesaria para el portal y el proceso de vacunación de cada individuo que se registra.

Contenido:

Campos	Tipo de Dato	Descripción	Llaves
dpi	bigInt()	Identificador del usuario	Primaria Llave
Nombre	Varchar(255)	Nombre del usuario	
Apellido	Varchar(255)	Apellido del usuario	
Correo	Varchar(255)	Correo donde se enviarán recordatorios al usuario y un correo de registro	
Clave	Varchar(255)	Llave con la que el usuario puede ingresar a su cuenta	
Fecha_de_nacimiento	Date	Fecha de nacimiento para poder saber si el usuario tiene permitido vacunarse	
Enfermedad	Varchar(150)	Campo para saber si el usuario padece de una enfermedad crónica para poder saber si el usuario tiene permitido vacunarse	
Celular	Int(8)	Celular del usuario	
tipoUsuario	Varchar(255)	Tipo de usuario para poder brindar a la persona que esta usando el portal ciertos permisos para efectuar	



		ciertas acciones dentro del portal	
grupoPrioritario	Varchar(5)	Campo para saber si el usuario pertenece a un grupo prioritario para poder saber si el usuario tiene permitido vacunarse	

Nombre: **usuariosvalidadosvacuna**

Función: Es la tabla encargada de verificar si el usuario cumple con los requisitos para ser vacunado, esto lo hace cambiando el dato de un campo entre “no” y “si”, este dato cambia al momento de llamar una función almacenada que se encarga de recibir los parámetros para saber si el usuario es apto o no de vacunarse.

Contenido:

Campos	Tipo de Dato	Descripción	Llaves
<b>dpi</b>	bigInt()	Identificador del usuario	Primaria Llave foranea(usuarios/dpi)
<b>Enfermedad</b>	Varchar(150)	Campo para saber si el usuario padece de una enfermedad crónica para poder saber si el usuario tiene permitido vacunarse	
<b>grupoPrioritario</b>	Varchar(5)	Campo para saber si el usuario pertenece a un grupo prioritario para poder saber si el usuario tiene permitido vacunarse	
<b>Fecha_de_nacimiento</b>	Date	Fecha de nacimiento para poder saber si el usuario tiene permitido vacunarse	
<b>validado</b>	Varchar(5)	Campo que muestra si el usuario tiene permitido o no vacunarse	

Nombre:**usuariovacunado**

Función: Tabla encargada de almacenar el dato si el usuario ya se puso una de las dosis o las dos dosis.

Contenido:

Campos	Tipo de Dato	Descripción	Llaves
dpi	bigInt()	Identificador del usuario	Primaria Llave foranea(usuarios/dpi)
primerDosisPuesta	Varchar(10)	Muestra si el usuario ya se colocó la primera dosis.	
segundaDosisPuesta	Varchar(10)	Muestra si el usuario ya se colocó la segunda dosis.	

Nombre:**usuariovacunado**

Función: Tabla encargada de almacenar la vacuna que fue administrada al usuario.

Contenido:

Campos	Tipo de Dato	Descripción	Llaves
dpi	bigInt()	Identificador del usuario	Primaria Llave foranea(usuarios/dpi)
Vacuna	Varchar(255)	Nombre de la vacuna colocada	

Nombre: **vacunas**

Función: Tabla encargada de almacenar todas las vacunas disponibles.

Contenido:

Campos	Tipo de Dato	Descripción	Llaves
numero	Int(11)	Identificador de la vacuna	Primaria
nombre	Varchar(35)	Nombre de la vacuna	
numeroDosis	Int(11)	Cuántas dosis se deben aplicar por persona de la vacuna ingresada	
tiempoEntreDosis	Int(11)	El tiempo entre cada dosis	

# Funciones

Nombre: saludoAlUsuario

Función: Concatenar el nombre y apellido del usuario en una oración de saludo para poder mostrar que tipos de permiso tiene el usuario.

Parámetros: nombre(varchar(250)), apellido(varchar(250)), tipoUsuario(varchar(250))

Instrucciones:

```
1 BEGIN
2     RETURN CONCAT("Hola ", nombre, " ", apellido, " sus
    permisos de usuario son de: ", tipoUsuario);
3 END
```

## Procedimientos almacenados

Nombre: **borrarUsuario**

Parámetros: dpiP(bigInt || IN)

Función: Eliminar toda la información de un usuario en todas las tablas que estén vinculadas al dpi del usuario (crear un usuario).

Tablas afectadas:

- usuarios
- fechavacunacion
- usuariovacunado
- centroVacunacionE
- usuariovalidadovacuna
- vacunacolocada
- token

Instrucciones:

```
1 BEGIN
2     delete from fechavacunacion where dpi = dpiP;
3     delete from usuariovacunado where dpi = dpiP;
4     delete from centrovacunacione where dpi = dpiP;
5     delete from usuariosvalidadosvacuna WHERE dpi = dpiP;
6     delete from vacunacolocada WHERE dpi = dpiP;
7     delete from usuarios WHERE dpi = dpiP;
8     delete from token where dpi = dpiP;
9     COMMIT;
10 END
```

Nombre: **fechaUsuario**

Parámetros: dpiP(bigInt || IN), nombre(varchar(255) || IN), apellido(varchar(255) || IN), correo(varchar(255) || IN), clave(varchar(255) || IN), fecha\_de\_nacimiento(date || IN), enfermedad(varchar(255) || IN), celular(int(255) || IN), tipoUsuario(varchar(255) || IN), uPrioritario(varchar(5) || IN), centroVacunacionEs(varchar(255) || IN)

Función: Insertar en todas las tablas relacionadas con los usuarios información nueva para registrar la información del usuario nuevo.

Tablas afectadas:

- usuarios
- fechavacunacion
- usuariovacunado
- centroVacunacionE
- usuariovalidadovacuna
- vacunacolocada
- token

Instrucciones:

```
2      INSERT INTO usuarios (dpi, nombre, apellido, correo,
3      clave, fecha_de_nacimiento, enfermedad, celular,
4      tipoUsuario, grupoPrioritario)
5      VALUES (dpi,nombre,apellido,correo,
6      clave,fecha_de_nacimiento,enfermedad, celular, tipoUsuario,
7      uPrioritario);
8      insert into fechavacunacion (dpi)
9      VALUES (dpi);
10     INSERT INTO usuariovacunado(dpi, primerDosisPuesta,
11     segundaDosisPuesta) VALUES(dpi, 'no', 'no');
12     INSERT INTO centroVacunacionE(dpi, centroVacunacionE)
13     VALUES(dpi, centroVacunacionEs);
14     INSERT INTO usuariosvalidadosvacuna(dpi, enfermedad,
15     grupoPrioritario, fecha_de_nacimiento, validado)
16     VALUES(dpi, enfermedad, uPrioritario, fecha_de_nacimiento,
17     "no");
18     INSERT INTO vacunacolocada(dpi,vacuna)VALUES(dpi, "null");
```

Nombre: **registroVacuna**

Parámetros: dpiP(bigInt || IN), segundaDosis (date || IN), vacunaP(varchar(20) || IN), primerDosisPuesta(varchar(10) || IN), segundaDosisPuesta(varchar(10) || IN)

Función: Actualizar el estado de vacunación de los usuarios al presentarse al centro de vacunación escogido, actualizando la marca de la vacuna que se colocó, cambiando el estado su vacunación de “no” a “si”

Tablas afectadas:

- fechavacunacion
- usuariovacunado
- vacunacolocada

Instrucciones:

```
1 BEGIN
2     START TRANSACTION;
3     UPDATE fechavacunacion SET segundaDosis =segundaDosis
WHERE dpi=dpiP;
4     UPDATE usuariovacunado SET primerDosisPuesta
=primerDosisPuesta, segundaDosisPuesta = segundaDosisPuesta
WHERE dpi =dpiP;
5     UPDATE vacunacolocada SET vacuna = vacunaP WHERE dpi = dpiP;
6     COMMIT;
7 END
```



Nombre: **updateUsuario**

Parámetros: Parámetros: dpiP(bigInt || IN), nombre(varchar(255) || IN), apellidoP(varchar(255) || IN), correo(varchar(255) || IN), claveP(varchar(255) || IN), fecha\_de\_nacimientoP(date || IN), enfermedadP(varchar(255) || IN), celularP(int(255) || IN), tipoUsuarioP(varchar(255) || IN), grupouPrioritarioP(varchar(5) || IN)

Función: Permite que el administrador actualice o cambie datos del usuario desde el CRUD de la aplicación.

Tablas afectadas:

- usuarios
- usuariosvalidadosvacuna

Instrucciones:

```
1 BEGIN
2 UPDATE usuarios SET nombre=nombreP, apellido=apellidoP,
  correo= correoP, clave=claveP, fecha_de_nacimiento=
  fecha_de_nacimientoP, enfermedad= enfermedadP,
  celular=celularP, tipoUsuario = tipoUsuarioP, grupoPrioritario
  = grupoPrioritarioP WHERE dpi=dpiP;
3 COMMIT;
4
5 UPDATE usuariosvalidadosvacuna SET enfermedad = enfermedadP,
  grupoPrioritario = grupoPrioritarioP, fecha_de_nacimiento =
  fecha_de_nacimientoP WHERE dpi = dpiP;
6 END
```

Nombre: **updateVacunaValidacion**

Parámetros: enfermedadP(enfermedadP || IN), uPrioritario (date || IN), fecha\_de\_nacimientoP (varchar(20) || IN)

Función: actualiza las tablas que cumplan con la condición indicada desde el sitio web, este procedimiento ayuda a validar a las personas que pueden vacunarse, solo si estas cumplen los requisitos dados por el administrador.

Tablas afectadas:

- usuariosvalidadosvacuna

Instrucciones:

```
1 BEGIN
2 UPDATE usuariosvalidadosvacuna SET validado = enfermedadP
3 where enfermedad = 'si';
4 UPDATE usuariosvalidadosvacuna SET validado = uPrioritario
5 where grupoPrioritario = 'si';
6 UPDATE usuariosvalidadosvacuna SET validado = 'si'
7 where fecha_de_nacimiento < fecha_de_nacimientoP;
8 END
```

## Vistas

Nombre: correo

Función: obtener el correo, la fecha de la primer y segunda dosis de vacunación del usuario y el centro de vacuna que escogió para poder enviarle un correo de recordatorio al usuario una semana antes.

Campos:

Campo	Tipo de dato	Tabla
Correo	Varchar(255)	Usuarios
primerDosis	Date	fechaVacunacion
segundaDosis	Date	fechaVacunacion
centroVacunacionE	Varchar(255)	centrovacunacione

Nombre: datosusuario

Función: obtener todos los datos del usuario vinculados a su dpi para poder insertar estos en el local storage de la página y así poder controlar los datos del usuario desde react.

Campos:

Campo	Tipo de dato	Tabla
dpi	bigint(20)	Usuarios
nombre	varchar(255)	Usuarios
apellido	varchar(255)	Usuarios
Correo	varchar(255)	Usuarios
Clave	varchar(50)	Usuarios
Fecha_de_nacimiento	date	Usuarios
Enfermedad	varchar(255)	Usuarios
Celular	Int(8)	Usuarios
tipoUsuario	varchar(255)	Usuarios

<b>grupoPrioritario</b>	varchar(5)	Usuarios
<b>centroVacunacionE</b>	varchar(255)	Centrovacunacione
<b>primerDosis</b>	date	fechavacunacion
<b>segundaDosis</b>	date	fechavacunacion
<b>Validado</b>	varchar(5)	Usuariovalidadovacuna
<b>primerDosisPuesta</b>	varchar(10)	usuariovacunado
<b>segundaDosisPuesta</b>	varchar(10)	usuariovacunado
<b>Vacuna</b>	varchar(255)	Vacunacolocada
<b>Saludo</b>	varchar(255)	
<b>token</b>	Bigint(20)	Token

Nombre: entre\_fechas

Función: muestra el dpi, el nombre, apellido, fecha de la primer y segunda dosis, el centro de vacunación escogido, y si la primer dosis o segunda dosis ya fueron puesta.

Campos:

<b>Campo</b>	<b>Tipo de dato</b>	<b>Tabla</b>
<b>dpi</b>	bigint(20)	Usuarios
<b>nombre</b>	varchar(255)	Usuarios
<b>apellido</b>	varchar(255)	Usuarios
<b>centroVacunacionE</b>	varchar(255)	Centrovacunacione
<b>primerDosis</b>	date	fechavacunacion
<b>segundaDosis</b>	date	fechavacunacion
<b>primerDosisPuesta</b>	varchar(10)	usuariovacunado
<b>segundaDosisPuesta</b>	varchar(10)	usuariovacunado

# Triggers

Nombre: colocarPrimerDosisAfter

Tabla: usuariosvalidadosvacuna

Tiempo: AFTER

Evento: UPDATE

Funcion: Actualizar la tabla fechavacunacion al momento de actualizar un dato en usuariosvalidadosvacuna(cuando se valida un usuario), colocando una fecha 17 dias después del día de hoy para así colocar la primer dosis.

Definición:

```
1 BEGIN
2 DECLARE dpiOtraTabla bigint;
3
4 Select dpi into dpiOtraTabla from
  usuariosvalidadosvacuna where dpi = New.dpi;
5 UPDATE fechavacunacion SET primerDosis =
  date_add(now(), interval 17 day)
6 where dpi = dpiOtraTabla;
7
8 END
```

Nombre: colocarSegundaFecha

Tabla: vacunacolocada

Tiempo: AFTER

Evento: UPDATE

Funcion: Actualizar el campo segundaDosis que hace referente a la segunda fecha de vacunación del usuario, este campo se actualiza al momento de ingresar el tipo de vacuna que el usuario se colocó.

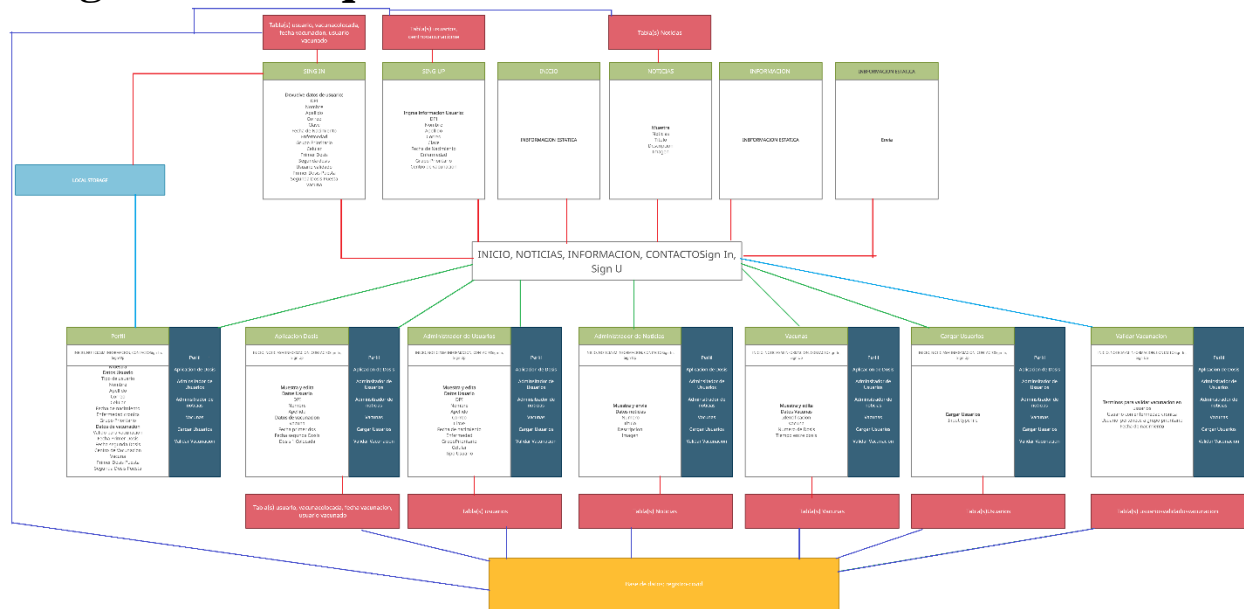
También actualiza el campo primerDosisPuesta de la tabla usuariovacunado para poner que el usuario si se vacuno.

Definición:

```
BEGIN
    UPDATE fechavacunacion SET
segundaDosis = date_add(now(), interval
365 day) WHERE dpi = new.dpi;

    UPDATE usuariovacunado SET
primerDosisPuesta = "si" where dpi =
new.dpi;
END
```

# Diagrama de arquitectura



## Vista, modelo, controlador

