



**Manual Técnico API Tiqueteria Móvil  
Coonorte  
2019**



LMSOFTWARE S.A.S.  
RESPALDO PERMANENTE  
NIT 900.011.659-5

## Vistazo General

La APIs Rest TMovil (Representational State Transfer -Transferencia de Estado Representacional) para la elaboración de Tiquetes y Libros de Viaje desde un dispositivo móvil de la empresa Coonorte. Es una API basada en HTTP que aplicaciones externas pueden usar para el manejo de la caja de un usuario (abrir, cerrar, cuadrar), asimismo le permite a este mismo usuario abrir y cerrar libros de viaje para la agencia asociada al dispositivo móvil que utiliza la API, igualmente consultar los diferentes libros de viaje para una ruta y horario determinado. Además de lo anteriormente mencionado permite vender, anular y devolver tiquetes.

## API Host

Para realizar peticiones se tienen dos urls , una para el ambiente de pruebas y otra para el ambiente de producción.

Url Pruebas

<https://host:63780/>

url Producción

<https://host:52520/>

## Códigos de estado de respuesta HTTP

La API de TMovil de Coonorte utiliza los [Códigos de estado de respuesta HTTP](#) más utilizados.

Se puede considerar todas las respuestas que no sean devueltas con un código HTTP 200 como un error.

## Formato de respuestas de la API

Siempre que se realice una petición correcta a la API se obtendrá una respuesta en formato JSON con los atributos de datos especificados para cada método y un objeto respuesta que indicara si el proceso fue exitoso ó no.

```
{  
  "tiquetes": [],  
  "respuesta": {  
    "codigorespuesta": "CodigoMetodo-00",  
    "mensaje": "Mensaje que especifica el código obtenido "  
  }  
}
```

## Como autenticarse en la API

Para usar los métodos de la API, se debe obtener un token de acceso válido. Una vez se obtenga un token válido, se deberá adjuntar en las cabeceras HTTP con cada petición que se realice a la API y servirá para autenticarse en la API TMovil de Coonorte. Este método proporciona un acceso seguro a la API y permite asociar las llamadas a la API con su cuenta de usuario.

Para obtener un nuevo token, usa el método de [autenticación](#).

Si la autenticación es válida, se obtendrá un token valido que expirará al día siguiente. Cuando el token expire, se deberá pedir un nuevo token para seguir realizando peticiones a la API. Utilice el atributo “*expiration*” que le informa en cuantos tiempo expirará el token.



LMSOFTWARE S.A.S.  
RESPALDO PERMANENTE  
NIT 900.011.659-5

```
{  
  "token":  
    "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiJtZW5hbmdd1bG9AZ21haWwuY29tliwianRpljoiNDdmNTYxNDktNmRiZi00MDAwLTkzMmQMTThjOTJiNjA4ZDU2liwiZXhwIjoxNTY5NTM1ODM5LCJpc3MiOiJsbnh2NhbGhvc3QiLCJhdWQiOiJ1c2VycyJ9.IOudFb6a87nSHvtG3hQWpPWzUQxdaotH7yBjltwbT5s",  
  "expiration": "2019-09-26T22:10:39Z"  
}
```

Para realizar peticiones a los métodos de la API, se debe adjuntar en las cabeceras de su petición una cabecera *Authorization* añadiendo el tipo de token obtenido y el valor del token

Authorization: Bearer

eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiJtZW5hbmdd1bG9AZ21haWwuY29tliwianRpljoiNDdmNTYxNDktNmRiZi00MDAwLTkzMmQMTThjOTJiNjA4ZDU2liwiZXhwIjoxNTY5NTM1ODM5LCJpc3MiOiJsbnh2NhbGhvc3QiLCJhdWQiOiJ1c2VycyJ9.IOudFb6a87nSHvtG3hQWpPWzUQxdaotH7yBjltwbT5s

Realizar una petición sin la cabecera de *Authorization* o con un token inválido o caducado generará el siguiente error:

```
{  
  "error": {  
    "code": "UNAUTHORIZED",  
    "http_code": 401,  
    "message": "Login required. You must provide a valid access token."  
  }  
}
```

Todos los métodos esperan la entrada de datos en formato Json y las respuestas también serán dadas en este mismo formato.



LMSOFTWARE S.A.S.  
RESPALDO PERMANENTE  
NIT 900.011.659-5

A continuación, se describen los diferentes controladores (clases y métodos) de integración que por medio de esta librería ejecutará las diferentes peticiones.

Para la autenticación

1. [CrearToken](#)

Para la Caja.

1. [AbrirCaja](#)
2. [CerrarCaja](#)
3. [IngresarDocumento](#)
4. [CuadreCaja](#)

Para los Libros de Viaje, los libros de viaje solo se pueden abrir y cerrar para la agencia asociada al dispositivo móvil.

1. [AbrirLibroViaje](#)
2. [CerrarLibroViaje](#)
3. [ConsultaLibrosViaje](#)
4. [ImprimirLibroViaje](#)

Para Tiquetes

1. [IngresarVenta](#)
2. [DevolverTiquete](#)
3. [AnularTiquete](#)
4. [ReimprimirTiquete](#)
5. [ActualizarReimpresiones](#)
6. [GetCliente](#)

Para realizar la carga de los catálogos de rutas, servicios ruta y motivos de anulación/devolución de tiquetes, se tienen los siguientes métodos:

1. [GetRutasByAgencia](#)
2. [GetRutasServiciosByAgencia](#)
3. [GetTipoDevolucion](#)

## Métodos Contenidos en la API TMovil Coonorte

### Método CrearToken

Permite genera un token válido para acceder todos los métodos de la api de tiqueteria móvil.

*Url: host:puerto/Login/CreateToken*

Parámetros de entrada :

NombreParametro	TipoDato	Explicación
Username	string(15)	Id Usuario válido en TransNet que maneje caja
Password	String(30)	Clave entregada a el usuario
Serial	String(50)	Cadena que identifica el dispositivo móvil

Respuesta Recibida

token	string
Expiration	Datetime
CodigoRespuesta	String(5) con formato XX-NN donde XX es el id del método NN es el número del mensaje El Código 00 es una respuesta exitosa
Mensaje	String(200)

Una vez se obtiene un token válido, se podrán realizar peticiones a la API.

Los tipos de mensaje que se devuelven son:

CodigoRespuesta	Mensaje
VP-00	Datos Validos
VP-01	Datos Inválidos

## Método AbrirCaja

Permite iniciar la caja por usuario y día. Solo se puede abrir una caja siempre y cuando se haya cerrado la del ultimo día en que se trabajo con dicha caja.

*Url: host:puerto/api/Cajas/AbrirCaja*

Parámetros de entrada :

NombreParametro	TipoDato	Explicación
SVM	Seguridad	Contiene los campos de Seguridad
FechaApertura	String(10)	Fecha En la que se desea abrir la caja. Formato AAAA-MM-DD

Respuesta Recibida.

idCaja	string(20)	Numero que identifica la caja asignada al usuario
nombreCaja	string(50)	Nombre dado en TramNet a la caja asociada al equipo Movil que invoca el método
CodigoRespuesta	String(5)	Con formato XX-NN donde XX es el id del método, en este caso CA NN es el numero del mensaje El Código 00 ex una respuesta exitosa
Mensaje	String(200)	

Los tipos de mensaje que se devuelven son:

CodigoRespuesta	Mensaje
AC-00	Caja abierta para el día 2019-09-30 de manera exitosa
AC-00	La caja ya esta abierta para la fecha solicitada
AC-00	La caja ya esta cerrada para la fecha solicitada, NO se puede



**LMSOFTWARE S.A.S.**  
**RESPALDO PERMANENTE**  
**NIT 900.011.659-5**

	volver a abrir
AC-01	La fecha para la que se solicita abrir la caja no puede ser diferente a la fecha actual
AC-02	No se puede abrir una caja para una fecha menor a la fecha actual
AC-03	Debe cerrar la caja de la fecha 2019-09-29 para poder continuar



## Método CerrarCaja

Permite cerrar en cualquier momento la caja asociada a un usuario en un día específico.

*Url: host:puerto/api/Cajas/CerrarCaja*

Parámetros de entrada :

NombreParametro	TipoDato	Explicación
SVM	Seguridad	Contiene los campos de Seguridad
FechaApertura	String(10)	Fecha En la que se abrió la caja. Formato AAAA-MM-DD

Respuesta Recibida.

logEstadoCaja	String (“Abierta”, ”Cerrada”)
DtmFechaCierre	DateTime
Respuesta	Objeto Respuesta

Los tipos de mensaje que se devuelven son:

CodigoRespuesta	Mensaje
CC-00	Proceso de cierre realizado exitosamente, recuerde no podrá volver a abrirse nuevamente
CC-00	El proceso de cierre se realizo, pero los movimientos para este caja no están cuadrados por favor revise
CC-00	La caja ya se encuentra cerrada
CC-01	El cierre de la caja no se pudo realizar
CC-02	No existe una caja para la fecha solicitada
CC-03	La fecha de cierre de la caja no corresponde con la fecha de hoy

## Método IngresarDocumento

Este método permite ingresar el valor de la consignación diaria de la caja asociada al dispositivo móvil.

*Url: host:puerto/api/Cajas/IngresarDocumento*

Parametros de entrada :

NombreParametro	TipoDato	Explicación
SVM	Seguridad	Contiene los campos de Seguridad
ValorAConsignar	double	Valor de la consignación
Observaciones	String(100)	Observaciones acerca de la consignación efectuada

Respuesta Recibida.

Respuesta	Objeto Respuesta
-----------	------------------

Los tipos de mensaje que se devuelven son:

CodigoRespuesta	Mensaje
IC-00	El Ingreso de la consignación termino de manera Exitosa
IC-99	Hubo algún error en el proceso

## Método CuadreCaja

Este método permitirá conocer en cualquier momento los movimientos realizados de una caja asociada con un usuario en un día específico.

*Url: host:puerto/api/Cajas/CuadreCaja*

Parametros de entrada :

NombreParametro	TipoDato	Explicación
SVM	Seguridad	Contiene los campos de Seguridad
FechaApertura	String(10)	Fecha En la que se abrió la caja. Formato AAAA-MM-DD

Respuesta Recibida.

Estado Caja	String (“Abierta”, ”Cerrada”)
List<Detalles>	Lista de objetos <a href="#">DetalleCaja</a>
Respuesta	Objeto Respuesta

Los tipos de mensaje que se devuelven son:

CodigoRespuesta	Mensaje
CU-00	Cuadre de caja listado con éxito.
CU-01	No existe una caja para la fecha solicitada

## Método AbrirLibroViaje

Este método permitirá abrir un libro de viaje para la agencia asignada al dispositivo móvil que hace uso de la API.

*Url: host:puerto/api/LibrosViaje/AbrirLibroViaje*

Parametros de entrada :

NombreParametro	TipoDato	Explicación
SVM	Seguridad	Contiene los campos de Seguridad
FechaSalida	String(10)	Fecha de salida del libro del viaje Formato AAAA-MM-DD
HoraSalida	string(8)	Hora de salida del Libro de Viaje Formato hh:mm:ss o hh:mm
Movil	string(6)	Identificador del móvil al cual esta asignado el libro de viaje
IdRuta	integer	Id de la ruta asignada al libro de viaje
IdServicio	string(2)	Servicio asignado a el libro de viaje

Respuesta Recibida.

NumeroNuevoLibro	Integer – identificador del libro de viaje creado
List<Detalles>	Lista de objetos <a href="#">DetalleCaja</a>
Respuesta	Objeto Respuesta

Los tipos de mensaje que se devuelven son:

CodigoRespuesta	Mensaje
AL-00	Operación Exitosa



**LMSOFTWARE S.A.S.**  
**RESPALDO PERMANENTE**  
**NIT 900.011.659-5**

AL-01	La fecha de salida del libro no puede ser menor a la fecha Actual.
AL-02	La hora de salida del libro no puede ser menor a la hora Actual
AL-03	El móvil xxx al cual desea asociar el libro no existe.
AL-04	La Ruta xx para la que desea abrir el libro, no existe para la agencia a la que pertenece el dispositivo móvil.
AL-05	No existe ningún servicio asociado a la ruta informada en el libro que se desea crear.
AL-99	Mensaje de error de una excepción ocurrida durante el proceso de apertura del libro de viaje

## Método CerrarLibroViaje

Este método permitirá cerrar un libro de viaje para la agencia asignada al dispositivo móvil que hace uso de la API.

*Url: host:puerto/api/LibrosViaje/AbrirLibroViaje*

Parametros de entrada :

NombreParametro	TipoDato	Explicación
SVM	Seguridad	Contiene los campos de Seguridad
IdLibro	integer	Identificador del Libro que se va a cerrar
Movil	string(6)	Identificador del móvil al cual esta asignado el libro de viaje

Respuesta Recibida.

Respuesta	Objeto Respuesta
-----------	------------------

Los tipos de mensaje que se devuelven son:

CodigoRespuesta	Mensaje
CL-00	El libro xxx para la agencia xx, a la cual pertenece el dispositivo móvil ya se encuentra cerrado.
CL-00	El libro fue cerrado de manera exitosa.
CL-01	El libro xxx que desea cerrar no existe para la agencia xx a la cual esta asociada el dispositivo móvil
CL-02	No se puede cerrar un libro con un código de anticipado en el numero del móvil



LMSOFTWARE S.A.S.  
RESPALDO PERMANENTE  
NIT 900.011.659-5

CL-03	NO SE PUEDE cerrar el libro, la fecha es de un libro Anticipado
CL-04	No se puede cerrar un libro que no tiene asignado conductor
CL-99	Mensaje de error de una excepción ocurrida durante el proceso de apertura del libro de viaje

## Método ConsultarLibrosViaje

Este método permite consultar los libros de viaje para una fecha determinada y para un rango de horas

Url: *host:puerto/api/LibrosViaje/ConsultaLibrosViaje*

Parametros de entrada :

NombreParametro	TipoDato	Explicación
SVM	Seguridad	Contiene los campos de Seguridad
AgenciaVenta	integer	Agencia para la cual se quieren buscar los libros de viaje
FechaSalidaVehiculo	string(10)	Fecha de salida del libro Formato AAAA-MM-DD
HoraInicial	String(8)	Hora inicial para la consulta formato hh:mm:ss, se puede entrar solo hh:mm
HoraFinal	String(8)	Hora final para la consulta formato hh:mm:ss, se puede entrar solo hh:mm
Movil	String	Numero del móvil a consultar. Opcional

Respuesta Recibida.

LibrosdeViaje	Lista de objetos del tipo <a href="#">LibroViaje</a>
Respuesta	Objeto <a href="#">Respuesta</a>

Los tipos de mensaje que se devuelven son:

CodigoRespuesta	Mensaje
CL-00	Consulta Realizada con Éxito
CL-01	No existe una caja abierta para la fecha solicitada
CL-02	No hay libros disponibles para la información suministrada



## Método ImprimirLibroViaje

Permite imprimir la información correspondiente a un libro de viaje en particular.

Url: *host:puerto/api/LibrosViaje/ImprimirLibroViaje*

Parametros de entrada :

NombreParametro	TipoDato	Explicación
SVM	Seguridad	Contiene los campos de Seguridad
AgenciaLibro	Integer	Agencia a la que pertenece el libro a imprimir
IdLibro	Integer	Nro del libro de viaje que se desea imprimir

Respuesta Recibida.

FechaSalida	DateTime
HoraSalida	string
Movil	string
Placa	string
Conductor	string
EstadoLibro	string
Origen	string
Destino	string
FechaCierre	DateTime
ValorTotalLibro	double
ValorTotalDevoluciones	double
Destinos	List< <a href="#">Destino</a> >

Pasajeros	List< <a href="#">DetallePasajero</a> >
Respuesta	Objeto tipo <a href="#">Respuesta</a>

Los tipos de mensaje que se devuelven son:

CodigoRespuesta	Mensaje
IL-00	Datos para imprimir el libro traídos exitosamente
IL-01	La agencia a la cual pertenece el libro no existe
IL-02	El libro a imprimir no existe para la agencia informada
IL-03	El libro debe estar cerrado para poderse imprimir

## Método IngresarVenta

*Url: host:puerto/api/Tiquetes/IngresarVentaTiquete*

Este método permite ingresar la venta de tiquetes en un libro y agencia determinada.

Parametros de entrada :

NombreParametro	TipoDato	Explicación
SVM	Seguridad	Contiene los campos de Seguridad
LibroViaje	int	Libro para el cual se venderán los pasajes
AgenciaVenta	int	Agencia para la cual se venden los pasajes
FechaDespacho	String(8)	Fecha de salida del libro Formato AAAA-MM-DD
Pasajeros	Lista<Pasajero>	Lista de objetos <a href="#">Pasajero</a>

Respuesta Recibida.

Tiquetes	Lista<Tiquete>
Respuesta	Objeto <a href="#">Respuesta</a>

Los tipos de mensaje que se devuelven son:

CodigoRespuesta	Mensaje
VT-00	El Ingreso de la venta termino de manera Exitosa
VT-01	La fecha de realización de la venta no puede diferente a la fecha actual
VT-02	No hay una caja abierta para la fecha solicitada
VT-03	El numero de Libro no existe
VT-04	El numero de puestos que se pretende vender excede el numero de puestos del móvil
VT-05	Alguno de los puestos asignados ya están vendidos



**LMSOFTWARE S.A.S.**  
**RESPALDO PERMANENTE**  
**NIT 900.011.659-5**

VT-06	Alguno de los puestos asignados es mayor al numero de puestos del móvil
VT-07	La ruta es un dato requerido
VT-08	No existe una ruta con el servicio reportado
VT-09	El puesto #n ya fue utilizado por otro usuario
VT-10	El puesto #n esta siendo utilizado por una transacción web No PUEDE VENDERLO, NI RESERVARLO
VT-11	La ruta asignada al pasajero xxx No existe para la agencia xx
VT-12	Movimiento descuadrado Verifique
VT-99	Cuando ocurre alguna excepción , contendrá el mensaje de error enviado por sql

## Método DevolverTiquete

*Url: host:puerto/api/Tiquetes/DevolverTiquete*

Este método permite devolver parte o la totalidad del dinero de un tiquete. Se pueden hacer devoluciones en libro abierto o cerrado. Las devoluciones aparecen en el libro de viaje al volver a imprimir despumes de realizarlas.

Parametros de entrada :

NombreParametro	TipoDato	Explicación
SVM	<a href="#">Seguridad</a>	Contiene los campos de Seguridad
AgenciaVenta	Integer	Agencia a la que pertenece el tiquete
LibroVenta	Integer	Nro del libro de venta asociado al tiquete
NroTiquete	Integer	Nro de identificación del tiquete que se va a devolver
ValorADevolver	Double	Valor que se devuelve puede ser una parte o todo el valor
CodigoMotivoDevolucion	string	Un código de la tabla de motivos de devolución

Respuesta Recibida:

Respuesta	Objeto tipo <a href="#">Respuesta</a>
-----------	---------------------------------------

Los tipos de mensaje que se devuelven son:

CodigoRespuesta	Mensaje
VT-00	La devolución del tiquete se realizo e manera exitosa.
DT-01	La caja no esta abierta
DT-02	El libro no existe para esa agencia



LMSOFTWARE S.A.S.  
RESPALDO PERMANENTE  
NIT 900.011.659-5

DT-03	El número de tiquete no existe para el libro y la agencia reportados
DT-04	El tiquete ya se encuentra anulado
DT-05	El tiquete ya fue devuelto
DT-06	La fecha de devolución es mayor a 30 días
DT-07	El Valor a devolver no puede ser mayor al valor del tiquete y devoluciones anteriores
DT-99	Cuando ocurre alguna excepción , contendrá el mensaje de error enviado por sql

## Método AnularTiquete

*Url: host:puerto/api/Tiquetes/AnularTiquete*

Este método se utiliza para anular la venta total de un tiquete. Las anulaciones solo son posibles en libro abierto. Las anulaciones no aparecen en el libro de viaje.

Parametros de entrada :

NombreParametro	TipoDato	Explicación
SVM	Seguridad	Contiene los campos de Seguridad
AgenciaVenta	Integer	Agencia a la que pertenece el tiquete
LibroVenta	Integer	Nro del libro de venta asociado al tiquete
NroTiquete	Integer	Nro de identificación del tiquete que se va a devolver
CodigoMotivoAnulacion	string	Un código de la tabla de motivos de devolución

Respuesta Recibida.

Respuesta	Objeto tipo <a href="#">Respuesta</a>
-----------	---------------------------------------

Los tipos de mensaje que se devuelven son:

CodigoRespuesta	Mensaje
AT-00	El tiquete fue anulado de manera exitosa
AT-01	La caja no esta abierta
AT-02	El libro no existe para esa agencia
AT-03	Las anulaciones NO SON posibles sobre un libro cerrado, por favor haga una devolución



**LMSOFTWARE S.A.S.**  
**RESPALDO PERMANENTE**  
**NIT 900.011.659-5**

AT-04	El número de tiquete no existe para el libro y la agencia reportados
AT-05	El tiquete ya se encuentra anulado
AT-06	El tiquete fue devuelto anteriormente
AT-99	Cuando ocurre alguna excepción , contendrá el mensaje de error enviado por sql



## Método ReimprimirTiquete

*Url: host:puerto/api/Tiquetes/ReimprimirTiquetesTiquete*

Este método se utiliza para la reimpresión de un tiquete.

Parametros de entrada :

NombreParametro	TipoDato	Explicación
SVM	Seguridad	Contiene los campos de Seguridad
FechadeViaje	string(10)	Fecha en que se realizará el viaje. Formato yyyy.MM-dd. Este parámetro es de caracter obligatorio
IdCliente	string(13)	Numero de identificación del cliente que desea viajar. Este parámetro es de caracter opcional
HoraSalida	string(8)	Hora de salida. Formato hh:mm:ss se permite hh:mm. Este parámetro es de caracter opcional
RutaViaje	integer	Identificador de la ruta del viaje. Este parámetro es de caracter opcional
NumeroTiquete	integer	Numero del tiquete que se desea reimprimir. Este parámetro es de caracter opcional

Respuesta Recibida.

Tiquetes	Lista de objetos del tipo CamposITiquete
Respuesta	Objeto tipo <a href="#">Respuesta</a>



**LMSOFTWARE S.A.S.**  
**RESPALDO PERMANENTE**  
**NIT 900.011.659-5**

Los tipos de mensaje que se devuelven son:

CodigoRespuesta	Mensaje
RT-00	La información para la reimpresión fue obtenida de manera exitosa
RT-99	Cuando ocurre alguna excepción , contendrá el mensaje de error enviado por sql

## Método ActualizarReimpresiones

*Url: host:puerto/api/Tiquetes/ActualizarReimpresiones*

Este método se utiliza para actualizar la cantidad de veces que un tiquete ha sido reimpresso. Esto con el fin de poder realizar el control de la regla de negocio que dice que solo se puede reimprimir un tiquete un máximo de 3 veces.

Parametros de entrada :

NombreParametro	TipoDato	Explicación
SVM	Seguridad	Contiene los campos del objeto Seguridad
AgenciaDespacho	integer	Id de la agencia del libro de viaje
IdLibro	integer	Numero de identificación del libro de viaje
NumTiquete	integer	Numero asignado al tiquete dentro del libro de viaje
NumReimpresiones	integer	Num anterior de veces que ha sido impreso el tiquete. Este parameto se corresponde con el ultimo parámetro recibido al consumir el método ReimprimirTiquete.

Respuesta Recibida.

Tiquetes	Lista de objetos del tipo CamposITiquete
Respuesta	Objeto tipo <a href="#">Respuesta</a>

Los tipos de mensaje que se devuelven son:

CodigoRespuesta	Mensaje
AR-00	Actualización de Reimpresiones Exitosa



**LMSOFTWARE S.A.S.**  
**RESPALDO PERMANENTE**  
**NIT 900.011.659-5**

AR-99	Cuando ocurre alguna excepción , contendrá el mensaje de error enviado por sql
-------	--

## Método GetCliente

Url: *host:puerto/api/Cientes/GetCliente*

Este método se utiliza para obtener la información de un cliente para la venta de tiquetes.

Parametros de entrada :

Nombre del Parámetro	TipoDato	Explicación
IdCliente	string(13)	Numero de identificación del cliente.

Respuesta Recibida.

Nombre del Parámetro	Tipo Dato
Nombres	String(60)
apellidoUno	String(30)
apellidoDos	String(30)
telefono	string(15)
Respuesta	Objeto tipo <a href="#">Respuesta</a>

Los tipos de mensaje que se devuelven son:

CodigoRespuesta	Mensaje
GC-00	Datos del cliente xxxx traídos con éxito
GC-01	El cliente xxxx No existe en la tabla de clientes
GC-99	Cuando ocurre alguna excepción , contendrá el mensaje de error enviado por sql

## Método GetRutasByAgencia

*Url: host:puerto/api/Rutas/GetRutasByAgencia*

Este método permite obtener el catalogo de las Rutas existentes para una agencia.

Recibe como parámetro de entrada el token de autenticación en la cabecera de la petición Https y el identificador de la agencia para la cual se solicitan las rutas

Nombre Parámetro	Tipo de dato	Explicación
IdAgencia	Integer	Identificador de la agencia para al cual se desean consultar las rutas

La respuesta es una lista de rutas con los siguientes campos:

Nombre Parámetro	Tipo de dato	Explicación
IdAgencia	Integer	Identificador de agencia
IdRuta	Integer	Identificador de ruta
NombreRuta	String	Nombre de la ruta

un ejemplo de un ítem de salida en formato Json:

```
{
  "idAgencia":2,
  "idRuta":2100,
  "nombreRuta": "MARINILLA <- BOGOTA"
}
```

## Método GetRutasServicioByAgencia

Url: *host:puerto/api/RutasServicios/GetRutasServicioByAgencia*

Este método permite obtener el catalogo de las Rutas existentes para una agencia determinada.

Recibe como parámetro de entrada el token de autenticación en la cabecera de la petición Https y el identificador de la agencia para la cual se solicitan los servicios de las rutas de esta agencia

Nombre Parametro	Tipo de dato	Explicación
IdAgencia	Integer	Identificador de la agencia para al cual se desean consultar las rutas

La respuesta es una lista de rutas con los siguientes campos:

Nombre Parametro	Tipo de dato	Explicación
IdAgencia	Integer	Identificador de agencia
IdRuta	Integer	Identificador de ruta
IdServicio	String	Identificador del Servicio
NombreServicio	String	Nombre del Servicio
Tarifa	Double	Valor del Servicio
TarifaTemporada	Double	Valor del servicio en temporada



**LMSOFTWARE S.A.S.**  
**RESPALDO PERMANENTE**  
**NIT 900.011.659-5**

un ejemplo de un ítem de salida en formato Json:

```
{  
  "idAgencia": 2,  
  "idRuta": 100,  
  "idServicio": "2",  
  "tarifa": 65000.0,  
  "tarifaTemporada":0.0  
}
```



## Método GetTipoDevolucion

*Url: host:puerto/api/Tipos/GetTipoDevolucion*

Este método permite obtener el catalogo de los motivos de devolución/anulación

Recibe como parámetro de entrada el token de autenticación en la cabecera de la petición Https

La respuesta es una lista con el código y la descripción de los tipos de anulación/devolución:

Nombre Parametro	Tipo de dato
Código	String(10)
Descripción	String(60)

Un ejemplo de un elemento de la lista de salida en formato Json

```
[  
  {  
    "código": "1",  
    "descripcion": "1.Cambio destino"  
  },  
]
```

## Tipos de datos

### SVM

Nombre campo	Tipo de Dato	Explicación
Username	String	Id del usuario de la tiquetera Móvil
Password	String	Clave del usuario
Serial	String	Nro de Serial de la tiquetera Móvil

### Respuesta

Nombre campo	Tipo de Dato	Explicación
CodigoRespuesta	String(5) con formato XX-NN donde XX es el id del método, en este caso AC NN es el numero del mensaje El Código 00 ex una respuesta exitosa	Es el código asociado a un evento dentro de cada método
Mensaje	String(200)	Detalle del Evento

### DetalleCaja

Nombre campo	Tipo de Dato	Explicación
TipoDocumento	String(2)	Tipo de documento que origino el movimiento en la caja(LV → Libro de viaje, TQ → Tiquete
IdDocumento	Integer	Numero del documento afectado
DetalleGeneral	String	Descripción de el movimiento
ValorDebito	double	Suma de valores que aumentan caja
ValorCredito	double	Suma de valores que disminuyen caja
TipoMovimiento	string(1)	D → Débito C → Crédito
Cantidad	double	Cantidad

ValorUnidad	double	Valor unitario
-------------	--------	----------------

## LibroViaje

Nombre campo	Tipo de Dato	Explicación
strMovil	String	# del móvil asociado al Libro
intIdRuta	integer	Nro de la ruta asociada al Libro
StrDescripcionRuta	string	Descripción de la Ruta
DtmFechaSalida	datetime	Fecha de salida del vehículo (yyyy-MM-dd)
StrHoraSalida	string	Hora de salida del vehículo (hh:mm)
IntIdLibro	integer	Numero del Libro de Viaje
StrTipoLibro	string	Tipo de Libro (GE)
IntRutaPrincipal	integer	Id de la ruta principal
IntPuestosDisponibles	integer	Numero de puestos disponibles del libro
FltTarifaRuta	double	Tarifa asociada a la ruta y el servicio
IntPuestos	integer	Numero de puestos del móvil
StrIdServicio	String	Numero asociado al el servicio y ruta
PuestosLibres	list<puestos>	Lista del objeto Puesto
estadoLibro	String(3)	Identifica el estado del libro 'ABI','CER'

## Puesto

Nombre campo	TipoDato	Explicación
IntIdPuesto	integer	Numero del puesto
Estado Puesto	booleano	0 Libre

## Tiquete

Nombre campo	TipoDato	Explicación
NumeroTiquete	integer	Numero del tiquete asignado en la Venta

Puesto	integer	Numero del puesto para ese tiquete
--------	---------	------------------------------------

## Pasajero

NombreCampo	TipoDato	Explicación
intIdPuesto	integer	Puesto que se pretende vender
strIdPasajero	integer	Id del pasajero (CC)
strNombres	string	Nombres del pasajero
strApellidoUno	string	Primer Apellido Pasajero
strApellidoDos	string	Segundo Apellido Pasajero
intIdRutaVendida	integer	Ruta que se vende al pasajero
intIdServicioVendido	integer	Servicio asociado a la ruta

## DetallePasajero

Nombre campo	TipoDato	Explicación
Puesto	integer	Numero del puesto del pasajero
NroTiquete	integer	Numero del pasaje
DestinoPasajero	string	Lugar donde termina su viaje el pasajero
NombrePasajero	string	Nombres y apellidos del pasajero
Tarifa	double	Valor del tiquete en esa ruta para el servicio seleccionado
IdPasajero	string	Numero de identificación del pasajero, normalmente cedula de ciudadanía

## Destino

Nombre campo	TipoDato	Explicación
CiudadDestino	string	Ciudad destino Libro
CantidadTiquetes	integer	Cantidad de tiquetes vendidos para ese destino
Tarifa	double	Valor del destino
ValorTotalDestino	double	Cantidad multiplicado por tarifa

## CamposITiquete

Nombre campo	TipoDato	Explicación
Bus	string	Numero del móvil asignado a el tiquete
Valor	double	Valor del tiquete
FechaSalidaLibro	string	Fecha de salida del libro de viaje
Hora	string	Hora de salida del libro de viaje
NumPuesto	integer	Numero de puesto asignado al pasajero
IdAgenciaLibro	integer	Agencia a la que pertenece el libro
Origen	string	Nombre de el lugar donde se origina el viaje
Destino	string	Nombre de el lugar donde termina el viaje
IdPasajero	string	Numero de identificación del pasajero
NombrePasajero	string	Nombre y apellidos del pasajero
IdTiquete	integer	Numero de tiquete que se esta reimprimiendo
IdLibro	integer	Identificación del libro del viaje para el cual se vendió el tiquete
CajaVende	string	Caja que vendió el tiquete
FechaVenta	string	Fecha en que fue vendido el tiquete
AgenciaVenta	integer	Id de la agencia que vendió el tiquete
NombreAgenciaVenta	string	Nombre de la agencia que vendió el tiquete



**LMSOFTWARE S.A.S.**  
**RESPALDO PERMANENTE**  
**NIT 900.011.659-5**

Observaciones	string	Observaciones si las tiene del tiquete
CodigoBarras	string	Código de barras identificador del tiquete
NumImpresiones	integer	Cantidad de veces que ha sido impreso el tiquete.



LMSOFTWARE S.A.S.  
RESPALDO PERMANENTE  
NIT 900.011.659-5

## Consideraciones de la Instalación

La API esta desarrollada en Visual Studio 2019 , en lenguaje C# en framework .NET Core 2.2 y Base de datos Microsoft SQL Server 2014.

La API se encuentra instalada en un servidor de aplicaciones IIS(Internet Information Services). Como consideración en el despliegue de la API instaló en el en el servidor de aplicaciones el runtime: .NET Core 2.2 Runtime & Hosting Bundle for Windows (v2.2.7)

El cual se descarga de la pagina de descargas de Microsoft <https://dotnet.microsoft.com/download>