Manual De Configuración Base de Datos

DIGITURNO SMS MADS

Versión: 1.2

**HISTORIAL DE REVISIÓN**

| **VERSIÓN** | **ELABORACIÓN** | | **REVISIÓN** | | **APROBACIÓN** | |
| --- | --- | --- | --- | --- | --- | --- |
| **Fecha** | **Responsable** | **Fecha** | **Responsable** | **Fecha** | **Responsable** |
| V-1.0 | 15/09/2021 | ALEXIS CARREÑO | 15/09/2021 | MONICA ARDILA |  |  |
| V-1.1 | 24/02/2022 | ALEXIS CARREÑO | 24/02/2022 | MONICA ARDILA |  |  |
| V-1.2 | 25/02/2022 | ALEXIS CARREÑO | 25/02/2022 | MONICA ARDILA |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

**CAMBIOS RESPECTO A LA VERSIÓN ANTERIOR**

| **VERSIÓN** | **MODIFICACIÓN RESPECTO VERSIÓN ANTERIOR** |
| --- | --- |
| V-1.0 | Documento inicial |
| V-1.1 | se empieza a trabajar en el documento con el formato del proyecto |
| V-1.2 | Se actualizan las tablas de la base de datos, los schemas, y las relaciones que estaban pendientes. |
| V-1.3 | Se realiza definición de las abreviaturas. |
|  |  |
|  |  |

**Tabla de contenido**

[1. Introducción](#_heading=h.30j0zll) **4**

[2. Alcance](#_heading=h.1fob9te) **5**

[**3. Definiciones siglas y abreviaturas**](#_heading=h.eujw391oa34r) **5**

[4. Responsables e involucrados](#_heading=h.3znysh7) **6**

[**5. Modelo Entidad Relación (MER)**](#_heading=h.jotps66yjq2f) **6**

[6. Diccionario de Datos.](#_heading=h.tyjcwt) **8**

[**7. Modelo Relacional.**](#_heading=h.6ghq5u95lyju) **12**

[8. Justificación Motor Seleccionado](#_heading=h.1t3h5sf) **13**

[9. Requisitos de Configuración](#_heading=h.4d34og8) **13**

[**10. Scripts**](#_heading=h.cg36x54ml892) **13**

[**12. Configuración y Ejecución de la Base de Datos**](#_heading=h.drflwrnhspdj) **31**

[13. Otras Consideraciones](#_heading=h.3rdcrjn) **32**

# 1. Introducción

En colombia existe una problemática con respecto a la sobre asignación de turnos en diferentes entidades públicas y privadas, este es un problema que ha existido desde siempre y las empresas tratan de adaptar diferentes sistemas o estrategias que ayuden a organizar y reducir los tiempos de espera y las largas filas, esta problemática creció más durante la pandemia, entendiendo esto desde el distanciamiento social y evitar el sobre cupo en los establecimientos públicos.

Los usuarios pasan por extensos tiempos de espera dejando de la lado acciones adicionales, en donde si dejan su turno de espera pueden perder el llamado haciendo que el tiempo invertido en la espera sea ineficiente, el tiempo promedio de atención oscila entre los 25 y 45 minutos (según datos recopilados), aunque en algunos casos solo son de 5 minutos teniendo en cuenta que existen turnos de consulta rápida.

Es por eso que encontramos una forma de reducir estas filas, en el que los usuarios evitan aglomeraciones y puedan realizar tranquilamente otras diligencias cercanas o ubicarse en un sitio cercano a su cita o lugar de diligencia sin perder el turno y su tiempo de espera.

DIGITURNO SMS MADS ofrece mediante su web service, la opción para que las empresas puedan construir sus propias soluciones, en donde DIGITURNO SMS MADS ofrecerá el gateway de sms con patrones definidos y necesarios para que puedan construir herramientas completas y robustas.

De esta forma pacientes, usuarios y clientes que tienen turnos muy largos podrán estar realizando otras actividades como realizar diligencias, tomar un café en un sitio cercano sin correr el riesgo de perder el llamado de su turno.

Como plus adicional a nuestros clientes, mediante la definición y la selección de los paquetes, ellos pueden usar la cantidad de mensajes que quieran usar por usuario.

# 2. Alcance

Con la creación de las soluciones corporativas, en las cuales DIGITURNO SMS MADS, quiere ser pionera mediante el uso de tecnologías existentes, se establece la creación de tres aspectos importantes, teniendo en cuenta que cada uno de ellos cumple funciones indispensables en el desarrollo del proyecto.

Página WEB: Mostrará la información de la empresa, cada uno de los planes con los que los clientes pueden diseñar sus paquetes, e implementar la solución que quieren para sus usuarios.

“les permita desarrollar sus soluciones a partir del servicio REST que tenemos expuesto para las empresas que desean consumir”.

para la implementación de la página, se utilizarán herramientas que da al usuario un óptimo desempeño, estas tecnologías son: HTML, CSS, JAVASCRIPT, REACT

DashBoard / Cliente: conecta la información del API con los datos reales mediante su consumo, adicional a esto mostrará el detalle de cada consumo y el consolidado los recursos disponibles (en esta caso los mensajes), esta sección del servicio está disponible, para las empresas o clientes corporativos que usen el servicio, y no al cliente final (usuarios).

PHP laravel, Composer, HTML y CSS.

API / DIGITURNO SMS MADS: Este es un servicio, que permitirá el consumo entre el DashBoard y los diferentes elementos del sistema, permitiendo la integración de los elementos.

Desde el punto de vista geográfico, se tiene que inicialmente la cobertura se realizará a nivel regional, que permita cubrir los diferentes establecimientos, y en estos primeros clientes llevar un monitoreo constante, dentro de los primeros 6 meses, mediante este monitoreo los resultados permitirán tener los primeros resultados, que darán el índice de efectividad del servicio, refinando las posibles fallas que se puedan presentar. Permitiendo empezar a cubrir el territorio nacional.

# 3. Definiciones siglas y abreviaturas

**Triggers:** Un trigger o disparador es un objeto que se asocia con tablas y se almacena en la base de datos. Su nombre se deriva por el comportamiento que presentan en su funcionamiento, ya que se ejecutan cuando sucede algún evento sobre las tablas a las que se encuentra asociado.

**Incremental:** Los datos incrementales, son aquellos que van creciendo conforme se van anexando en las tablas de datos.

**Roles:** Los roles de base de datos simplifican el proceso de gestión de privilegios, ya que se pueden otorgar privilegios a un rol y luego otorgar el rol a usuarios. Cuando desee revocar privilegios para un usuario, simplemente tiene que revocar la autorización de rol del usuario, en vez de revocar cada privilegio individual.

**Permisos:** Los permisos de Motor de base de datos se administran en el nivel de servidor asignados a los inicios de sesión y roles de servidor, y en el nivel de base de datos asignados a usuarios de base de datos y roles base de datos.

# Scripts: Un script, secuencia de comandos​ o guión​​​ (traduciendo desde inglés) es un término informal que se usa para designar a un programa relativamente simple.

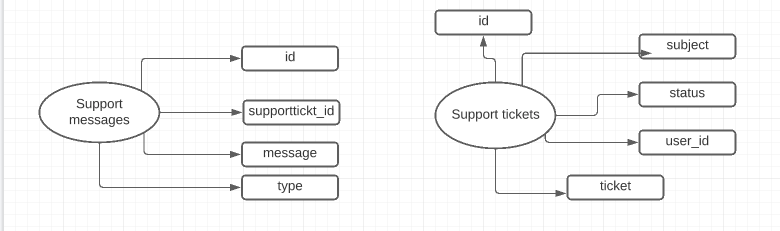
**Logs:** Explican el comportamiento de nuestros sistemas o programas, suelen escribirse en ficheros. Aunque más adelante pueden ser examinados, estos archivos por sí solos no nos permitirán detectar si se ha producido un error.

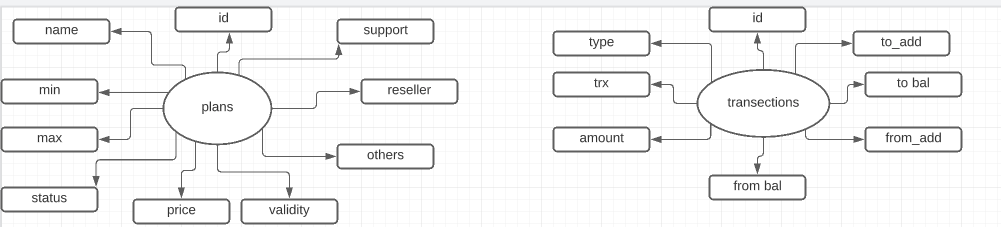
**Users:** El usuario de la base de datos es la identidad del inicio de sesión cuando está conectado a una base de datos. El usuario de la base de datos puede utilizar el mismo nombre que el inicio de sesión, pero no es necesario.

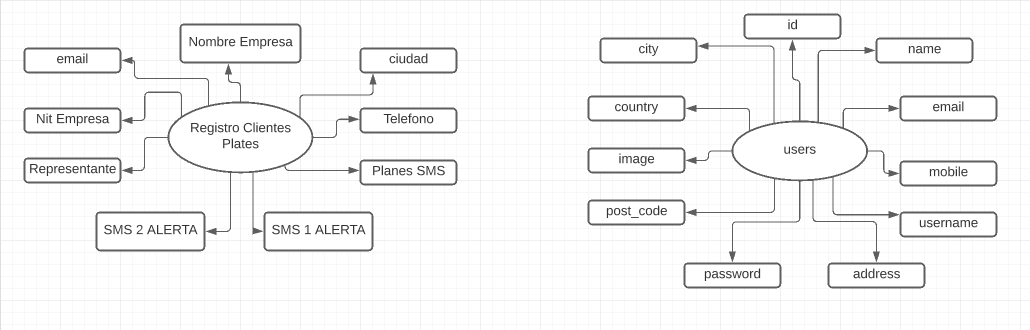
# 4. Responsables e involucrados

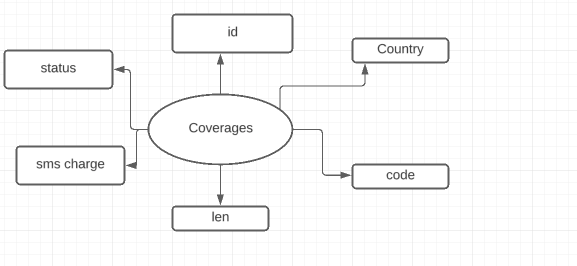
| **Nombre** | **Tipo (Responsable/ Involucrado)** | **Rol** |
| --- | --- | --- |
| *Darwin Tusarma Taba* | *Responsable* | *Desarrollador y cofundador* |
| *Monica Ardila Agudelo* | *Responsable* | *Desarrollador y cofundador* |
| *Santiago Diaz* | *Responsable* | *Desarrollador y cofundador* |
| *Elver Alexis Carreño* | *Responsable* | *Desarrollador Cofundador* |

# 5. Modelo Entidad Relación (MER)

Modelo entidad de relación support messages and support tickets Figura[1]

modelo de plans and transactions figura[2]

Modelo de registro de clients and users figura[3]



Modelo de coverages figura[4]

# 6. Diccionario de Datos.

Aquí podemos encontrar las definiciones de los datos y sus propiedades (atributos , dominios, asociaciones y operaciones). Asimismo identificar claramente el propósito,alcance y campo de aplicación ,sirviendo a los usuarios como orientación tipo informativa.

| **USERS** | | | | | |
| --- | --- | --- | --- | --- | --- |
| **Atributo** | Descripción | Características | Tipo de dato | Dominio de valores | Índice y relaciones |
| **id** | Número de identificación usuario | El campo es obligatorio y no permite valores nulos. | char (20) | Código por rol | Llave principal |
| **Name** | Nombre del usuario | El campo es obligatorio y no permite valores nulos | char (100) | Nombre del plan | Not null |
| **E-mail** | Correo del usuario | El campo es obligatorio y no permite valores nulos | char (100) | Correo en sistema del usuario | Not null |
| **Mobile** | Teléfono del usuario | El campo es obligatorio y no permite valores nulos | char (50) | Numero de telefono del usuario | Null |
| **Username** | nickName | El campo es obligatorio y no permite valores nulos | char(20) | registro en sistema de usuario | Not null |
| **Password** | Contraseña | El campo es obligatorio y no permite valores nulos | Varchar(50) | Contraseña para inicio de sesion de usuario | Not null |
| **Image** | imagen | Campo opcional | blob | Foto usuario | Null |
| **Country** | país | El campo es obligatorio y no permite valores nulos | Char(100) | Ciudad del usuario | Null |
| **City** | ciudad | El campo es obligatorio y no permite valores nulos | char(40) | Ciudad del usuario | Null |
| **Post Code** | Código postal | El campo es obligatorio y no permite valores nulos | char(12) | Codigo postal de ciudad del usuario | Null |
| **address** | Dirección del usuario | Campo opcional | Varchar(200) | Direccion del usuario | Null |
| **Remember token** | Token de la contraseña | El campo es obligatorio y no permite valores nulos | Varchar(20) | Codigo de inicio unico para cada usuario | Null |
| **status** | Estado del usuario | El campo es obligatorio y no permite valores nulos | Tinyint | 1=Activo  2=Inactivo | 1 |

Diccionario de tabla users Tabla[1]

| Registró Clientes | | | | | |
| --- | --- | --- | --- | --- | --- |
| **Atributo** | Descripción | Características | Tipo de dato | Dominio de valores | Índice y relaciones |
| **E-mail** | Correo del  cliente | El campo es obligatorio y no permite valores nulos | Varchar(100) | Correo del  cliente | Not null |
| **Nombre Empresa** | Nombre de  l cliente | El campo es obligatorio y no permite valores nulos | Varchar(50) | Nombre de la empresa del cliente | Not null |
| **NIT** | NIT | El campo es obligatorio y no permite valores nulos | Varchar(20) | Registro tributario de la empresa | Not null |
| **Ciudad** | ciudad | El campo es obligatorio y no permite valores nulos | char(40) | Ubicación de la  empresa | Not null |
| **Teléfono** | Teléfono del  clientel | El campo es obligatorio y no permite valores nulos | char(50) | Teléfono del  cliente | Not null |
| **Representante** | Nombre del representante | El campo es obligatorio y no permite valores nulos | char (100) | Nombre del  representante de la  empresa | Not null |
| **Planes SMS** | Valor del plan | El campo es obligatorio y no permite valores nulos | float | Recurso de  mensajería para  notificación | Not null |
| **SMS 1 ALERTA** | Estado | El campo es obligatorio y no permite valores nulos | Tinyint | 0= NO enviada  1= Enviada | Not null |
| **SMS 2 ALERTA** | Estado | El campo es obligatorio y no permite valores nulos | Tinyint | 0= NO enviada  1= Enviada | Not null |

Diccionario tabla registro clientes tabla [2]

| **Plans** | | | | | |
| --- | --- | --- | --- | --- | --- |
| **Atributo** | Descripción | Características | Tipo de dato | Dominio de valores | Índice y relaciones |
| **ID** | Codigo de identificacion | El campo es obligatorio y no permite valores nulos | chart(15) | Identificación del  plan | Llave principal |
| **Name** | Nombre de plans | El campo es obligatorio y no permite valores nulos | chart(100) | Nombre del plan | Not null |
| **Min** | Cantidad SMS mínimo | El campo es obligatorio y no permite valores nulos | int | Cantidad de min  mínimos. | Not null |
| **Max** | Cantidad SMS  máximo | El campo es obligatorio y no permite valores nulos | int | Cantidad de max  mínimos. | Not null |
| **Price** | Precio del  contrato | El campo es obligatorio y no permite valores nulos | float | Precio del plan | Not null |
| **Validity** | Estado | El campo es obligatorio y no permite valores nulos | Tinyint | 0=activo  1=inactivo | Null |
| **Support** | Soporte | El campo es obligatorio y no permite valores nulos | Varchar | Se define por  satisfacción | Null |
| **Reseller** | Códigos  reservados | El campo es obligatorio y no permite valores nulos | Varchar | Códigos reservados  para planes | Null |
| **Others** | Otras  consideraciones | El campo es obligatorio y no permite valores nulos | Varchar | Observaciones de diferente tipo | Null |
| **Status** | Activo  No activo | El campo es obligatorio y no permite valores nulos | TINYINT | 0=activo  1=inactivo | 1 |

Diccionario de tabla plan tabla [3]

| **COVERAGES** | | | | | |
| --- | --- | --- | --- | --- | --- |
| **Atributo** | Descripción | Características | Tipo de dato | Dominio de valores | Índice y relaciones |
| **Id** | identificación  de coberturas | Primary Key  Autoincrement  Not Null | Primary Key  Autoincrement  Not Null | Se define por frecuencias | Primary Key  Autoincrement  Not Null |
| **Contry** | País en la que  se instala la  cobertura | El campo es obligatorio y no permite valores nulos | Varchar | Identificación  en nombre | Not Null |
| **Code** | Código del  país | El campo es obligatorio y no permite valores nulos | Varchar | Identificación  del código | Not Null |
| **Len** | len | El campo es obligatorio y no permite valores nulos | int | len | Null |
| **SMS\_change** | Intercambio de  MSM | El campo es obligatorio y no permite valores nulos | Varchar | Entrada y  Salida | Null |
| **Status** | Activo inactivo | El campo es obligatorio y no permite valores nulos | Tinyint | 0=activo  1=inactivo | 1 |

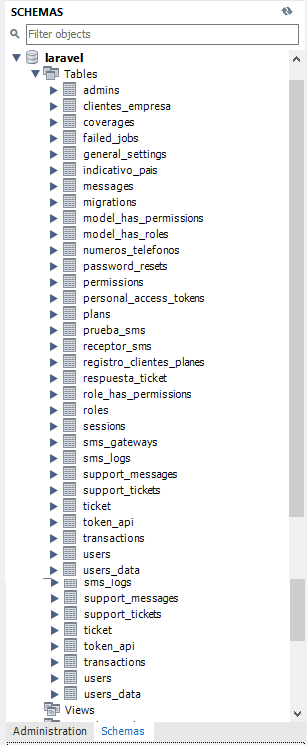
Diccionario tabla coverages tabla[4]

# 

# 

# 

# 7. Modelo Relacional.



*.*Modelo relacional figura[5]

# 8. Justificación Motor Seleccionado

El motor de bases de datos seleccionado es MySQL ya que este es uno de los gestores de base de datos más utilizados por su fácil instalación, administración y bajo costo también resulta sencillo de usar por que es compatible con una amplia lista de sistemas operativos.

Este motor de bases de datos es uno de los mas populares, esto hace que exista una gran cantidad de tutoriales que hace que la curva de aprendizaje sea más corta.

# 9. Requisitos de Configuración

A continuación se describen los requisitos necesarios para desplegar el proyecto en un ambiente de producción teniendo en cuenta los parámetros necesarios para una funcionalidad correcta, segura y estable, actualmente estamos en localhost.

* Servidor tipo linux 18.04 lts vps dedicado.
* PHP V- 8.3
* Mysql V 5.7
* instalar laravel 8
* composer
* Curl php (XAMPP)
* SSL

# 

# 

# 

# 

# 10. Scripts

Estructura de tabla para la tabla `admins`

--

CREATE TABLE `admins` (

`id` int(10) UNSIGNED NOT NULL,

`name` varchar(191) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`email` varchar(191) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`username` varchar(191) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`password` varchar(191) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`remember\_token` varchar(100) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`created\_at` timestamp NULL DEFAULT NULL,

`updated\_at` timestamp NULL DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

- --------------------------------------------------------

--

-- Estructura de tabla para la tabla `coverages`

--

CREATE TABLE `coverages` (

`id` int(10) UNSIGNED NOT NULL,

`country` varchar(191) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`code` varchar(191) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`len` int(11) DEFAULT NULL,

`sms\_charge` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`status` tinyint(4) NOT NULL DEFAULT '1',

`created\_at` timestamp NULL DEFAULT NULL,

`updated\_at` timestamp NULL DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

-- Estructura de tabla para la tabla `general\_settings`

CREATE TABLE `general\_settings` (

`id` int(10) UNSIGNED NOT NULL,

`title` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`base\_color` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`currency\_symbol` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`email\_verification` tinyint(4) DEFAULT NULL,

`sms\_verification` tinyint(4) NOT NULL DEFAULT '0',

`email\_notification` tinyint(4) DEFAULT NULL,

`sms\_notification` tinyint(4) DEFAULT NULL,

`recaptcha` tinyint(4) NOT NULL DEFAULT '1',

`site\_key` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`secret\_key` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`sms\_charge` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`default\_gateway` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`e\_sender` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`e\_message` longtext COLLATE utf8mb4\_unicode\_ci,

`sms\_api` text COLLATE utf8mb4\_unicode\_ci,

`contact\_address` longtext COLLATE utf8mb4\_unicode\_ci,

`contact\_phone` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`contact\_email` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`created\_at` timestamp NULL DEFAULT NULL,

`updated\_at` timestamp NULL DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

-- --------------------------------------------------------

--

-- Estructura de tabla para la tabla `migrations`

--

CREATE TABLE `migrations` (

`id` int(10) UNSIGNED NOT NULL,

`migration` varchar(191) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`batch` int(11) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

-- Estructura de tabla para la tabla `password\_resets`

--

CREATE TABLE `password\_resets` (

`email` varchar(191) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`token` varchar(191) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`status` tinyint(4) DEFAULT '0',

`created\_at` timestamp NULL DEFAULT NULL,

`updated\_at` timestamp NULL DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

-- --------------------------------------------------------

--

-- Estructura de tabla para la tabla `plans`

CREATE TABLE `plans` (

`id` int(10) UNSIGNED NOT NULL,

`name` varchar(191) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`min` varchar(191) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`max` varchar(191) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`price` varchar(191) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`validity` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`support` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`reseller` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`others` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`status` tinyint(4) NOT NULL DEFAULT '1',

`created\_at` timestamp NULL DEFAULT NULL,

`updated\_at` timestamp NULL DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

-- --------------------------------------------------------

--

-- Estructura de tabla para la tabla `Registro\_clientes\_planes`

CREATE TABLE `Registro\_clientes\_planes` (

`Email` varchar(250) COLLATE utf8\_bin NOT NULL,

`Nombre Empresa` text COLLATE utf8\_bin NOT NULL,

`Nit Empresa` varchar(250) COLLATE utf8\_bin NOT NULL,

`Ciudad` varchar(250) COLLATE utf8\_bin NOT NULL,

`Telefono` int(250) NOT NULL,

`Representante` varchar(250) COLLATE utf8\_bin NOT NULL,

`Planes SMS` varchar(250) COLLATE utf8\_bin NOT NULL,

`SMS 1 ALERTA` varchar(250) COLLATE utf8\_bin NOT NULL,

`SMS 2 ALERTA` varchar(250) COLLATE utf8\_bin NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8\_bin;

-- --------------------------------------------------------

--

-- Estructura de tabla para la tabla `sms\_gateways`

CREATE TABLE `sms\_gateways` (

`id` int(10) UNSIGNED NOT NULL,

`name` varchar(191) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`val1` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`val2` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`val3` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`status` tinyint(4) NOT NULL DEFAULT '0',

`created\_at` timestamp NULL DEFAULT NULL,

`updated\_at` timestamp NULL DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

-- --------------------------------------------------------

-- Estructura de tabla para la tabla `support\_messages`

CREATE TABLE `support\_messages` (

`id` int(10) UNSIGNED NOT NULL,

`supportticket\_id` varchar(191) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`type` tinyint(4) NOT NULL,

`message` longtext COLLATE utf8mb4\_unicode\_ci NOT NULL,

`created\_at` timestamp NULL DEFAULT NULL,

`updated\_at` timestamp NULL DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

-- Estructura de tabla para la tabla `support\_tickets`

--

-- --------------------------------------------------------

CREATE TABLE `support\_tickets` (

`id` int(10) UNSIGNED NOT NULL,

`user\_id` varchar(191) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`ticket` varchar(191) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`subject` varchar(191) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`status` tinyint(4) NOT NULL DEFAULT '0',

`created\_at` timestamp NULL DEFAULT NULL,

`updated\_at` timestamp NULL DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

-- --------------------------------------------------------

--

-- Estructura de tabla para la tabla `sms\_logs`

CREATE TABLE `sms\_logs` (

`id` int(10) UNSIGNED NOT NULL,

`user\_id` varchar(191) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`number` varchar(191) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`status` varchar(191) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`created\_at` timestamp NULL DEFAULT NULL,

`updated\_at` timestamp NULL DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

-- --------------------------------------------------------

-- Estructura de tabla para la tabla `transections`

CREATE TABLE `transections` (

`id` int(10) UNSIGNED NOT NULL,

`to\_add` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`to\_bal` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`from\_add` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`from\_bal` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`amount` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`type` tinyint(4) DEFAULT NULL,

`trx` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`created\_at` timestamp NULL DEFAULT NULL,

`updated\_at` timestamp NULL DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

-- --------------------------------------------------------

-- Estructura de tabla para la tabla `users`

CREATE TABLE `users` (

`id` int(10) UNSIGNED NOT NULL,

`name` varchar(191) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`email` varchar(191) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`mobile` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`username` varchar(191) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`password` varchar(191) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`image` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`country` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`city` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`post\_code` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`address` text COLLATE utf8mb4\_unicode\_ci,

`remember\_token` varchar(100) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`status` tinyint(4) NOT NULL DEFAULT '1',

`verification\_time` datetime DEFAULT NULL,

`verification\_code` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`email\_verify` tinyint(4) NOT NULL DEFAULT '1',

`sms\_verify` tinyint(4) NOT NULL DEFAULT '1',

`two\_step\_verify` tinyint(4) NOT NULL DEFAULT '0',

`two\_step\_verification` tinyint(1) NOT NULL DEFAULT '1',

`two\_step\_code` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`balance` varchar(191) COLLATE utf8mb4\_unicode\_ci NOT NULL DEFAULT '0',

`refer\_by` varchar(191) COLLATE utf8mb4\_unicode\_ci NOT NULL DEFAULT '0',

`api\_key` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`sms` varchar(191) COLLATE utf8mb4\_unicode\_ci DEFAULT '0',

`gateway` varchar(191) COLLATE utf8mb4\_unicode\_ci NOT NULL DEFAULT '0',

`roll` tinyint(4) NOT NULL DEFAULT '0',

`created\_at` timestamp NULL DEFAULT NULL,

`updated\_at` timestamp NULL DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

--

-- Índices para tablas volcadas

-- Indices de la tabla `admins`

--

ALTER TABLE `admins`

ADD PRIMARY KEY (`id`),

ADD UNIQUE KEY `admins\_email\_unique` (`email`);

--

-- Indices de la tabla `coverages`

--

ALTER TABLE `coverages`

ADD PRIMARY KEY (`id`);

--

-- Indices de la tabla `general\_settings`

--

ALTER TABLE `general\_settings`

ADD PRIMARY KEY (`id`);

--

-- Indices de la tabla `migrations`

--

ALTER TABLE `migrations`

ADD PRIMARY KEY (`id`);

--

-- Indices de la tabla `password\_resets`

--

ALTER TABLE `password\_resets`

ADD KEY `password\_resets\_email\_index` (`email`);

--

-- Indices de la tabla `plans`

--

ALTER TABLE `plans`

ADD PRIMARY KEY (`id`);

--

-- Indices de la tabla `sms\_gateways`

--

ALTER TABLE `sms\_gateways`

ADD PRIMARY KEY (`id`);

--

-- Indices de la tabla `sms\_logs`

--

ALTER TABLE `sms\_logs`

ADD PRIMARY KEY (`id`);

--

-- Indices de la tabla `support\_messages`

--

ALTER TABLE `support\_messages`

ADD PRIMARY KEY (`id`);

--

-- Indices de la tabla `support\_tickets`

--

ALTER TABLE `support\_tickets`

ADD PRIMARY KEY (`id`);

--

-- Indices de la tabla `transections`

--

ALTER TABLE `transections`

ADD PRIMARY KEY (`id`);

--

-- Indices de la tabla `users`

--

ALTER TABLE `users`

ADD PRIMARY KEY (`id`),

ADD UNIQUE KEY `users\_email\_unique` (`email`);

--

-- AUTO\_INCREMENT de las tablas volcadas

--

--

-- AUTO\_INCREMENT de la tabla `admins`

--

ALTER TABLE `admins`

MODIFY `id` int(10) UNSIGNED NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=2;

--

-- AUTO\_INCREMENT de la tabla `coverages`

--

ALTER TABLE `coverages`

MODIFY `id` int(10) UNSIGNED NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=243;

--

-- AUTO\_INCREMENT de la tabla `general\_settings`

--

ALTER TABLE `general\_settings`

MODIFY `id` int(10) UNSIGNED NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=2;

--

-- AUTO\_INCREMENT de la tabla `migrations`

--

ALTER TABLE `migrations`

MODIFY `id` int(10) UNSIGNED NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=33;

--

-- AUTO\_INCREMENT de la tabla `plans`

--

ALTER TABLE `plans`

MODIFY `id` int(10) UNSIGNED NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=4;

--

-- AUTO\_INCREMENT de la tabla `sms\_gateways`

--

ALTER TABLE `sms\_gateways`

MODIFY `id` int(10) UNSIGNED NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=25;

--

-- AUTO\_INCREMENT de la tabla `sms\_logs`

--

ALTER TABLE `sms\_logs`

MODIFY `id` int(10) UNSIGNED NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=5;

--

-- AUTO\_INCREMENT de la tabla `support\_messages`

--

ALTER TABLE `support\_messages`

MODIFY `id` int(10) UNSIGNED NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=8;

--

-- AUTO\_INCREMENT de la tabla `support\_tickets`

--

ALTER TABLE `support\_tickets`

MODIFY `id` int(10) UNSIGNED NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=6;

--

-- AUTO\_INCREMENT de la tabla `transections`

--

ALTER TABLE `transections`

MODIFY `id` int(10) UNSIGNED NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=30;

--

-- AUTO\_INCREMENT de la tabla `users`

--

ALTER TABLE `users`

MODIFY `id` int(10) UNSIGNED NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=13;

* Logs auditoría

create table log\_auditorial(

id int auto\_increment primary key,

codmodulo char(10),

opcion varchar(100),

operacion char(3),

detalle varchar(250),

fecha datetime,

nombremaquina char(30)

);

create table clientes(

documento varchar(255) primary key,

nombre varchar(255),

operacion char(3)

)

Trigger

si un usuario ingresa a la opción de crear clientes y graba un cliente, en el log de auditoría debería crearse un registro al insertar los datos del cliente en la tabla clientes

DELIMITER $$

create trigger creacion\_auditoria AFTER insert on clientes

for each row

begin

if new.operacion='GRA' then

insert into log\_auditoria(codmodulo,opcion,operacion,detalle,fecha,nombremaquina) values

('factura','crear cliente',new.operacion,

concat('',new.nombre,new.documento),now(),'MaquinaN');

end if;

end $$

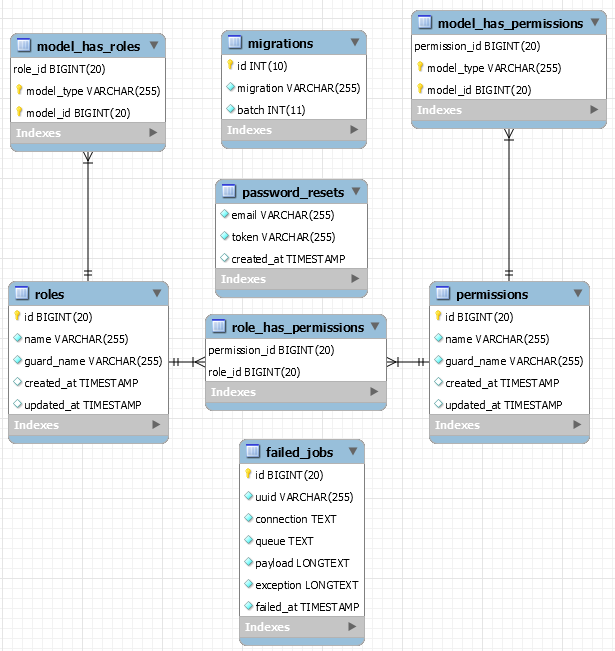
**11.Roles y Permisos**

Nuestra plataforma de turnos SMS contará con un sistema de roles y permisos el cual le va a permitir al administrador y a los clientes crear nuevos roles y asignar permisos de lectura, escritura o edición, así como podrán limitar los accesos visuales de las vistas o secciones existentes en las cuentas del super admin.

Para esta funcionalidad contaremos con un diseño de tablas y relaciones el cual tendrá 5 tablas.

* **permissions**
* **roles**
* **roles\_has\_permissions**
* **users\_has\_permissions**
* **user\_has\_roles**

Se debe de tener en cuenta que vamos a manejar varios guardias o múltiples protectores con la finalidad de que los permisos que se almacenan allí pueda verificarse si el usuario cuenta con su permiso o no.

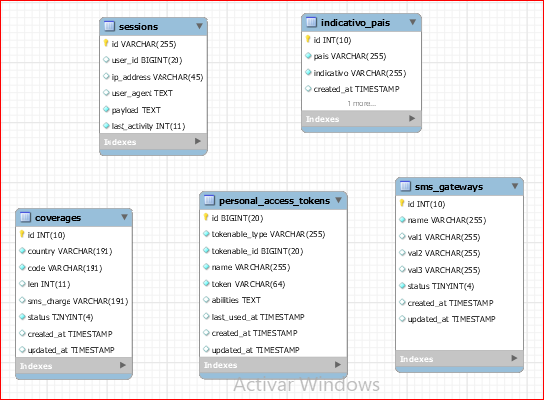


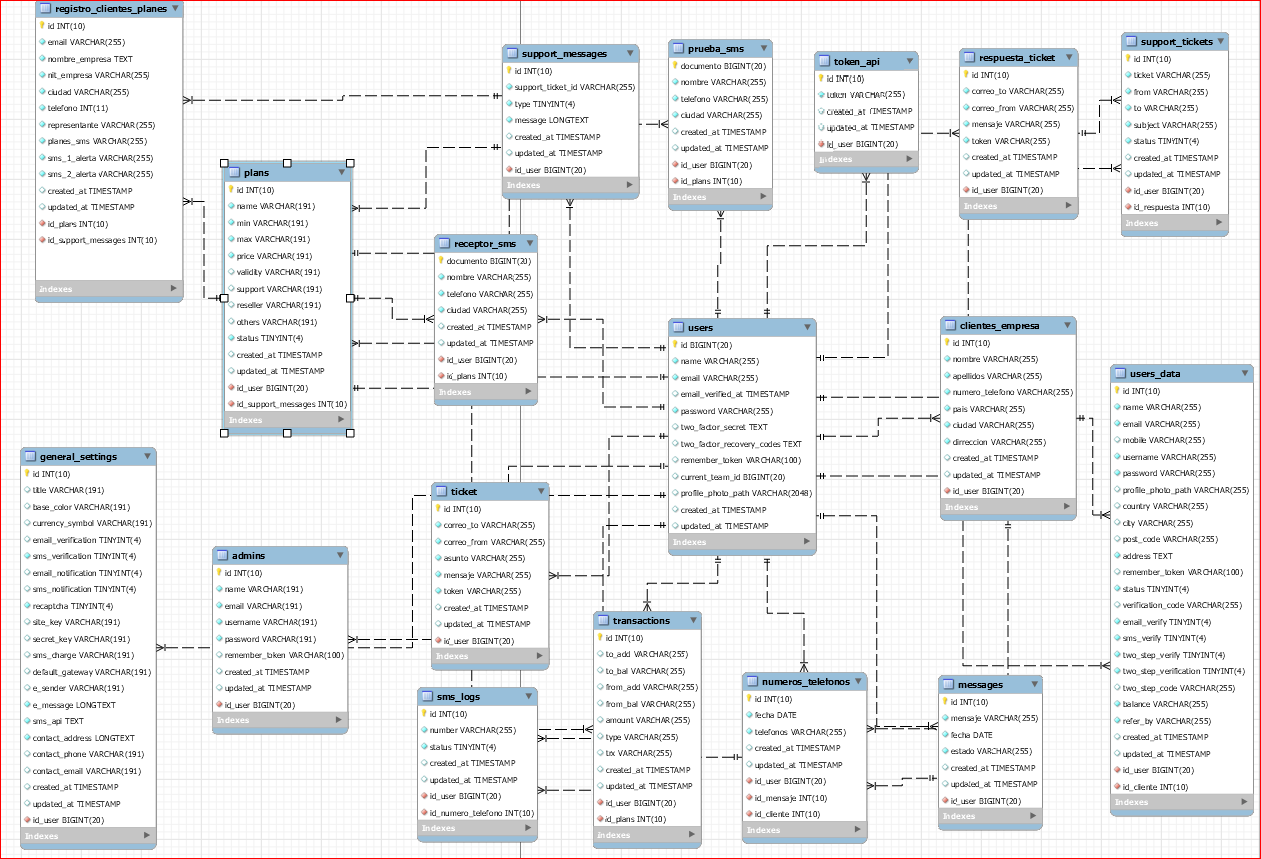
*Relaciones tablas roles y permisos[6]*

Los usuarios se asociaron a con cada rol y permiso en el sistema según corresponda, cada rol está asociado con múltiples permisos, de esta manera podemos asignar un permiso a un rol o varios permisos a un rol.

Los roles iniciales son el super Administrador el cual es el que se encargará de crear los roles de otros usuarios, en el caso de nuestra plataforma cada empresa o cliente tendrá un super admin y a partir de allí estos podrán crear otros roles en la plataforma, es importante aclarar cada empresa está identificada c por un id principal que nos va a permitir que la información de cada cliente sea única pero que a su vez cada cliente gozará de las mismas funcionalidades del sistema bajo los roles que se crean.

Los super administradores tendrán definida una regla global mientras los demás usuarios creados por el super admin solo tendrán acceso a las funciones que el superadmin les defina.





# 12. Configuración y Ejecución de la Base de Datos

La configuración de la base de datos, se realiza usando las herramientas Laravel eloquent usando un rds de aws.

**Laravel incluye Eloquent**: un mapeador relacional de objetos (ORM) que hace que sea agradable interactuar con su base de datos. Cuando se usa Eloquent, cada tabla de la base de datos tiene un "Modelo" correspondiente que se usa para interactuar con esa tabla. Además de recuperar registros de la tabla de la base de datos, los modelos Eloquent le permiten insertar, actualizar y eliminar registros de la tabla también.

Antes de comenzar, asegúrese de configurar una conexión de base de datos en el config/database.php archivo de configuración de su aplicación .

**Radio Data System (RDS)**, acrónimo en inglés de sistema de radiodifusión de datos, es un protocolo de comunicaciones que permite enviar pequeñas cantidades de datos digitales, inaudibles para el radioescucha, con la señal de una emisora de radio FM; parte de dichos datos se ven presentados en una pantalla del aparato …(Amazon RDS) es un servicio administrado que facilita las tareas de configuración, operación y escalado de una base de datos relacional en la nube. es un servicio web que facilita la configuración, el funcionamiento y la escala de una base de datos relacional en la nube AWS. ... Para obtener información acerca de cómo usar Amazon Aurora, consulte la Guía del usuario de Amazon Aurora.

¿**Qué es Amazon Web Services (AWS)?** Amazon Web Services es un proveedor de servicios en la nube, nos permite disponer de almacenamiento, recursos de computación, aplicaciones móviles, bases de datos y un largo etcétera en modalidad de cloud computing.

**Route 53 Configuración de dominios:** Amazon Route 53 es un servicio de sistema de nombres de dominio escalable y de alta disponibilidad. conecta de forma efectiva las solicitudes del usuario con la infraestructura en ejecución en AWS, como instancias de Amazon EC2, equilibradores de carga de Elastic Load Balancing o buckets de Amazon S3. Además, puede utilizarse para dirigir usuarios a infraestructuras externas a AWS.

**load Balanced Balanceadores de Carga:** Herramienta, la cual direcciona a un cliente al servidor web que se encuentre con mayor disponibilidad entre los que cuentan con el mismo contenido.

**Waf Módulo de seguridad:** Protege de múltiples ataques al servidor de aplicaciones web en el backend. La función del WAF es garantizar la seguridad del servidor web mediante el análisis de paquetes de petición HTTP / HTTPS y modelos de tráfico.

**WEBGRAFIA**

<https://laravel.com/docs/8.x/eloquent#introduction>

<https://laravel.com/docs/8.x/database#configuration>

<https://www.google.com/search?q=que+es+aws&rlz=1C1CHBD_esCO907CO907&sxsrf=AOaemvLE7gOOYae39Um-8h5z6h7GZ8PzZA%3A1632354227001&ei=sr9LYeG3PJGVwbkPtqeQqA8&oq=que+es++aws&gs_lcp=Cgxnd3Mtd2l6LXNlcnAQARgAMgQIIxAnMgUIABCABDIFCAAQgAQyBQgAEIAEMgUIABCABDIFCAAQgAQyBggAEAcQHjIFCAAQgAQyBQgAEIAEMgUIABCABDIICAAQgAQQiwMyCAgAEIAEEIsDMgkIABAHEB4QiwMyCQgAEAcQHhCLAzoHCAAQRxCwAzoICAAQCBAHEB5KBAhBGABQ5QhYzRNg-RpoAHADeACAAbIBiAGTBJIBAzAuM5gBAKABAcgBCLgBAcABAQ&sclient=gws-wiz-serp>

<https://www.youtube.com/watch?v=4SBFz4jG7KU>

# 13. Otras Consideraciones

Todos los elementos descritos anteriormente, están definidos uno por uno con el objetivo se seguir la secuencia para la correcta configuración de la base de datos, esta secuencia hace que la ejecución de la base de datos se de manera completa.

Por lo anterior es importante tener en cuenta que la correcta configuración de la base de datos permitirá, que se ejecuten procesos simultáneamente, mediante las integraciones que se plantean a los largo del proyecto como son.

El Dashboard, es una de las herramientas que muestra al usuario cada uno de los recursos disponibles, en donde de manera directa está relacionada con la base de datos.

El consumo del API propia del proyecto API DIGITURNO SMS MADS, en donde la definición de cada unos de sus EndPoints, tiene relación directa con el servicio WEB SERVICE, en donde se ejecutan procesos desde el DASHBOARD, el cliente y el intercambio de información con el uso del WEB SERVICE en tiempo Real.

**Importante**

Dentro de la definición y la implementación que se le dé al uso de las Bases de Datos en este proyecto, el lector debe entender la información que se le da en este documento en líneas anteriores, ya que en las consideraciones se da una vista rápida, de los aspectos más relevantes que se tienen en cuenta al momento de ejecutar el proyecto y la influencia que tiene la base de datos, como elemento integrador de todo el WEB SERVICE.