

Winegard App API Protocol Definition

Status:

Working Draft

Internal Review

Released

Revision to Released Document

Version: 1.2**Date:** 25 July 2017

© 2017 Winegard Company
Confidential

Abstract

This document details the protocol definition of the Winegard REST API for use with the OTA tower and satellite locator mobile app.

This is a controlled document.

Revision History

Date	Revision	Description	Author
06/29/2017	1.0	Initial document	Stacie Venghaus
07/05/2017	1.1	Added common URL for test server Added .php to URL commands	Stacie Venghaus
07/25/2017	1.2	Added commands Towers by Address - Full and Towers by Lat/Lon- Full to return both towers and shows data Fixed missing band from Towers by Address command Added Satellites – Full command to return all satellites Added satellite provider logo to Providers command	Stacie Venghaus

Table of Contents

REFERENCES	5
1 SCOPE.....	6
1.1 SCOPE.....	6
1.2 DEFINITIONS AND ACRONYMS	6
2 OVERVIEW	7
2.1 BASIC INTERFACE FUNCTIONALITY	7
2.2 INTERFACE COMMANDS	7
3 URL COMMANDS	8
3.1 APPLICATION INFO	8
3.2 CUSTOMER REGISTRATION.....	9
3.3 OTA TOWER INFO.....	10
3.4 SATELLITE INFO.....	13



Interface Control Document

Table of Tables

Table 1. Table of References.....5

REFERENCES

Document	Location
MindFire Statement of Work Version 2.0	
17-WC-0002-Wireframes-NM-V1-42817	
FCC Tower Files	https://www.fcc.gov/media/radio/cdbs-database-public-files

Table 1. Table of References

1 SCOPE

1.1 Scope

This document details the REST API designed to provide communication between a Winegard Company server and a third-party app. This communication protocol is designed to meet the requirements for data exchange outlined in the Statement of Work and Wireframes for the MindFire OTA tower and satellite locator mobile app. The intended audience for this document is Winegard software developers, MindFire software developers, and test engineers.

The scope of this document is limited to the subset of HTTP messages needed to support the OTA tower and satellite locator mobile app. The sections in this document detail the commands required to populate data and to retrieve status, tower, and satellite information. Information such as programming languages, database structure, and message processing by the Winegard server is beyond the scope of this document.

1.2 Definitions and Acronyms

OTA	Over-the-air
HTTP	Hypertext Transfer Protocol
JSON	JavaScript Object Notation

2 OVERVIEW

2.1 Basic Interface Functionality

The API described in this document uses an HTTP interface between the Winegard server and the third-party application. The third-party application opens an HTTP connection to the Winegard server to submit and retrieve information from the server.

This API is intended to be backwards compatible. Additional data may be added to the end of the data response in future revisions, but the URL and existing response values will remain the same.

Commands use the HTTP GET or POST operations with necessary resources or parameters. Commands are listed in 2.2 with details provided in section 3. The common URL for the test server is

www.winegard.com/apptest

Resources or parameters, defined in section 3, may be added to the URL to specify the data to be retrieved or saved. The Winegard server will interpret the command to retrieve or save the appropriate data. JSON format is used to return the command status and data. If the command is successful, status is set to true, and the retrieved data is listed. If the command is unsuccessful, status is set to false, and the data section lists the error code such as "Invalid Input", "Query Failed", or "Invalid Command".

An example of a successful command in JSON is

```
{"status":true,"data":{"support_link":"http://www.winegard.com/consumer-contacts"}}
```

An example of an unsuccessful command in JSON is

```
{"status":false,"data":"Invalid Input"}
```

2.2 Interface Commands

The overall commands that the Winegard App API supports are listed below. Section 3 provides details and examples for each command.

Command	Description	Controller	Use Cases
Application Info	Application Information	/appinfo.php	Versions, Revisions
Customers	Customer information	/customers.php	Register email address
Towers	OTA Tower Information	/towers.php	Callsign, Location, Channels, Shows, etc.
Satellites	Satellite Information	/satellites.php	Longitude, Sat Providers, Pointing angles

3 URL COMMANDS

3.1 Application Info

Summary:

This command retrieves the mobile application version number, revisions, terms and conditions, and support information depending on the chosen mode and resource parameters.

Application Usage:

Depending on the provided mode, this GET command provides the latest version number of the application on the chosen operating system, the revisions since the given version and OS, the terms and conditions per OS, the version of the terms and conditions per OS, or the support link. This command returns the command status and either the command failure code if a failure occurs or the requested data such as the application version.

HTTP Verb:

GET

Command Structure:

Command	Response	URL	Parameters
Version	Returns latest version number for given OS	/appinfo.php	mode=ver&os=<android ios>
Revisions	Returns list of changes since given version for OS	/appinfo.php	mode=rev&ver=<version>&os=<android ios>
Terms and Conditions	Returns terms and conditions for given OS	/appinfo.php	mode=terms&os=<android ios>
Terms and Conditions Version	Returns version of latest terms and conditions for given OS	/appinfo.php	mode=termsver&os=<android ios>
Support Link	Returns the support link url	/appinfo.php	mode=support

Examples:

Command	Response
http://<server>/appinfo.php?mode=ver&os=android	{"status":true,"data":{"version":"01.01"}}
http://<server>/appinfo.php?mode=rev&ver=01.00&os=ios	{"status":true,"data":[{"revision":"iOS – another beta version to be tested"}]}
http://<server>/appinfo.php?mode=terms&os=ios	{"status":true,"data":{"terms_conditions":"<h1>Privacy Policy</h1>"}}
http://<server>/appinfo.php?mode=termsver&os=ios	{"status":true,"data":{"terms_version":"1.0"}}
http://<server>/appinfo.php?mode=support	{"status":true,"data":{"support_link":"http://www.winegard.com/consumer-contacts"}}

3.2 Customer Registration

Summary:

This command registers the posted customer email address.

Application Usage:

This POST command accepts the customer email address and stores it in the Winegard database. This command returns the command status and either the command failure code if a failure occurs or the email address to serve as a verification.

HTTP Verb:

POST

Command Structure:

Command	Response	URL	Parameters
Registration	Returns the email address	/customers.php	e=<email address>

Example:

Command	Response
http://<server>/customers.php?e=help@winegard.com	{"status":true,"data":"help@winegard.com"}

3.3 OTA Tower Info

Summary:

This command retrieves the OTA tower information.

Application Usage:

This GET command provides the OTA tower information such as tower detail with shows or the list of towers via address or latitude/longitude within the given range. Maximum advisable range is 70 miles. If it is not provided, the range is defaulted to 70 miles. Latitude and longitude are to be in +/- decimal degrees. Tower detail lists network shows or the network name if no shows are found. Any towers missing the network name within the FCC database results in n/a. This command returns the command status and either the command failure code if a failure occurs or the requested data such as the list of towers in range.

HTTP Verb:

GET

Command Structure:

Command	Response	URL	Parameters
Towers by Address	Returns tower IDs, tower data, distance	/towers.php	address=<address>&range=<miles>
Tower by Address - Full	Returns tower IDs, tower data, distance, shows, show image URL	/towers.php	address=<address>&range=<miles>&details=full
Towers by Lat/Lon	Returns tower IDs, tower data distance	/towers.php	lat=<latitude>&lon=<longitude>&range=<miles>
Towers by Lat/Lon - Full	Returns tower IDs, tower data, distance, shows, show image URL	/towers.php	lat=<latitude>&lon=<longitude>&range=<miles>&details=full
Tower Detail	Returns towers shows, show image URL	/towers.php	id=<tower id>

Examples:

Command	Response
http://<server>/towers.php?address=3000%20kirkwood%20street,%20burlington,%20ia&range=50	{ "return":true,"response":[{"id":"4327","city":"BURLINGTON","channel":"41","vchannel":"26","band":"UHF","callsign":"KGCW-TV","network":"CW","url":"www.winegard.com/select/img/lg-network-cw.png","latitude":"41.135555","longitude":"-90.808334","distance":"27.8"}, {"id":"39020","city":"MACOMB","channel":"21","vchannel":"22","band":"UHF","callsign":"WMEC","network":"PBS","url":"www.winegard.com/select/img/lg-network-pbs.png","latitude":"40.398335","longitude":"-90.731941","distance":"36.1"}, {"id":"28373","city":"OTTUMWA","channel":"15","vchannel":"15","band":"UHF","callsign":"KYOU-TV","network":"FOX","url":"www.winegard.com/select/img/lg-network-fox.png","latitude":"41.195000","longitude":"-91.954170","distance":"49.8"}]}
http://<server>/towers.php?address=3000%20kirkwood%20street,%20burlington,%20ia&range=40&details=full	{ "return":true,"response":[{"id":"4327","city":"BURLINGTON","channel":"41","vchannel":"26","band":"UHF","callsign":"KGCW-TV","network":"CW","url":"www.winegard.com/select/img/lg-network-cw.png","latitude":"41.135555","longitude":"-90.808334","distance":"27.8"}, {"id":"39020","city":"MACOMB","channel":"21","vchannel":"22","band":"UHF","callsign":"WMEC","network":"PBS","url":"www.winegard.com/select/img/lg-network-pbs.png","latitude":"40.398335","longitude":"-90.731941","distance":"36.1","shows":[{"title":"Sesame Street","url":"www.winegard.com/select/img/lg-show-sesame-street.jpg"}, {"title":"Downton Abbey","url":"www.winegard.com/select/img/lg-show-downton-abbey.jpg"}, {"title":"America's Test Kitchen","url":"www.winegard.com/select/img/lg-show-americas-test-kitchen.jpg"}, {"title":"Charlie Rose","url":"www.winegard.com/select/img/lg-show-charlie-rose.jpg"}, {"title":"Arthur","url":"www.winegard.com/select/img/lg-show-arthur.jpg"}, {"title":"Frontline","url":"www.winegard.com/select/img/lg-show-frontline.jpg"}]}]}
http://<server>/towers.php?lat=40.81&lon=-91.14&range=50	{ "return":true,"response":[{"id":"4327","city":"BURLINGTON","channel":"41","vchannel":"26","band":"UHF","callsign":"KGCW-TV","network":"CW","url":"www.winegard.com/select/img/lg-network-cw.png","latitude":"41.135555","longitude":"-90.808334","distance":"28.4"}, {"id":"39020","city":"MACOMB","channel":"21","vchannel":"22","band":"UHF","callsign":"WMEC","network":"PBS","url":"www.winegard.com/select/img/lg-network-pbs.png","latitude":"40.398335","longitude":"-90.731941","distance":"35.6"}]}
http://<server>/towers.php?lat=40.81&lon=-91.14&range=40&details=full	{ "return":true,"response":[{"id":"4327","city":"BURLINGTON","channel":"41","vchannel":"26","band":"UHF","callsign":"KGCW-TV","network":"CW","url":"www.winegard.com/select/img/lg-network-cw.png","latitude":"41.135555","longitude":"-90.808334","distance":"28.4"}, {"id":"39020","city":"MACOMB","channel":"21","vchannel":"22","band":"UHF","callsign":"WMEC","network":"PBS","url":"www.winegard.com/select/img/lg-network-pbs.png","latitude":"40.398335","longitude":"-90.731941","distance":"35.6","shows":[{"title":"Sesame Street","url":"www.winegard.com/select/img/lg-show-sesame-street.jpg"}, {"title":"Downton Abbey","url":"www.winegard.com/select/img/lg-show-downton-abbey.jpg"}, {"title":"America's Test Kitchen","url":"www.winegard.com/select/img/lg-show-americas-test-kitchen.jpg"}, {"title":"Charlie Rose","url":"www.winegard.com/select/img/lg-show-charlie-rose.jpg"}, {"title":"Arthur","url":"www.winegard.com/select/img/lg-show-arthur.jpg"}, {"title":"Frontline","url":"www.winegard.com/select/img/lg-show-frontline.jpg"}]}]}

Interface Control Document

Command	Response
http://<server>/towers.php?id=3182	{ "status":true,"data":[{"title":"Sesame Street","url":"www.winegard.com/select/img/lg-show-sesame-street.jpg"}, {"title":"Downton Abbey","url":"www.winegard.com/select/img/lg-show-downton-abbey.jpg"}, {"title":"America's Test Kitchen","url":"www.winegard.com/select/img/lg-show-americas-test-kitchen.jpg"}, {"title":"Charlie Rose","url":"www.winegard.com/select/img/lg-show-charlie-rose.jpg"}, {"title":"Arthur","url":"www.winegard.com/select/img/lg-show-arthur.jpg"}, {"title":"Frontline","url":"www.winegard.com/select/img/lg-show-frontline.jpg"}]}
http://<server>/towers.php?id=178	{"return":true,"response":"n/a"}
http://<server>/towers.php?id=10690	{"return":true,"response":"TELEMUNDO"}

3.4 Satellite Info

Summary:

This command retrieves the satellite providers and the list of satellites with pointing angles per provider.

Application Usage:

This GET command provides the list of satellite providers with IDs in alphabetical order or the list of provider's satellites with pointing angles. The pointing angles consist of three angles: azimuth (0 to 360 deg), elevation (0 to 95 deg), and skew (-180 to 180 deg). Latitude and longitude are to be provided in +/- decimal degrees. This command returns the command status and either the command failure code if a failure occurs or the requested data such as the list of satellite providers.

HTTP Verb:

GET

Command Structure:

Command	Response	URL	Parameters
Providers	Returns the list of satellite providers with image urls	/satellites.php	
Satellites	Returns list of satellites with pointing angles per provider	/satellites.php	id=<provider id>&lat=<latitude>&lon=<longitude>
Satellites - Full	Returns list of all satellites with pointing angles	/satellites.php	id=<>&lat=<latitude>&lon=<longitude>&details=full

Examples:

Command	Response
http://<server>/satellites.php	{ "status":true,"data":[{"id":"bell","provider":"Bell (Canada)","url":"www.winegard.com/select/img/lg-belltv.png"}, {"id":"directv","provider":"DirecTv","url":"www.winegard.com/select/img/lg-directv.png"}, {"id":"dish","provider":"Dish Network","url":"www.winegard.com/select/img/lg-dish.png"}, {"id":"shaw","provider":"Shaw (Canada)","url":"www.winegard.com/select/img/lg-shaw.png"}]}
http://<server>/satellites.php?id=dish&lat=40.81&lon=-91.14	{ "status":true,"data":[{"orb":-61,"az":"138.38","el":"33.65","sk":-29.39}, {"orb":-72,"az":"152.02","el":"38.88","sk":-20.27}, {"orb":-77,"az":"158.91","el":"40.62","sk":-15.41}, {"orb":-110,"az":"207.59","el":"38.99","sk":"20.00"}, {"orb":-119,"az":"218.96","el":"34.87","sk":"27.67"}, {"orb":-129,"az":"229.94","el":"29.10","sk":"34.44"}]}
http://<server>/satellites.php?id=directv&lat=40.81&lon=-91.14	{ "status":true,"data":[{"orb":-99,"az":"191.92","el":"42.12","sk":"8.77"}, {"orb":-101,"az":"194.89","el":"41.73","sk":"10.93"}, {"orb":-103,"az":"197.81","el":"41.26","sk":"13.05"}, {"orb":-110,"az":"207.59","el":"38.99","sk":"20.00"}, {"orb":-119,"az":"218.96","el":"34.87","sk":"27.67"}]}
http://<server>/satellites.php?lat=40.81&lon=-91.14&details=full	{ "status":true,"data":[{"id":"dish","orb":-61,"az":"138.38","el":"33.65","sk":-29.39}, {"id":"dish","orb":-72,"az":"152.02","el":"38.88","sk":-20.27}, {"id":"dish","orb":-77,"az":"158.91","el":"40.62","sk":-15.41}, {"id":"bell","orb":-82,"az":"166.17","el":"41.88","sk":-10.17}, {"id":"bell","orb":-91,"az":"179.78","el":"42.81","sk":-0.16}, {"id":"directv","orb":-99,"az":"191.92","el":"42.12","sk":"8.77"}, {"id":"directv","orb":-101,"az":"194.89","el":"41.73","sk":"10.93"}, {"id":"directv","orb":-103,"az":"197.81","el":"41.26","sk":"13.05"}, {"id":"shaw","orb":-107,"az":"203.49","el":"40.07","sk":"17.11"}, {"id":"directv","orb":-110,"az":"207.59","el":"38.99","sk":"20.00"}, {"id":"dish","orb":-110,"az":"207.59","el":"38.99","sk":"20.00"}, {"id":"shaw","orb":-111,"az":"208.92","el":"38.59","sk":"20.92"}, {"id":"directv","orb":-119,"az":"218.96","el":"34.87","sk":"27.67"}, {"id":"dish","orb":-119,"az":"218.96","el":"34.87","sk":"27.67"}, {"id":"dish","orb":-129,"az":"229.94","el":"29.10","sk":"34.44"}]}