|  |  |
| --- | --- |
| Geo Parking | Web Service  2015 |

El documento describe información de la API de Google Maps que servirá como capacitación en el desarrollo del producto Geo Parking.

# Control de la documentación

### Control de la Configuración.

|  |  |
| --- | --- |
| Título: | Web Service |
| Referencia: | GeoP\_Documentacion\_WebService.docx |
| Autores: | Lucas Toneatto |
| Fecha: | 01/052/2014 |

### Histórico de Versiones.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Versión | Fecha | Estado | Responsable | Cambios |
| 1.0\_DraftA | 04/02/2015 | Pendiente de Revisión | Lucas Toneatto[autor] |  |
| 1.0\_DraftB | 06/02/2015 | Revisado, pendiente de corrección | Ignacio Frigerio [Revisor] | Comentarios sobre algunos cambios |
| 1.0\_DraftC | 06/02/2015 | Revisado, pendiente de corrección | Lucas Toneatto[autor] | Quedan dos puntos pendientes a quedar de acuerdo: |
| 1.0\_DraftD | 10/02/2015 | Pendiente de Revisión | Lucas Toneatto [autor] |  |

Contenido

[Control de la documentación 2](#_Toc411342520)

[Control de la Configuración. 2](#_Toc411342521)

[Histórico de Versiones. 2](#_Toc411342522)

[Introducción 4](#_Toc411342523)

[Descripción Web Services 5](#_Toc411342524)

[PlayasController (Controlador de Playas) 6](#_Toc411342525)

[GetPlayas (ciudad) 6](#_Toc411342526)

[Get (idPlaya) 6](#_Toc411342527)

[GetUbicacionesPlayas (ciudad, idTipoVehículo) 6](#_Toc411342528)

[GetUbicacionesPlayasPorDistancia (latitud, longitud, idTipoVehículo) 6](#_Toc411342529)

[PostActualizarNombreEmailPlaya 7](#_Toc411342530)

[PostActualizarTipoPlaya 7](#_Toc411342531)

[PostActualizarHorarioPlaya 7](#_Toc411342532)

[DisponibilidadController (Controlador de Disponibilidades) 9](#_Toc411342533)

[GetDisponibilidadPlayaPorTipoVehiculo (idPlaya, idTipoVehiculo) 9](#_Toc411342534)

[GetDisponibilidadPlayasPorTipoVehiculo (idPlayas, idTipoVehiculo) 9](#_Toc411342535)

[PostActualizarDisponibilidad 9](#_Toc411342536)

[PostActualizarDisponibilidadGeneral 9](#_Toc411342537)

[ServiciosController (Controlador de Servicios) 11](#_Toc411342538)

[PostReistrarServicio ( 11](#_Toc411342539)

[PostActualizarCapacidadServicio ( 11](#_Toc411342540)

[PostCancelarServicio ( 11](#_Toc411342541)

[PreciosController (Controlador de Precios) 13](#_Toc411342542)

[PostRegistrarPrecio ( 13](#_Toc411342543)

[PostActualizarPrecio ( 13](#_Toc411342544)

[GetPreciosPlayas (idTipoVehículo, idPlayas) 13](#_Toc411342545)

[EstadisticasController (Controlador de Estadisticas) 15](#_Toc411342546)

[PostGuardarConsulta ( 15](#_Toc411342547)

[ContactoController (Controlador de Contacto) 16](#_Toc411342548)

[PostEnviarEmailDeContacto (( 16](#_Toc411342549)

# Introducción

EL presente documento detalla la estructura y las funcionalidades que brinda el Web Service de GeoParking, y los cuales son consumidos por la Aplicación Móvil y por los sistemas de las playas de estacionamiento que estén conectadas a GeoParking.

Un servicio web (en inglés, Web Service o Web Services) es una tecnología que utiliza un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones. Distintas aplicaciones de software desarrolladas en lenguajes de programación diferentes, y ejecutadas sobre cualquier plataforma, pueden utilizar los servicios web para intercambiar datos en redes de ordenadores como Internet. La interoperabilidad se consigue mediante la adopción de estándares abiertos.

Dentro de las ventajas de proporcionadas por los Web Services podemos mencionar:

* Aportan interoperabilidad entre aplicaciones de software independientemente de sus propiedades o de las plataformas sobre las que se instalen.
* Los servicios Web fomentan los estándares y protocolos basados en texto, que hacen más fácil acceder a su contenido y entender su funcionamiento.
* Permiten que servicios y software de diferentes compañías ubicadas en diferentes lugares geográficos puedan ser combinados fácilmente para proveer servicios integrados.

# Descripción Web Services

El modelo de servicios web implementado por GeoParking es un modelo API REST. Una API REST es una API, o librería de funciones, a la que se accede por el protocolo HTTP. Una REST API, por tanto, se accede a través de direcciones web o URLs en las que enviamos los datos de nuestra consulta. Como respuesta a la consulta sobre el REST API se obtienen datos en diferentes formatos, como pueden ser texto plano, XML, JSON, etc.

La aplicación que brinda los servicios está estructurada en controladores que son quienes reciben las peticiones tanto de la Aplicación Web como de los sistemas de las playas de estacionamiento.

A continuación se detallarán los servicios ofrecidos por cada uno de los controladores.

## PlayasController (Controlador de Playas)

Este controlador ofrece servicios de consulta y modificación de datos directamente relacionados con las playas de estacionamiento que forman parte del sistema GeoParking.

GetPlayas (ciudad)**:**Retorna todas las playas que pertenezcan a una determinada ciudad.

URL: geoparking.com.ar/api/Playas/GetPlayas?ciudad=**<ciudad>**

PARAMETROS:

* *<ciudad> (string)*: cadena de texto de la ciudad buscada.

RESPUESTA: listado de playas pertenecientes a la ciudad en formato JSON.

Get(idPlaya)**:**Retorna toda la información relacionada a una playa.

URL: geoparking.com.ar/api/Playas/**<idPlaya>**

PARAMETROS:

* *<idPlaya> (int)*: número de identificación único de la playa.

RESPUESTA: información de una playa de estacionamiento en formato JSON.

GetUbicacionesPlayas (ciudad, idTipoVehículo)**:** Retorna las direcciones de todas las playas de estacionamiento ubicadas en una ciudad y que brinden servicio de estacionamiento para un tipo de vehículo en particular.

URL: geoparking.com.ar /api/Playas/GetUbicacionesPlayas?

ciudad=**<ciudad>**&tipoVehiculoId=**<idTipoVehiculo>**

PARAMETROS:

* *<ciudad> (string)*: cadena de texto de la ciudad buscada.
* *<idTipoVehiculo> (int)*: número de identificación de tipo de vehículo.

RESPUESTA: direcciones de playas de estacionamiento en formato JSON.

GetUbicacionesPlayasPorDistancia (latitud,longitud, idTipoVehículo)**:**Retornalas direcciones de todas las playas que se encuentren a menos de 10km de distancia de una ubicación establecida y que tengan lugares de estacionamiento para un tipo de vehículo en particular.

URL: geoparking.com.ar/api/Playas/GetUbicacionesPlayasPorDistancia?

latitud=**<latitud>**&longitud=**<longitud>**&tipoVehiculo=**<idTipoVehiculo>**

PARAMETROS:

* *<latitud>(string)*: coordenada latitudinal de la dirección.
* *<longitud>(string)*: coordenada longitudinal de la dirección.
* *<idTipoVehiculo> (int)*: número de identificación de tipo de vehículo.

RESPUESTA: direcciones de playas de estacionamientos en formato JSON.

PostActualizarNombreEmailPlaya([*FromBody*]**PlayaControlador**playa)**:** Actualiza el nombre y el mail de la playa de estacionamiento.

URL: geoparking.com.ar/api/Playas/PostActualizarNombreEmailPlaya

PARAMETROS:

* *<****PlayaControlador****>(Objeto)*: objeto del controlador que debe tener seteado los siguientes parámetros:
  + IdPlaya*(int)*: número de identificación único de la playa
  + Nombre*(string)*: cadena de texto con el nombre de la playa.
  + Mail*(string)*: cadena de texto con el mail de la playa.

RESPUESTA: “True” si la acción se realizó con éxito,”False” para cualquier inconveniente que haya impedido la actualización.

PostActualizarTipoPlaya([*FromBody*]**PlayaControlador**playa)**:** Actualiza el tipo de la playa.

URL: geoparking.com.ar/api/Playas/PostActualizarTipoPlaya

PARAMETROS:

* *<****PlayaControlador****>(Objeto)*: objeto del controlador que debe tener seteado los siguientes parámetros:
  + IdPlaya*(int)*: número de identificación único de la playa.
  + TipoPlayaId*(int)*: número de identificación de tipo de vehículo.

RESPUESTA: “True” si la acción se realizó con éxito,”False” para cualquier inconveniente que haya impedido la actualización.

PostActualizarHorarioPlaya([*FromBody*]**PlayaControlador**playa)**:** Actualiza el horario de la playa.

URL: geoparking.com.ar/api/Playas/PostActualizarHorarioPlaya

PARAMETROS:

* *<****PlayaControlador****>(Objeto)*: objeto del controlador que debe tener seteado los siguientes parámetros:
  + IdPlaya*(int)*: número de identificación único de la playa.
  + DiaAtencionId*(int)*: número de identificación del día de atención.
  + HoraDesde*(string)*: hora de apertura.
  + HoraHasta*(string)*: hora de cierre.

RESPUESTA: “True” si la acción se realizó con éxito,”False” para cualquier inconveniente que haya impedido la actualización.

**Atributos PlayaControlador**

* **int** IdPlaya {get; set; }
* **public** string Nombre {get; set; }
* **public** string Mail {get; set; }
* **public** int TipoPlayaId {get; set; }
* **public** int DiaAtencionId {get; set; }
* **public** string HoraDesde {get; set; }
* **public** string HoraHasta {get; set; }

## DisponibilidadController (Controlador de Disponibilidades)

Este controlador ofrece servicios de consulta y modificación de datos directamente relacionados con la disponibilidad de cada tipo de vehículos de las playas de estacionamiento que forman parte del sistema GeoParking.

GetDisponibilidadPlayaPorTipoVehiculo (idPlaya, idTipoVehiculo)**:** Retorna la disponibilidad de un tipo de vehiculo para una playa en particular.

URL: geoparking.com.ar/api/Disponibilidad/GetDisponibilidadPlayaPorTipoVehiuclo?

idPlaya=**<idPlaya>**&idTipoVehiculo=**<idTipoVehiculo>**

PARAMETROS:

* *<idPlaya> (int)*: número de identificación de la playa a consultar.
* *<idTipoVehiculo> (int)*: número de identificación del tipo de vehiculo.

RESPUESTA: disponibilidad para el tipo de vehiculo de esa playa de estacionamiento.

GetDisponibilidadPlayasPorTipoVehiculo (idPlayas, idTipoVehiculo)**:** Retorna la disponibilidad de un tipo de vehiculo para una lista de playas.

URL: geoparking.com.ar/api/Disponibilidad/GetDisponibilidadPlayasPorTipoVehiuclo?

idPlayas=**<idPlayas>**&idTipoVehiculo=**<idTipoVehiculo>**

PARAMETROS:

* *<idPlayas> (string(“1,2,3,…,n”))*: listado de números de identificación de las playas a consultar.
* *<idTipoVehiculo> (int)*: número de identificación del tipo de vehiculo.

RESPUESTA: disponibilidad para el tipo de vehiculo de esa playa de estacionamiento.

PostActualizarDisponibilidad([*FromBody*]**DisponibilidadControlador**disponibilidad)**:** Actualiza la disponibilidad de un tipo de vehículo para una playa en particular de acuerdo a la ocurrencia de un evento, ya sea un ingreso o egreso de vehículo.

URL: geoparking.com.ar/api/Disponibilidad/PostActualizarDisponibilidad

PARAMETROS:

* *<***DisponibilidadControlador***>(Objeto)*: objeto del controlador que debe tener seteado los siguientes parámetros:
  + IdPlaya*(int)*: número de identificación único de la playa.
  + IdTipoVehiculo*(int)*: número de identificación del tipo de vehículo.
  + IdEvento*(int)*: número de identificación del evento.
  + Fecha*(string):*fecha de la actualización.

RESPUESTA: “True” si la acción se realizó con éxito,”False” para cualquier inconveniente que haya impedido la actualización.

PostActualizarDisponibilidadGeneral([*FromBody*]**DisponibilidadControlador**disponibilidad)**:** Actualiza la disponibilidad de un tipo de vehiculo para una playa en particular de acuerdo a la disponibilidad actual de la playa de estacionamiento.

URL: geoparking.com.ar/api/Disponibilidad/PostActualizarDisponibilidadGeneral

PARAMETROS:

* *<***DisponibilidadControlador***>(Objeto)*: objeto del controlador que debe tener seteado los siguientes parámetros:
  + IdPlaya*(int)*: número de identificación único de la playa.
  + IdTipoVehiculo*(int)*: número de identificación del tipo de vehículo.
  + Disponibilidad*(int)*: lugares disponibles.
  + IdEvento*(int)*: número de identificación único del evento.
  + Fecha*(string):*fecha de la actualización.

RESPUESTA: “True” si la acción se realizó con éxito,”False” para cualquier inconveniente que haya impedido la actualización.

**AtributosDisponibilidadControlador**

* **int** IdPLaya { get; set; }
* **int** IdTipoVehiculo { get; set; }
* **int** Disponibilidad { get; set; }
* **int** IdEvento { get; set; }
* **string** Fecha { get; set; }

## ServiciosController (Controlador de Servicios)

Este controlador ofrece servicios de consulta y modificación de datos directamente relacionados con los servicios de las playas de estacionamiento que forman parte del sistema GeoParking.

PostReistrarServicio ([*FromBody*]ServicioControladorservicio)**:** Registra un nuevo servicio para una playa de estacionamiento.

URL: geoparking.com.ar/api/Servicios/PostRegistrarServicio

PARAMETROS:

* *<***ServicioControlador***>(Objeto)*: objeto del controlador que debe tener seteado los siguientes parámetros:
  + IdPlaya*(int)*: número de identificación único de la playa.
  + IdTipoVehiculo*(int)*: número de identificación del tipo de vehículo.
  + Capacidad*(int)*: cantidad máxima de lugares disponibles.
  + Precios*(TipoPrecio [] )*: array de precios.

RESPUESTA: “True” si la acción se realizó con éxito,”False” para cualquier inconveniente que haya impedido la actualización.

PostActualizarCapacidadServicio ([*FromBody*]ServicioControladorservicio)**:**Actualiza la capacidad de un servicio para una playa de estacionamiento.

URL: geoparking.com.ar/api/Servicios/PostActualzarCapacidadServicio

PARAMETROS:

* *<***ServicioControlador***>(Objeto)*: objeto del controlador que debe tener seteado los siguientes parámetros:
  + IdPlaya*(int)*: número de identificación único de la playa.
  + IdTipoVehiculo*(int)*: número de identificación del tipo de vehículo.
  + Capacidad*(int)*: cantidad máxima de lugares disponibles.

RESPUESTA: “True” si la acción se realizó con éxito,”False” para cualquier inconveniente que haya impedido la actualización.

PostCancelarServicio ([*FromBody*]ServicioControladorservicio)**:** Da de baja un servicio de una playa de estacionamiento.

URL: geoparking.com.ar/api/Servicios/PostRegistrarServicio

PARAMETROS:

* *<***ServicioControlador***>(Objeto)*: objeto del controlador que debe tener seteado los siguientes parámetros:
  + IdPlaya*(int)*: número de identificación único de la playa.
  + IdTipoVehiculo*(int)*: número de identificación del tipo de vehículo.

RESPUESTA: “True” si la acción se realizó con éxito,”False” para cualquier inconveniente que haya impedido la actualización.

**AtributosServicioControlador**

* **int** IdPLaya { get; set; }
* **int** IdTipoVehiculo { get; set; }
* **int**Capacidad { get; set; }
* **TipoPrecios []** Precios { get; set; }

**Atributos TipoPrecio**

* **int** IdTiempo { get; set; }
* **Double** Monto { get; set; }

## PreciosController (Controlador de Precios)

Este controlador ofrece servicios de consulta y modificación de datos directamente relacionados con los precios de las playas de estacionamiento que forman parte del sistema GeoParking.

PostRegistrarPrecio([*FromBody*]PrecioControladorprecio)**:**Registra un nuevo precio, para el estacionamiento de un tipo de vehiculo, por un determinado tiempo.

URL: geoparking.com.ar/api/Precios/PostRegistrarPrecio

PARAMETROS:

* *<***PrecioControlador***>(Objeto)*: objeto del controlador que debe tener seteado los siguientes parámetros:
  + IdPlaya*(int)*: número de identificación único de la playa.
  + IdTiempo*(int)*: número de identificación del tiempo.
  + IdTipoVehiculo*(int)*: número de identificación del tipo de vehículo.
  + Precio*(Double)*: precio.

RESPUESTA: “True” si la acción se realizó con éxito,”False” para cualquier inconveniente que haya impedido la actualización.

PostActualizarPrecio([*FromBody*]PrecioControladorprecio)**:**Actualiza un precio, para el estacionamiento de un tipo de vehiculo, por un determinado tiempo.

URL: geoparking.com.ar/api/Precios/PostActualzarPrecio

PARAMETROS:

* *<***PrecioControlador***>(Objeto)*: objeto del controlador que debe tener seteado los siguientes parámetros:
  + IdPlaya *(int)*: número de identificación único de la playa.
  + IdTiempo *(int)*: número de identificación del tiempo.
  + IdTipoVehiculo *(int)*: número de identificación del tipo de vehículo.
  + Precio *(Double)*: precio.

RESPUESTA: “True” si la acción se realizó con éxito,”False” para cualquier inconveniente que haya impedido la actualización.

GetPreciosPlayas (idTipoVehículo, idPlayas)**:** Retorna los precios de estacionamiento del tipo vehículo indicado, de las playas a las cuales les perecen los identificadores enviados en la petición.

URL: geoparking.com.ar/api/Playas/GetPreciosPlayas?

tipoVehiculo=**<idTipoVehiculo>**&idPlayas=**<idPlayas>**

PARAMETROS:

* *<idTipoVehiculo> (int)*: número de identificación de tipo de vehículo.
* <idPlayas>*(int[])*: listado de identificadores de playas de estacionamiento.

RESPUESTA: precios por hora de las playas de estacionamiento solicitadas y del tipo de vehículo indicado, en formato JSON.

**Atributos PrecioControlador**

* **int** IdPLaya { get; set; }
* **int** IdTiempo { get; set; }
* **int** IdTipoVehiculo { get; set; }
* **double** Precio { get; set; }

## EstadisticasController (Controlador de Estadisticas)

Este controlador ofrece servicios de consulta y modificación de datos directamente relacionados con las estadisticas de las playas de estacionamiento que forman parte del sistema GeoParking.

PostGuardarConsulta([*FromBody*]PostConsultaEstadisticaconsulta)**:**Registra una consulta de la aplicación móvil para posteriores consultas estadísticas del sistema.

URL: geoparking.com.ar/api/Estadisticas/PostGuardarConsulta

PARAMETROS:

* *<***PrecioControlador***>(Objeto)*: objeto del controlador que debe tener seteado los siguientes parámetros:
  + IdPlaya*(int)*: número de identificación único de la playa.
  + IdTipoVehiculo*(int)*: número de identificación del tipo de vehículo.
  + Latitud*(string)*: coordenada latitudinal.
  + Longitud*(string)*:coordenada longitudinal.

RESPUESTA:no retorna nada.

**Atributos PostConsultaEstadistica**

* **int** IdPlaya { get; set; }
* **int** IdTipoVehiculo { get; set; }
* **string** Latitud { get; set; }
* **string** Longitud { get; set; }

## ContactoController (Controlador de Contacto)

Este controlador ofrece servicios envió de mensajes a través del dominio de la aplicación.

PostEnviarEmailDeContacto (([*FromBody*]Contacto mensaje)**:** Envía un mensaje vía mail.

URL: geoparking.com/api/Contacto/PostEnviarEmailDeContacto

PARAMETROS:

* *<***Contacto***>(Objeto)*: objeto del controlador que debe tener seteado los siguientes parámetros:
  + Nombre*(string)*: nombre del autor.
  + Apellido*(string)*: apellido del autor.
  + Telefono*(string)*: teléfono del autor.
  + Email*(string)*: email del autor.
  + Mensaje*(string)*: cadena de texto del mensaje.

RESPUESTA: no retorna nada.

**Atributos Contacto**

* **string** Nombre { get; set; }
* **string** Apellido { get; set; }
* **string** Telefono { get; set; }
* **string** Email { get; set; }
* **string** Mensaje { get; set; }