



Documentación Técnica

Base de Datos registro_covid

ADALI GARRAN JIMENEZ

TABLA DE CONTENIDOS:

INTRODUCCIÓN.....	4
SIMBOLOGÍA.....	4
TABLAS.....	5
• poblacion:	5
• correos:	6
• telefonos:	6
• vacunas:	7
• puesto_vacunacion:	8
• enfermedad_cronica:	9
• profesiones:.....	10
• poblacion_enferma:	10
• persona_profesion:	11
• tipo_usuario:	12
• usuarios:.....	13
• datos_registro:	14
• pdf_proceso:	15
• solicitudes_empleados:.....	17
• imagen_descripcion:.....	17
• imagen_texto_lado:	18
• titulo_texto:.....	18
VISTAS:	19
• cita_perdida:	19
• info_usuarios:	20

• reportes_centro:	21
FUNCIONES	22
• correo1_persona:	22
• correo2_persona:	22
• Count_usuario:	22
• dias_vacuna:	23
• dosis_vacuna:	23
• enfermedad:	23
• enfermedad_id:	23
• es_usuario:	24
• genero_persona:	24
• nombre_centro:	24
• nombre_persona:	25
• nombre_vacuna:	25
• profesion_persona:	25
• profesion_persona_id:	25
• tel1_persona:	25
• tel2_persona:	26
• tipo_usuario:	26
PROCEDIMIENTOS ALMACENADOS	26
• actualizar_registro:	26
• add_persona:	27
• aprobar_solicitud:	27
• delete_usuario:	27

• generar_pdf:	28
• generar_pdf:	28
• habilitar_edad:	28
• habilitar_enfermedad:	29
• habilitar_trabajo:	29
• insertVacunas:	29
• nuevo_registro:	30
• nuevo_usuario:	30
• solicitud_empleado:	30
• update_poblacion:	31
TRIGGERS	31
• aplica_v_enf:	31
• aplica_v_prof:	31
• asignar_fecha:	32
• nueva_persona_profesion:	32
• nueva_profesion:	32
• nuevo_centro:	32
• fecha_correo:	33
• fecha_tel:	33
• nueva_enfermedad:	33
• nueva_persona_enferma:	33
DIAGRAMA DE LA ARQUITECTURA DE LA SOLUCIÓN	34
DIAGRAMA DE LA APLICACIÓN	35

INTRODUCCIÓN

La base de datos registro_covid contendrá los datos de la aplicación web de sistema y registro de vacunación, para el control del proceso de vacunación de la población guatemalteca. La información está archivada en MySQL/MariaDB y el servidor de Apache utilizando xampp para un sistema operativo Windows 10.

La conexión entre el Front End de la aplicación y la base de datos se realiza con mysqli en archivos PHP en el servidor de aplicaciones.

El diseño de la base de datos se adapta a las necesidades de la aplicación y está compuesta por distintos objetos y elementos que se describirán en el presente manual.

SIMBOLOGÍA

En la presente documentación se utilizará la siguiente simbología para la representación de la composición de la base de datos:



Llave primaria de cada una de las tablas.



Llaves foráneas en cada una de las tablas.



Nombre de la tabla.



Nombre de la vista.



Nombre del Procedimiento almacenado.



Nombre de la función.




Nombre del Trigger.

TABLAS

A continuación, se presentará el grupo de tablas pertenecientes a la base de datos registro_covid que contienen toda la información utilizada por la aplicación.



poblacion:

Esta tabla fue creada para almacenar la información personal de cada miembro de la población guatemalteca y se utiliza como tabla maestra de todas las otras tablas que contengan algún tipo de información sobre las personas. Es una tabla con 6 campos y contiene un único índice correspondiente al utilizado para la llave primaria, el cual es la columna dpi. Las especificaciones de los campos de la tabla se muestran a continuación:

Nombre	Tipo de dato	Valores nulos	Descripción
dpi 	bigint(20)	No	Es el identificador único de cada miembro de la población correspondiente a su Documento Personal de Identificación (DPI)
nombre_completo	Varchar(150)	No	Contendrá el nombre completo de cada persona (nombre y apellido).
fecha_nacimiento	date	No	Especificará el día, mes y año de nacimiento de cada poblador.
Genero	Enum ('M', F')	No	Sexo del miembro de la población, M si es hombre (masculino) y F si es Mujer (Femenino).
Nacionalidad	varchar(50)	No	Nacionalidad de la persona.
aplica_vacuna	tinyint(1)	No	Especifica si el miembro de la población está habilitado para poder vacunarse de acuerdo con los parámetros establecidos de edad, enfermedad crónica o trabajo. 1 si aplica, 0 si no.

correos:



La tabla “correos” fue creada para contener la parte de la información de contacto de la población correspondiente a sus emails. Dentro de la plataforma se permite que los pobladores puedan actualizar esta data y registrar por lo menos dos correos. Esta información se utilizará para notificar a los usuarios información cuando sea necesario. Esta tabla cuenta con 3 campos y dos índices: el de la clave primaria y un índice de tipo BTREE para el campo dpi_persona con clave foránea. Las especificaciones de los campos de la tabla se muestran a continuación:

Nombre	Tipo de dato	Valores nulos	Descripción
correo 	varchar(50)	No	Este campo registrará los correos. Estos son únicos, por lo que se utiliza el campo como llave primaria de la tabla.
dpi_persona 	bigint(20)	No	Contendrá el número de Documento Personal de Identificación. Posee una llave foránea de tipo RESTRICT con el campo dpi de la tabla de población
fecha_ingreso	datetime	Sí	Especificará el día, mes, año y hora de creación del registro de cada correo. Esto con la finalidad de ordenarlos correctamente dentro de la plataforma.

telefonos:


La tabla “telefonos” fue creada para contener la parte de la información de contacto de la población correspondiente a su celular, teléfono fijo y otro. Dentro de la plataforma se permite que los pobladores puedan actualizar esta data y registrar por lo menos dos números de teléfono. Esta tabla cuenta con 3 campos y dos índices: el de la clave primaria y un índice

de tipo BTREE para el campo dpi_persona con clave foránea. Las especificaciones de los campos de la tabla se muestran a continuación:

Nombre	Tipo de dato	Valores nulos	Descripción
telefono 	int(11)	No	Este campo registrará los telefonos. Estos son únicos, por lo que se utiliza el campo como llave primaria de la tabla.
dpi_persona 	bigint(20)	No	Contendrá el número de Documento Personal de Identificación. Posee una llave foránea de tipo RESTRICT con el campo dpi de la tabla de población .
fecha_registro	datetime	Sí	Especifica el día, mes, año y hora de creación del registro de cada telefono. Esto con la finalidad de ordenarlos correctamente dentro de la plataforma.

vacunas:

Esta tabla fue creada para contener la información de todas las vacunas disponibles en el país para la población. Estas vacunas serán las únicas disponibles en el sistema. Dentro de la aplicación es posible agregar más vacunas y editar la data de las vacunas existentes. Es una tabla con 5 campos y contiene un único índice correspondiente al utilizado para la llave primaria, el cual es la columna id_vacuna. Las especificaciones de los campos de la tabla se muestran a continuación:


Nombre	Tipo de dato	Valores nulos	Descripción
id_vacuna 	int(11)	No	Es el identificador numérico único de cada vacuna

Nombre	varchar(30)	No	Contiene el nombre de la vacuna registrada.
Laboratorio	varchar(30)	No	Contiene el nombre del laboratorio que creó cada vacuna registrada.
Dosis	int(11)	No	Es el campo que especifica el número de dosis especificadas para aplicarse por cada vacuna registrada. Dentro del portal se permite ingresar y actualizar este valor únicamente con los valores: 1, 2 o 3.
dias_siguiente_dosis	int(11)	No	Es la cantidad de días que se debe esperar para aplicarse la vacuna correspondiente entre una dosis y la siguiente.



puesto_vacunacion:


Esta tabla fue creada para contener la información de todos los centros de vacunación existentes en el país. Estos centros serán los únicos disponibles para escoger en el sistema. Dentro de la aplicación es posible agregar más centros. Es una tabla con 4 campos y contiene un único índice correspondiente al utilizado para la llave primaria, el cual es la columna id_puesto. Las especificaciones de los campos de la tabla se muestran a continuación:

Nombre	Tipo de dato	Valores nulos	Descripción
id_puesto 	int(11)	No	Es el identificador numérico único de cada centro de vacunación.
Nombre	varchar(50)	Sí	Contiene el nombre del centro registrado.

departamento	varchar(50)	Sí	Contiene el nombre del departamento del país en el que está ubicado el centro de vacunación.
Dirección	varchar(100)	Sí	Contiene la dirección completa de la ubicación del centro de vacunación correspondiente.


enfermedad_cronica:

Esta tabla fue creada para contener la información de las enfermedades crónicas registradas en el sistema. Dentro de la aplicación es posible agregar más enfermedades si se considera que hace falta alguna, por ejemplo, para agregarla como parámetro de vacunación. Es una tabla con 3 campos y contiene un único índice correspondiente al utilizado para la llave primaria, el cual es la columna id_enfermedad. Las especificaciones de los campos de la tabla se muestran a continuación:

Nombre	Tipo de dato	Valores nulos	Descripción
id_enfermedad 	int(11)	No	Es el identificador numérico único de cada enfermedad.
nombre_descripcion	varchar(50)	No	Contiene el nombre o la descripción de cada enfermedad crónica.
parametro_actual	tinyint(1)	No	Especifica si el hecho de que alguien padezca esta enfermedad la hace apta para vacunarse o no. 1 si sí es parámetro de vacunación en la actualidad, 0 si no. En la plataforma es posible editar dicho valor conforme avancen las fases de vacunación en el país.

profesiones:




Esta tabla fue creada para contener la información de las profesiones disponibles para seleccionar en el sistema. Dentro de la aplicación es posible agregar más profesiones si se considera que hace falta alguna, por ejemplo, para agregarla como parámetro de vacunación. Es una tabla con 4 campos y contiene un único índice correspondiente al utilizado para el campo de la llave primaria, el cual es la columna id_profesion. Las especificaciones de los campos de la tabla se muestran a continuación:

Nombre	Tipo de dato	Valores nulos	Descripción
id_profesion 	int(11)	No	Es el identificador numérico único de cada profesión.
sector_publico	tinyint(1)	No	Especifica si la profesión pertenece al sector público o no. 1 si pertenece al público, 0 si es sector privado.
Nombre	varchar(50)	No	Contiene el nombre de la profesión registrada.
parametro_vacunación_actual	tinyint(1)	No	Especifica si el hecho de que alguien trabaje en esta profesión la hace apta para vacunarse o no. 1 si sí es parámetro de vacunación en la actualidad, 0 si no. En la plataforma es posible editar dicho valor conforme avancen las fases de vacunación en el país.

poblacion_enferma:


Esta tabla fue creada para almacenar a todas las personas que padecen alguna de las enfermedades registradas en la tabla “enfermedad_cronica”. Es una tabla con 3 campos, cada uno con un índice: el índice con el que se implementa la llave primaria en el campo id_persona_enferma, y dos índices de tipo BTREE en los campos con las llaves foráneas:



dpi_persona e id_enfermedad. Las especificaciones de los campos de la tabla se muestran a continuación:

Nombre	Tipo de dato	Valores nulos	Descripción
id_persona_ enferma 	int(11)	No	Es el identificador numérico único de cada registro.
id_enfermedad 	int(11)	No	Contiene el identificador único de la enfermedad que padece la persona, obtenido de la tabla “enfermedad_cronica”. Posee una llave foránea de tipo RESTRICT con el campo id_enfermedad de dicha tabla.
dpi_persona 	bigint(20)	No	Contiene el número de documento Personal de Identificación de la persona que padece la enfermedad, obtenido de la tabla “poblacion”. Posee una llave foránea de tipo RESTRICT con el campo dpi de dicha tabla.

persona_profesion:


Esta tabla fue creada para almacenar y relacionar a la población con alguna de las profesiones registradas en la tabla “profesiones”. Es una tabla con 3 campos cada uno con un índice: el índice con el que se implementa la llave primaria en el campo id_persona_profesion, y dos índices de tipo BTREE en los campos con las llaves foráneas: dpi_persona e id_profesion. Las especificaciones de los campos de la tabla se muestran a continuación:

Nombre	Tipo de dato	Valores nulos	Descripción
id_persona_ profesion 	int(11)	No	Es el identificador numérico único de cada registro.

id_profesion 	int(11)	No	Contiene el identificador único de la profesión de la persona, traído de la tabla “profesiones”. Posee una llave foránea de tipo RESTRICT con el campo id_profesion de dicha tabla.
dpi_persona 	bigint(20)	No	Contiene el número de documento Personal de Identificación de la persona que padece la enfermedad, obtenido de la tabla “poblacion”. Posee una llave foránea de tipo RESTRICT con el campo dpi de dicha tabla.




tipo_usuario:

La tabla “tipo_usuario” fue creada guardar los 3 tipos de usuarios posibles para el sistema (se podrían agregar más en el futuro). Es una tabla con 3 campos y contiene un único índice correspondiente al utilizado para la llave primaria, el cual es la columna id. Las especificaciones de los campos la tabla se muestran a continuación

Nombre	Tipo de dato	Valores nulos	Descripción
id 	int(11)	No	Es el identificador numérico único de cada tipo de usuario.
Nombre	varchar(30)	No	Guarda el nombre del tipo de usuario al que corresponde cada identificador
Clave	varchar(130)	Sí	Es un campo que contiene la contraseña correspondiente a cada tipo de usuario (si aplica), en caso de que sea un tipo de usuario con permisos más elevados (por ejemplo administrador) que no se desee que tenga cualquier usuario.




usuarios:


Esta tabla fue creada para contener la información de los usuarios ya registrados en el portal. Es una tabla con 5 campos y contiene dos índices; el primero, correspondiente al utilizado para el campo con la llave primaria (que también contiene una llave foránea), que es el campo dpi_usuario y otro de tipo BTREE con el que se implementó la llave foránea en el campo tipo_usuario. Las especificaciones de los campos de la tabla se muestran a continuación:

Nombre	Tipo de dato	Valores nulos	Descripción
dpi_usuario  	bigint(20)	No	Contiene el número de documento Personal de Identificación del usuario, obtenido de la tabla “poblacion”. Posee una llave foránea de tipo RESTRICT con el campo dpi de dicha tabla.
nombre_completo	varchar(150)	No	Contendrá el nombre completo de cada persona (nombre y apellido).
Clave	varchar(130)	No	Contiene la contraseña escogida por cada usuario a la hora de registrarse en el portal (ofuscada)
tipo_usuario 	int(11)	No	Contiene el identificador único del tipo de usuario de la persona, traído de la tabla “tipo_usuario”. Posee una llave foránea de tipo RESTRICT con el campo id de dicha tabla.
fecha_inscripcion	date	Sí	Contiene la fecha en que la persona creó su usuario en el portal de registro.

datos_registro:


Esta tabla fue creada para contener la información relacionada con el proceso de vacunación incluyendo fechas de vacunación, vacuna y centro de vacunación correspondiente para cada usuario existente. Es una tabla con 9 campos y contiene 3 índices; el primero, correspondiente al utilizado para el campo con la llave primaria (que también contiene una llave foránea), que es el campo dpi_persona y los otros dos de tipo BTREE con el que se implementaron dos llaves foráneas en el campo puesto_registro y en el campo vacuna. Las especificaciones de los campos de la tabla se muestran a continuación:



Nombre	Tipo de dato	Valores nulos	Descripción
dpi_persona  	bigint(20)	No	Contiene el número de documento Personal de Identificación del usuario, obtenido de la tabla “usuarios”. Posee una llave foránea de tipo RESTRICT con el campo dpi de dicha tabla.
puesto_registro 	int(11)	No	Contiene el identificador único del centro de vacunación asignado al usuario, traído de la tabla “puesto_vacunacion”. Posee una llave foránea de tipo RESTRICT con el campo id_puesto de dicha tabla.
fecha_primera_dosis	date	No	Especificará el día, mes y año de la cita de vacunación para primera dosis.
primera_dosis	tinyint(1)	Sí	Especifica si la primera dosis ya le fue asignada al usuario o no. 1 es sí y 0 es no.
fecha_segunda_dosis	date	Sí	Especificará el día, mes y año de la cita de vacunación para segunda dosis.

segunda_dosis	tinyint(1)	Sí	Especifica si la segunda dosis ya le fue asignada al usuario o no. 1 es sí y 0 es no.
fecha_tercera_dosis	date	Sí	Especificará el día, mes y año de la cita de vacunación para tercera dosis.
tercera_dosis	tinyint(1)	Sí	Especifica si la tercera dosis ya le fue asignada al usuario o no. 1 es sí y 0 es no.
vacuna 	int(11)	Sí	Contiene el identificador único de la vacuna asignada al usuario, traído de la tabla “vacunas”. Posee una llave foránea de tipo RESTRICT con el campo id_vacuna de dicha tabla.

pdf_proceso:



Esta tabla es utilizada para guardar los datos de los documentos en PDF creados por los usuarios cuando cada vez que quieren descargar el progreso de su proceso de vacunación. Es una tabla con 10 campos y contiene 3 índices; el primero, correspondiente al utilizado para el campo con la llave primaria, que es el campo id_archivo y los otros dos de tipo BTREE con el que se implementaron dos llaves foráneas en el campo dpi_persona y en el campo vacuna. Las especificaciones de los campos de la tabla se muestran a continuación:

Nombre	Tipo de dato	Valores nulos	Descripción
id_archivo 	bigint(20)	No	Contiene el identificador único del documento (un número de 20 dígitos) con el que el usuario puede identificar el documento descargado y la validez de los datos que mostrados en el mismo.

dpi_persona 	bigint(20)	No	Contiene el número de documento Personal de Identificación del usuario, obtenido de la tabla “usuarios”. Posee una llave foránea de tipo RESTRICT con el campo dpi de dicha tabla.
fecha_primera_dosis	date	Sí	Especificará el día, mes y año de la cita de vacunación para primera dosis.
Primera_dosis	tinyint(1)	Sí	Especifica si la primera dosis ya le fue asignada al usuario o no. 1 es sí y 0 es no.
Fecha_segunda_dosis	date	Sí	Especificará el día, mes y año de la cita de vacunación para segunda dosis.
Segunda_dosis	tinyint(1)	Sí	Especifica si la segunda dosis ya le fue asignada al usuario o no. 1 es sí y 0 es no.
Fecha_tercera_dosis	date	Sí	Especificará el día, mes y año de la cita de vacunación para tercera dosis.
Tercera_dosis	tinyint(1)	Sí	Especifica si la tercera dosis ya le fue asignada al usuario o no. 1 es sí y 0 es no.
vacuna 	int(11)	Sí	Contiene el identificador único de la vacuna asignada al usuario, traído de la tabla “vacunas”. Posee una llave foránea de tipo RESTRICT con el campo id_vacuna de dicha tabla.
fecha_creacion	datetime	No	Contiene el día, mes, año y hora en que el documento fue creado por el usuario.


solicitudes_empleados:

Esta tabla contiene a los usuarios (tipo_usuario 1) que han enviado una solicitud al administrador para obtener permisos de empleado en el portal. Es una tabla con 2 campos y contiene un solo índice en el campo de la llave primaria (que también tiene una llave foránea), correspondiente a la columna dpi_empleado. Las especificaciones de los campos de la tabla se muestran a continuación:

Nombre	Tipo de dato	Valores nulos	Descripción
dpi_persona  	bigint(20)	No	Contiene el número de documento Personal de Identificación del usuario, obtenido de la tabla “poblacion”. Posee una llave foránea de tipo RESTRICT con el campo dpi de dicha tabla.
Nombre_completo	varchar(150)	No	Guarda el nombre completo del usuario que hizo la solicitud a través del portal.

imagen_descripcion:


Esta tabla fue creada para guardar los datos de los elementos de la página de noticias de tipo imagen con descripción. Es una tabla con 3 campos y contiene un solo índice en el campo de la llave primaria, correspondiente a la columna id_imagen_desc. Las especificaciones de los campos de la tabla se muestran a continuación:

Nombre	Tipo de dato	Valores nulos	Descripción
id_imagen_desc 	int(11)	No	Contiene el identificador único de la imagen con descripción.
imagen	varchar(100)	No	Guarda el nombre de la imagen correspondiente al elemento guardada en el servidor.

descripcion	varchar(100)	No	Guarda la descripción que se mostrará bajo la imagen en el portal.
-------------	--------------	----	--


imagen_texto_lado:

Esta tabla fue creada para guardar los datos de los elementos de la página de noticias de tipo imagen con texto a la derecha e imagen con texto a la izquierda. Es una tabla con 4 campos y contiene un solo índice en el campo de la llave primaria, correspondiente a la columna id_imagen_texto. Las especificaciones de los campos de la tabla se muestran a continuación:

Nombre	Tipo de dato	Valores nulos	Descripción
id_imagen_texto 	int(11)	No	Contiene el identificador único de la imagen con texto a un lado.
imagen	varchar(100)	No	Guarda el nombre de la imagen correspondiente al elemento, guardada en el servidor.
texto	varchar(2000)	No	Guarda el texto que se colocará al lado de la imagen en el portal.
derecha	tinyint(1)	No	Especifica el lado al que irá la imagen en el elemento. 1 corresponde a la derecha y 0 a la izquierda.

titulo_texto:

Esta tabla fue creada para guardar los datos de los elementos de la página de noticias de tipo título y texto html. Es una tabla con 3 campos y contiene un solo índice en el campo de la llave primaria, correspondiente a la columna id_titulo. Las especificaciones de los campos de la tabla se muestran a continuación:

Nombre	Tipo de dato	Valores nulos	Descripción
id_imagen_texto 	int(11)	No	Contiene el identificador único del título o texto.
texto	varchar(1000)	No	Guarda el texto que se colocará al elemento, ya sea título o texto.
título	tinyint(1)	No	Especifica si el elemento será de tipo título o de tipo texto html. 1 corresponde a título y 0 a texto.

VISTAS:

A continuación, se presentará el grupo de vistas pertenecientes a la base de datos registro_covid.

cita_perdida:

Esta vista se utilizará en el módulo de reportes del portal para consultar las personas que no acudieron a alguna de las citas de vacunación que tenían programadas. La vista contiene 6 campos obtenidos de las uniones de consultas de la tabla “datos_registro” y funciones para obtener datos de otras tablas. La información de los campos de la vista se muestra a continuación:

Nombre	Tipo de dato	Descripción
fecha_dosis	date	Fecha de la dosis saltada datos obtenidos de los campos fecha_primera_dosis, fecha_segunda_dosis y fecha_tercera_dosis de la tabla “datos_registro”.
puesto_registro	int(11)	Puesto de vacunación asignado al usuario, obtenido del campo id_puesto de la tabla “datos_registro”.
nombre_centro	varchar_50	Nombre del centro de vacunación asignado al usuario. Obtenido de la tabla puesto_vacunacion.

dpi_persona	bigint(20)	Número de dpi del usuario, obtenido del campo dpi_persona de la tabla “datos_registro”.
nombre	varchar(50)	Nombre completo de la persona, obtenido utilizando la función nombre_persona (explicada posteriormente) con información de la tabla “poblacion”.
dosis_saltada	varchar(7)	Dosis para la cual el usuario no acudió a su cita, puede ser primera segunda o tercera. Obtenido de los campos de fecha y dosis de la tabla “datos_registro”.

info_usuarios:

Esta vista se utilizará en el perfil de usuario para que este pueda consultar todos sus datos. Es decir, incluye toda la información relacionada con su dpi en la base de datos y posee 24 campos. La información de los campos se muestra a continuación:

Nombre	Tipo de dato	Descripción
dpi	bigint(20)	Tabla “poblacion”.
nombre_completo	varchar(150)	Tabla “poblacion”.
fecha_nacimiento	date	Tabla “poblacion”.
genero	Enum (‘M’, F’)	Tabla “poblacion”.
nacionalidad	varchar(50)	Tabla “poblacion”.
aplica_vacuna	tinyint(1)	Tabla “poblacion”.
dpi_persona	bigint(20)	Tabla “datos_registro”.
puesto_registro	int(11)	Tabla “datos_registro”.
fecha_primera_dosis	date	Tabla “datos_registro”.
primera_dosis	tinyint(1)	Tabla “datos_registro”.
fecha_segunda_dosis	date	Tabla “datos_registro”.
segunda_dosis	tinyint(1)	Tabla “datos_registro”.
fecha_tercera_dosis	date	Tabla “datos_registro”.
tercera_dosis	tinyint(1)	Tabla “datos_registro”.

vacuna	int(11)	Tabla “datos_registro”.
centro	varchar(50)	Campo “nombre” tabla “puesto_vacunacion”
enfermedad	varchar(50)	Campo “nombre_descripcion” tabla “enfermedad_cronica”.
nombre_vacuna	varchar(30)	Campo “nombre” tabla “vacunas”.
tipo_usuario	int(11)	Tabla “usuarios”
tel2	int(11)	Teléfono más reciente tabla “teléfonos”.
tel1	int(11)	Teléfono más antiguo tabla “teléfonos”.
email1	varchar(50)	Correo más antiguo tabla “correos”.
email2	varchar(50)	Correo más reciente tabla “correos”.
profesion	varchar(50)	Campo “nombre” tabla “profesiones”.

reportes_centro:

Esta vista se utilizará en el perfil de usuario para que este pueda consultar todos sus datos. Es decir, incluye toda la información relacionada con su dpi en la base de datos y posee 24 campos. La información de los campos se muestra a continuación:

Nombre	Tipo de dato	Descripción
fecha_dosis	date	Fecha de la dosis saltada datos obtenidos de los campos fecha_primera_dosis, fecha_segunda_dosis y fecha_tercera_dosis de la tabla “datos_registro”.
puesto_registro	int(11)	Puesto de vacunación asignado al usuario, obtenido del campo id_puesto de la tabla “datos_registro”.
nombre_centro	varchar_50	Nombre del centro de vacunación asignado al usuario. Obtenido de la tabla puesto_vacunacion.
dpi_persona	bigint(20)	Número de dpi del usuario, obtenido del campo dpi_persona de la tabla “datos_registro”.
vacuna	varchar(50)	Campo id_vacuna de la tabla “vacunas”
nombre_vacuna	varchar(30)	Campo “nombre” de la tabla “vacunas”.

genero	char(1)	Tabla “poblacion”.
dosis_aplicada	varchar(7)	Dosis que le fue aplicada al usuario en dicho centro y fecha, puede ser primera segunda o tercera. Obtenido de los campos de fecha y dosis de la tabla “datos_registro”.

FUNCIONES

Para la correcta resolución las necesidades de la aplicación, se utilizaron distintas funciones; estas se muestran a continuación.

fo correo1_persona:

Parámetros que recibe:

- ✓ dpi_persona_p: con tipo de dato bigint(20) y dirección IN.

Retorna:

- ✓ Tipo de dato: varchar(50).
- ✓ Dato: el correo más antiguo del usuario con dpi = dpi_persona_p.

fo correo2_persona:

Parámetros que recibe:

- ✓ dpi_persona_p: con tipo de dato bigint(20) y dirección IN.

Retorna:

- ✓ Tipo de dato: varchar(50).
- ✓ Dato: el correo más reciente del usuario con dpi = dpi_persona_p.

fo Count_usuario:

Parámetros que recibe:

- ✓ dpi_persona_p: con tipo de dato bigint(20) y dirección IN.

Retorna:

- ✓ Tipo de dato: int(11)

- ✓ Dato: el conteo de la cantidad de usuarios existentes en la tabla de usuarios con `dpi_usuario = dpi_persona_p`.

fo **dias_vacuna:**

Parámetros que recibe:

- ✓ `id_vacuna_p`: con tipo de dato `int(11)` y dirección IN.
- ✓ `fecha_p`: con tipo de dato `date` y dirección IN.

Retorna:

- ✓ Tipo de dato: `date`
- ✓ Dato: la fecha correspondiente a la siguiente dosis luego de sumarle los días de intervalo entre dosis correspondientes a la vacuna con `id_vacuna = id_vacuna_p` a la `fecha_p`.

fo **dosis_vacuna:**

Parámetros que recibe:

- ✓ `vacuna_p`: con tipo de dato `int(11)` y dirección IN.

Retorna:

- ✓ Tipo de dato: `int(11)`
- ✓ Dato: la cantidad de dosis que corresponden a la vacuna con `id_vacuna = vacuna_p`.

fo **enfermedad:**

Parámetros que recibe:

- ✓ `dpi_persona_p`: con tipo de dato `bigint(20)` y dirección IN.

Retorna:

- ✓ Tipo de dato: `varchar(50)`
- ✓ Dato: el nombre de la enfermedad crónica que padece la persona con `dpi_persona = dpi_persona_p`.

fo **enfermedad_id:**

Parámetros que recibe:

- ✓ `dpi_persona_p`: con tipo de dato `bigint(20)` y dirección IN.

Retorna:

- ✓ Tipo de dato: int(11)
- ✓ Dato: el nombre id de la enfermedad crónica que padece la persona con dpi_persona = dpi_persona_p.

fo **es_usuario:**

Parámetros que recibe:

- ✓ dpi_p: con tipo de dato bigint(20) y dirección IN.

Funcionamiento:

- ✓ Realiza un conteo de la cantidad usuarios con dpi_usuario = dpi_p.

Retorna:

- ✓ Tipo de dato: tinyint(1)
- ✓ Dato: 1 si el conteo realizado es 1, 0 de lo contrario.

fo **genero_persona:**

Parámetros que recibe:

- ✓ dpi_p: con tipo de dato bigint(20) y dirección IN.

Retorna:

- ✓ Tipo de dato: enum('M', 'F')
- ✓ Dato: el sexo de la persona con dpi = dpi_p.

fo **nombre_centro:**

Parámetros que recibe:

- ✓ dpi_persona_p: con tipo de dato bigint(20) y dirección IN.

Retorna:

- ✓ Tipo de dato: varchar(50)
- ✓ Dato: el nombre del centro de vacunación asignado a la persona con DPI = dpi_persona_p.

fo **nombre_persona:**

Parámetros que recibe:

- ✓ dpi_persona_p: con tipo de dato bigint(20) y dirección IN.

Retorna:

- ✓ Tipo de dato: varchar(50)
- ✓ Dato: el nombre completo de la persona con DPI = dpi_persona_p.

fo **nombre_vacuna:**

Parámetros que recibe:

- ✓ id_vacuna_p: con tipo de dato int(11) y dirección IN.

Retorna:

- ✓ Tipo de dato: varchar(30)
- ✓ Dato: el nombre de la vacuna con id_vacuna = id_vacuna_p.

fo **profesion_persona:**

Parámetros que recibe:

- ✓ dpi_persona_p: con tipo de dato bigint(20) y dirección IN.

Retorna:

- ✓ Tipo de dato: varchar(50)
- ✓ Dato: el nombre de la profesión de la persona con DPI = dpi_persona_p.

fo **profesion_persona_id:**

Parámetros que recibe:

- ✓ dpi_persona_p: con tipo de dato bigint(20) y dirección IN.

Retorna:

- ✓ Tipo de dato: int(11)
- ✓ Dato: el id de la profesión de la persona con DPI = dpi_persona_p.

fo **tell_persona:**

Parámetros que recibe:

- ✓ dpi_persona_p: con tipo de dato bigint(20) y dirección IN.

Retorna:

- ✓ Tipo de dato: int(11)
- ✓ Dato: el teléfono más antiguo del usuario con DPI = dpi_persona_p.

fo **tel2_persona:**

Parámetros que recibe:

- ✓ dpi_persona_p: con tipo de dato bigint(20) y dirección IN.

Retorna:

- ✓ Tipo de dato: int(11)
- ✓ Dato: el teléfono más reciente del usuario con DPI = dpi_persona_p.

fo **tipo_usuario:**

Parámetros que recibe:

- ✓ dpi_p: con tipo de dato bigint(20) y dirección IN.

Retorna:

- ✓ Tipo de dato: int(11)
- ✓ Dato: el id del tipo de usuario registrado de la persona con DPI = dpi_p.

PROCEDIMIENTOS ALMACENADOS

Para lograr cumplir con las necesidades y requerimientos de la aplicación, se utilizaron distintos procedimientos almacenados en la base de datos; estos se muestran a continuación.

Po **actualizar_registro:**

Parámetros que recibe:

- ✓ dpi_p: con tipo de dato bigint(20) y dirección IN.
- ✓ tel1_p: con tipo de dato int(11) y dirección IN.
- ✓ tel2_p: con tipo de dato int(11) y dirección IN.
- ✓ email1_p: con tipo de dato varchar(50) y dirección IN.
- ✓ email2_p: con tipo de dato varchar(50) y dirección IN.

Funcionamiento: el procedimiento cuenta la cantidad de registros hay en las tablas de correos y teléfonos de la persona con DPI = dpi_p. Dependiendo de si la cantidad es 0, 1 o 2, inserta

o actualiza los registros con los nuevos datos enviados como parámetros (tel1_p y tel2_p, email1_p y email2_p) en las tablas “telefonos” y “correos” respectivamente. Actualizando así, la información de contacto de la persona.

Po add_persona:

Parámetros que recibe:

- ✓ dpi_p: con tipo de dato bigint(20) y dirección IN.
- ✓ nombre_p: con tipo de dato varchar(150) y dirección IN.
- ✓ nacionalidad_p: con tipo de dato varchar(50) y dirección IN.
- ✓ genero_p: con tipo de dato enum('M','F') y dirección IN.
- ✓ fecha_p: con tipo de dato date y dirección IN.
- ✓ trabajo_p: con tipo de dato int(11) y dirección IN.
- ✓ enfermedad_p: con tipo de dato int(11) y dirección IN.
- ✓ telefono_p: con tipo de dato int(11) y dirección IN.
- ✓ correo_p: con tipo de dato varchar(50) y dirección IN.

Funcionamiento: realiza inserts en diferentes tablas añadiendo toda la información de población, contacto, enfermedad y profesión de la persona con DPI = dpi_p. Es decir, información para las tablas “poblacion”, “persona_profesion”, “poblacion_enferma”, “correos” y “telefonos” para la persona con dicho dpi.

Po aprobar_solicitud:

Parámetros que recibe:

- ✓ dpi_p: con tipo de dato bigint(20) y dirección IN.

Funcionamiento: actualiza los permisos del usuario con dpi_usuario = dpi_p para que tenga permisos de usuario empleado (2). Luego, elimina la solicitud de permiso de la tabla de solicitudes, dado que ya fue aceptada.

Po delete_usuario:

Parámetros que recibe:

- ✓ dpi_p: con tipo de dato bigint(20) y dirección IN.

Funcionamiento: elimina los datos de registro y el usuario registrado en el portal de la persona con DPI = dpi_p.

Po generar_pdf:

Parámetros que recibe:

- ✓ dpi_p: con tipo de dato bigint(20) y dirección IN.

Funcionamiento: el procedimiento toma los datos actuales de la tabla “datos_registro” de la persona con DPI = dpi_p, correspondientes al estado actual de su proceso de vacunación además de la fecha de inscripción de su usuario. Luego, genera un número aleatorio de 20 dígitos (que será el identificador del documento). Después revisa si ya existe un documento con los datos actualizados creado anteriormente. De ser así, no se hace ningún cambio a la data; de lo contrario se hace un insert en la tabla “pdf_proceso” con los datos actuales del proceso y se inserta el número aleatorio como id del nuevo archivo.

Po generar_pdf:

Parámetros que recibe:

- ✓ dpi_p: con tipo de dato bigint(20) y dirección IN.

Funcionamiento: el procedimiento toma los datos actuales de la tabla “datos_registro” de la persona con DPI = dpi_p, correspondientes al estado actual de su proceso de vacunación además de la fecha de inscripción de su usuario. Luego, genera un número aleatorio de 20 dígitos (que será el identificador del documento). Después revisa si ya existe un documento con los datos actualizados creado anteriormente. De ser así, no se hace ningún cambio a la data; de lo contrario se hace un insert en la tabla “pdf_proceso” con los datos actuales del proceso y se inserta el número aleatorio como id del nuevo archivo.

Po habilitar_edad:

Parámetros que recibe:

- ✓ fecha_p: con tipo de dato varchar(10) y dirección IN.
- ✓ simbolo_p: con tipo de dato enum(‘=’, ‘<’, ‘>’) y dirección IN.

Funcionamiento: el procedimiento habilita a la población con una fecha de nacimiento mayor, menor o igual (dependiendo del valor de símbolo_p) a la fecha_p para que puedan registrarse en el portal y poder recibir una cita de vacunación.

Po habilitar_enfermedad:

Parámetros que recibe:

- ✓ id_enf_p: con tipo de dato int(11) y dirección IN.

Funcionamiento: el procedimiento actualiza el registro de la enfermedad crónica con id_enfermedad = id_enf_p para hacerla un parámetro de habilitación de vacunación actual. Posteriormente, habilita a la población que padece dicha enfermedad para poder registrarse en el portal y poder recibir una cita de vacunación.

Po habilitar_trabajo:

Parámetros que recibe:

- ✓ id_prof_p: con tipo de dato int(11) y dirección IN.

Funcionamiento: el procedimiento actualiza el registro de la profesión con id_profesion = id_prof_p para hacerla un parámetro de habilitación de vacunación actual. Posteriormente, habilita a la población que ejerce dicha profesión para poder registrarse en el portal y poder recibir una cita de vacunación.

Po insertVacunas:

Parámetros que recibe:

- ✓ nombre_p: con tipo de dato varchar(30) y dirección IN.
- ✓ laboratorio_p: con tipo de dato varchar(30) y dirección IN.
- ✓ dosis_p: con tipo de dato int(11) y dirección IN.
- ✓ dias_p: con tipo de dato int(11) y dirección IN.

Funcionamiento: el procedimiento inserta un nuevo registro en la tabla “vacunas” con la información de los parámetros enviados, utilizando como id_vacuna el siguiente número luego del id del último registro.

Po nuevo_registro:

Parámetros que recibe:

- ✓ dpi_p: con tipo de dato bigint(20) y dirección IN.
- ✓ tel1_p: con tipo de dato int(11) y dirección IN.
- ✓ tel2_p: con tipo de dato int(11) y dirección IN.
- ✓ email1_p: con tipo de dato varchar(50) y dirección IN.
- ✓ email2_p: con tipo de dato varchar(50) y dirección IN.
- ✓ centro_p: con tipo de dato int(11) y dirección IN.

Funcionamiento: el procedimiento crea un nuevo registro en la tabla “datos_registro” para el usuario con dpi_usuario = dpi_p asignándole su puesto de registro correspondiente (centro_p). Luego, procede a actualizar la información de contacto (teléfonos y correos) del usuario en las tablas correspondientes con los valores de los parámetros tel1_p, tel2_p, email1_p, email2_p.

Po nuevo_usuario:

Parámetros que recibe:

- ✓ dpi_p: con tipo de dato bigint(20) y dirección IN.
- ✓ password_p: con tipo de dato varchar(130) y dirección IN.
- ✓ tipo_usuario_p: con tipo de dato int(11) y dirección IN.

Funcionamiento: el procedimiento revisa que la persona con dpi=dpi_p esté habilitado para poder registrarse a vacunación, si este es el caso crea un nuevo usuario con dpi_usuario = dpi_p, tipo_usuario = tipo_usuario_p, contraseña password_p y fecha de inscripción = fecha actual; en caso contrario hace un “SIGNAL” para generar un error.

Po solicitud_empleado:

Parámetros que recibe:

- ✓ dpi_p: con tipo de dato bigint(20) y dirección IN.
- ✓ password_p: con tipo de dato varchar(130) y dirección IN.

Funcionamiento: el procedimiento crea una nueva solicitud de permisos de empleado para la persona con dpi = dpi_p. Luego llama al procedimiento nuevo_usuario con el parámetro tipo_usuario_p = 1, hasta que la solicitud sea aprobada.

update_poblacion:

Parámetros que recibe:

- ✓ dpi_p: con tipo de dato bigint(20) y dirección IN.
- ✓ nombre_p: con tipo de dato varchar(150) y dirección IN.
- ✓ nacionalidad_p: con tipo de dato varchar(50) y dirección IN.
- ✓ genero_p: con tipo de dato enum('M','F') y dirección IN.
- ✓ fecha_p: con tipo de dato date y dirección IN.
- ✓ trabajo_p: con tipo de dato int(11) y dirección IN.
- ✓ enfermedad_p: con tipo de dato int(11) y dirección IN.

Funcionamiento: el procedimiento actualiza los datos de la persona con dpi=dpi_p en la tabla de población al igual que los registros en las tablas “persona_profesion” y “poblacion_enferma”, cambiando los datos por los de los parámetros enviados en las tablas correspondientes.

TRIGGERS

A continuación, se presentará el grupo de triggers utilizados la base de datos registro_covid, con sus tablas asociadas.

aplica_v_enf:

- ✓ Tabla asociada: poblacion_enferma
- ✓ Tiempo: BEFORE
- ✓ Evento: INSERT
- ✓ Función: el trigger observa si la enfermedad con el ID igual al del id de la enfermedad del nuevo registro que se insertará es parámetro de vacunación actual. De ser así, se actualizará la tabla de población para habilitar a la persona con el dpi igual al dpi del nuevo registro para que pueda registrarse en el portal y poder recibir una cita de vacunación en caso de que no estuviera habilitada ya. Si la enfermedad no es parámetro de vacunación no se realiza ninguna acción.

aplica_v_prof:

- ✓ Tabla asociada: persona_profesion

- ✓ Tiempo: BEFORE
- ✓ Evento: INSERT
- ✓ Función: el trigger observa si la profesión con el ID igual al del id de la profesión del nuevo registro que se insertará es parámetro de vacunación actual. De ser así, se actualizará la tabla de población para habilitar a la persona con el dpi igual al dpi del nuevo registro para que pueda registrarse en el portal y poder recibir una cita de vacunación, en caso de que no estuviera habilitada ya. Si la profesión no es parámetro de vacunación no se realiza ninguna acción.

asignar_fecha:

- ✓ Tabla asociada: datos_registro
- ✓ Tiempo: BEFORE
- ✓ Evento: UPDATE
- ✓ Función: una vez que una dosis de vacunación es aplicada a un usuario, el trigger asigna automáticamente la fecha de la siguiente dosis (si quedan aún) de acuerdo con la cantidad de dosis y días de intervalo entre dosis de la vacuna aplicada a la persona.

nueva_persona_profesion:

- ✓ Tabla asociada: persona_profesion
- ✓ Tiempo: BEFORE
- ✓ Evento: INSERT
- ✓ Función: cada vez que se crea un registro en la tabla persona_profesion se le asigna como valor de id_persona_profesion el número siguiente al id del último registro.

nueva_profesion:

- ✓ Tabla asociada: profesiones
- ✓ Tiempo: BEFORE
- ✓ Evento: INSERT
- ✓ Función: cada vez que se crea un registro en la tabla profesiones se le asigna como valor de id_profesion el número siguiente al id del último registro.

nuevo_centro:

- ✓ Tabla asociada: puesto_vacunacion
- ✓ Tiempo: BEFORE

- ✓ Evento: INSERT
- ✓ Función: cada vez que se crea un registro en la tabla puesto_vacunacion se le asigna como valor de id_puesto el número siguiente al id del último registro.

fecha_correo:

- ✓ Tabla asociada: correos
- ✓ Tiempo: BEFORE
- ✓ Evento: INSERT
- ✓ Función: cada vez que se crea un registro en la tabla de correos con el campo de fecha_ingreso como NULL se le asigna como valor la fecha y hora actual.

fecha_tel:

- ✓ Tabla asociada: telefonos
- ✓ Tiempo: BEFORE
- ✓ Evento: INSERT
- ✓ Función: cada vez que se crea un registro en la tabla de telefonos con el campo de fecha_registro como NULL se le asigna como valor la fecha y hora actual.

nueva_enfermedad:

- ✓ Tabla asociada: enfermedad_cronica
- ✓ Tiempo: BEFORE
- ✓ Evento: INSERT
- ✓ Función: cada vez que se crea un registro en la tabla enfermedad_cronica se le asigna como valor de id_enfermedad el número siguiente al id del último registro.

nueva_persona_enferma:

- ✓ Tabla asociada: poblacion_enferma
- ✓ Tiempo: BEFORE
- ✓ Evento: INSERT
- ✓ Función: cada vez que se crea un registro en la tabla poblacion_enferma se le asigna como valor de id_persona_enferma el número siguiente al id del último registro.

DIAGRAMA DE LA ARQUITECTURA DE LA SOLUCIÓN

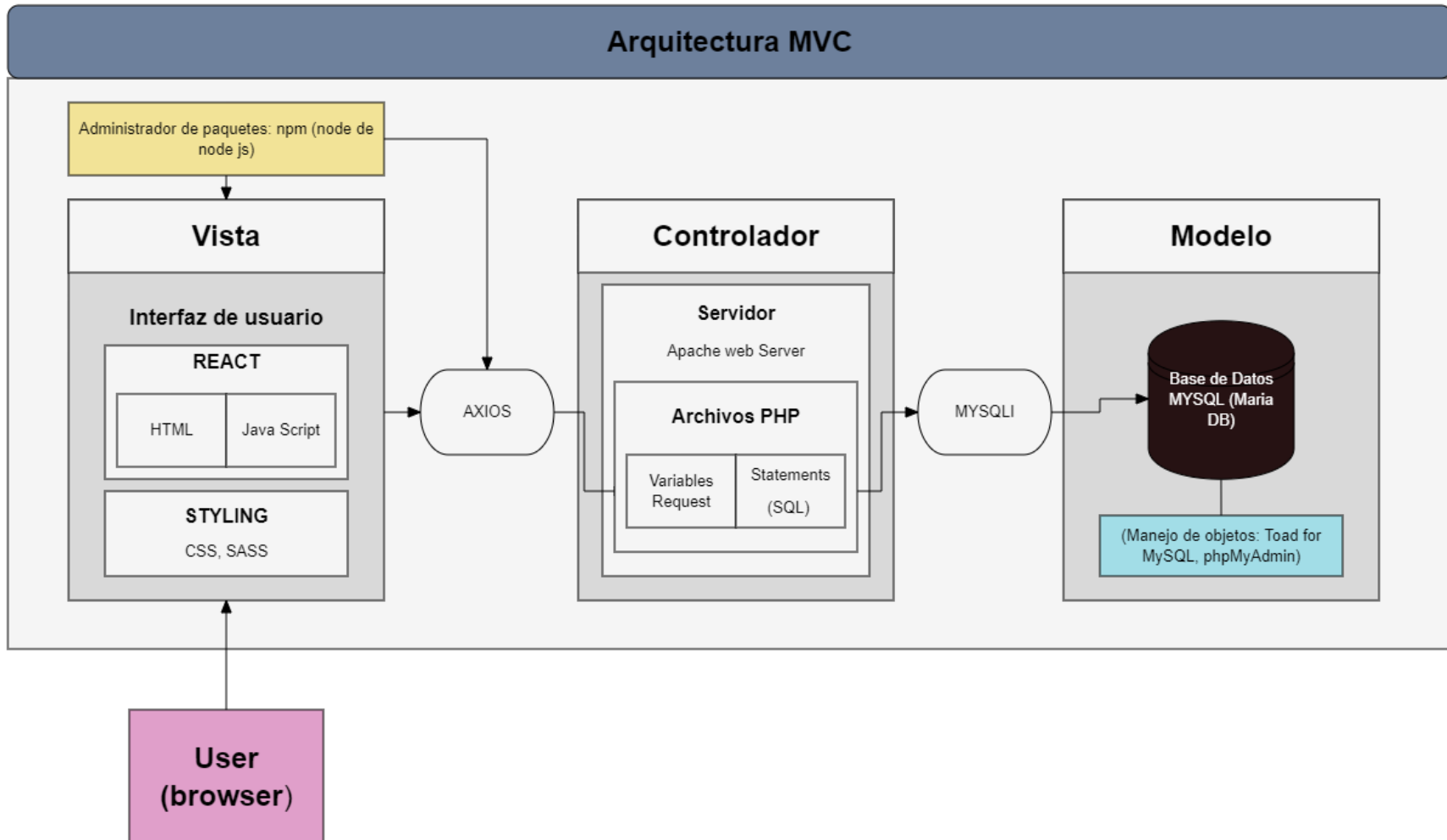


DIAGRAMA DE LA APLICACIÓN

