



Simpleview Membership Management API v. 1.0

GETTING STARTED

Dear Developer,

Thank you for your interest in the Simpleview CRM API for Membership Management. This version of the API is currently limited to a one-way pull of listings data for display on a remote web site.

The following steps are required in order to get started. Have your client contact their project manager to set up your API Login and password and provide you with the client's base API URL. Once that is complete and you have this document in hand, you are ready to go. The base url will differ from client to client, but assuming Simpleview was the client, the URL would likely look like **<https://api.somesite.com>**.

The next step is to decide which version of the API will be easiest for you to use. There is either the SOAP API (also referred to in this documentation as Web Service), or the XML API. The SOAP API takes in objects over the SOAP protocol and returns them the same way. The XML API accepts data as name-value pairs (herein referred to as NVP's) and returns an XML document. In general, whether you choose the XML implementation or the Web Service (SOAP) implementation, the values passed back and forth are the same. The XML implementation uses the SOAP implementation as its base, so it would be rare to find functionality differences between the two, but there may from time to time be slight differences either in naming conventions or results returned. We will try to make note of any of these differences under their respective methods.

At the moment there are no code samples available for interfacing with the API, but we hope to get some set up shortly in most of the common web development languages such as PHP, JSP, ASP.NET, and Coldfusion.

UPDATES TO THE API AND THIS DOCUMENT

This version of the API as well as its documentation may be updated from time to time. This would encompass handling various tasks such as bug fixes and omissions. We will make every effort to alert and give you advanced warning before any changes are being made that may affect your implementation. This is likely to be limited to adding additional data or at worst minor modifications to formatting.

WHAT TO SEND AND WHERE TO SEND IT

WHERE TO SEND IT (XML)

All requests sent for listings data should be sent to the following URL:

<<clientbaseurl>>/webapi/listings/xml/listings.cfm

WHAT TO SEND (XML)

The method invoked is specified by passing the action to the API. All values should be sent via HTTP POST parameters. Any XML values should be sent with the full XML header and should always be encoded in UTF-8. Additional requirements may be specified in the future.

All XML requests require the following parameter specifying the request type.

Action: *String* The type of request you are making.

WHERE TO SEND IT (SOAP)

The URL for the listings Web Service is as follows:

<<clientbaseurl>>/webapi/listings/soap/listings.cfc?wsdl

WHAT TO SEND (SOAP)

The format required for each method differs and the values expected are documented with each individual method.

Note that the method names, when accessed via SOAP, are case-sensitive. In all cases, method names use the “camel-case” convention: the first “word” in the method name is un-capitalized, while the initial letter of each subsequent word is capitalized (e.g. getListingRegions).

WHAT TO SEND (GENERAL)

All requests require the following parameters:

Username: *String* The username you were assigned

Password: *String* The password you were assigned

GENERAL RESPONSE INFORMATION

In the XML Version, all response information is contained with the <response> root node. All responses you receive will contain the following information:

Data (Name varies): Object Containing the data to be received
The Contents, **name**, and format of the data object varies with each method.

RequestStatus: Container for all status information regarding the request sent and response received.
The contents of the RequestStatus object also vary with each method but at minimum they will include the following:

- **Status.HasErrors (Boolean)** – flag indicating all non-transport errors
- **Status.Errors (Array)** – Array of errors found. At this time all errors returned implies a programming error – either improper formatting of parameters or attempting to access a field(s) that does not exist. As such, they are not meant to be shown to the user. Each error contains the following fields:
 - ErrorType:** String
 - ErrorDetail:** String
 - ErrorMessage:** String
- **Status.Results (Integer)** – Number of results returned by the Method

The following is a sample response element shown in XML Format:

```
<?xml version="1.0" encoding="UTF-8"?>
<RESPONSE>
  <LISTINGTYPES>
    <LISTINGTYPE>
      <TYPEID>11</TYPEID>
      <WEBSITE>1</WEBSITE>
      <NAME>Web Listings</NAME>
    </LISTINGTYPE>
  </LISTINGTYPES>
  <REQUESTSTATUS>
    <ERRORS></ERRORS>
    <HASERRORS>0</HASERRORS>
    <RESULTS>1</RESULTS>
  </REQUESTSTATUS>
</RESPONSE>
```

METHOD OVERVIEW

The following methods are supported and documented in this API

- getListings
- getListing
- getListingTypes
- getListingRegions
- getCouponCats
- getListingAmenities
- getListingCats
- getListingSubCats
- getCoupons
- getCoupon

- getCouponsByCats
- updateHits
- getCouponsByListingID

API METHODS

GETLISTINGS – RETURNS STRUCT

ACTION: GETLISTINGS

METHOD NAME:

getListings

PARAMETERS:

STRING USERNAME

STRING PASSWORD

INTEGER PAGENUM

INTEGER PAGESIZE

STRUCT FILTERS (SEE APPENDIX A FOR FILTERS)

INTEGER DISPLAYAMENITIES

DESCRIPTION:

This is the most common method sent. This is the method used to search the listings and return back your result set. Paging is supported. A default page size of 15 is used when no page size is specified or when the page size is sent as 0 and the maximum page size allowed is 50.

GETLISTINGS DATA OBJECT (SOAP):

The data is sent back as an array of maps (also referred to hereafter as structs or structures). Each map by default contains the following fields.

Field	Type	Description
ListingID	Integer	Unique identifier for the Listing
TypeID	Integer	Unique identifier for the Listing Type of the Listing
TypeName	String	The Listing Type name of the Listing
AcctID	Integer	Unique identifier for the Account
Company	String	The listing's name
CatID	Integer	The ID of the listing's primary category
CatName	String	The name of the listing's primary category
SubCatID	Integer	The ID of the listing's primary subcategory
SubCatname	String	The name of the listing's primary subcategory
Addr1	String	The first line of the address associated with the listing
Addr2	String	The second line of the address associated with the listing
Addr3	String	The third line of the address associated with the listing
City	String	The city associated with the listing

State	String	The state associated with the listing
Zip	String	The zip code (may be Zip+4) of the address associated with the listing
Latitude	String	The Latitude of the Listing as it appears in CRM
Longitude	String	The Longitude of the Listing as it appears in CRM
TollFree	String	The listing's tollfree number
Phone	String	The listing's phone number
AltPhone	String	The listing's alternate phone number
Fax	String	The listing's fax number
Email	String	The listing's e-mail address
WebURL	String	The listing's webpage URL
RegionID	Integer	The Unique Identifier for the region for the listing's account
Region	String	The name of the region associated with the listing's account
AcctStatus	String	The account's status (generally not shown on web)
PrimaryContactFullName	String	The name of the contact associated with this listing
PrimaryContactTitle	String	The title of the contact associated with this listing
RankID	Integer	The ID of the membership level associated with this listing
RankName	String	The name of the membership level associated with this listing.
Description	String	Description of the listing
LogoFile	String	Name of the file to display the listing's logo
PhotoFile	String	Name of the file to display the listing's primary photo
ImgPath	String	The URL of the folder where the LogoFile and PhotoFile are located.
Created	Date	The date the Listing was created
LastUpdated	Date	The date the Listing was last updated
SocialMedia	Array	List of the Social media fields with related information

DATA OBJECT (XML):

An XML Sample of the default field set is provided below.

GETLISTINGS RESPONSE (XML REPRESENTATION)

```
<?xml version="1.0" encoding="UTF-8"?>
  <RESPONSE>
    <LISTINGS>
      <LISTING>
        <LISTINGID>41661</LISTINGID>
        <TYPEID>1</TYPEID>
        <TYPENAME>Web Listing</TYPENAME>
        <ACCTID>119764</ACCTID>
        <COMPANY>Simpleview Inc.</COMPANY>
        <CATID>1574</CATID>
        <CATNAME>Convention & Group Services</CATNAME>
        <SUBCATID>48</SUBCATID>
        <SUBCATNAME>Destination Management Companies</SUBCATNAME>
        <ADDR1>7430 N La Cholla Blvd</ADDR1>
        <ADDR2/>
      </LISTING>
    </LISTINGS>
  </RESPONSE>
```

```

<ADDR3/>
<CITY>Tucson</CITY>
<STATE>AZ</STATE>
<ZIP>85741</ZIP>
<LATITUDE />
<LONGITUDE />
<TOLLFREE/>
<PHONE>(520) 575-1151</PHONE>
<ALTPHONE/>
<FAX>(520) 575-1171</FAX>
<EMAIL>info@simpleviewinc.com</EMAIL>
<WEBURL>www.simpleviewinc.com</WEBURL>
<REGIONID>2</REGIONID>
<REGION>North</REGION>
<ACCTSTATUS>Active</ACCTSTATUS>
<PRIMARYCONTACTFULLNAME>R. George</PRIMARYCONTACTFULLNAME>
<PRIMARYCONTACTTITLE/>
<RANKID>1</RANKID>
<RANKNAME>Default</RANKNAME>
<DESCRIPTION>Listing description...</DESCRIPTION>
<LOGOFILE>sv_signature2.gif</LOGOFILE>
<PHOTOFILE/>
<IMGPATH>http://www.simpleviewinc.com/images/</IMGPATH>
<CREATED>01-01-2002</CREATED>
<LASTUPDATED>01-10-2009</LASTUPDATED>
</LISTING>
</LISTINGS>
<REQUESTSTATUS>
  <ERRORS></ERRORS>
  <HASERRORS>0</HASERRORS>
  <RESULTS>1</RESULTS>
  <PAGESIZE>2</PAGESIZE>
  <PAGENUM>1</PAGENUM>
</REQUESTSTATUS>
</RESPONSE>

```

GETLISTING – RETURNS STRUCT**ACTION: GETLISTING****METHOD NAME:**

getListing

PARAMETERS:

STRING USERNAME

STRING PASSWORD

INTEGER LISTINGID

INTEGER UPDATEHITS

DESCRIPTION:

This method is used to retrieve a single listing, given a Listing ID.

GETLISTING DATA OBJECT (SOAP):

The data is sent back as a map, containing the following fields by default.

Field	Type	Description
ListingID	Integer	Unique Identifier for the Listing
AcctID	Integer	Unique identifier for the Account
Company	String	The listing's name
CatID	Integer	The ID of the listing's primary category
CatName	String	The name of the listing's primary category
SubCatID	Integer	The ID of the listing's primary subcategory
SubCatname	String	The name of the listing's primary subcategory
Addr1	String	The first line of the address associated with the listing
Addr2	String	The second line of the address associated with the listing
Addr3	String	The third line of the address associated with the listing
City	String	The city associated with the listing
State	String	The state associated with the listing
Zip	String	The zip code (may be Zip+4) of the address associated with the listing
TollFree	String	The listing's tollfree number
Phone	String	The listing's phone number
AltPhone	String	The listing's alternate phone number
Fax	String	The listing's fax number
Email	String	The listing's e-mail address
WebURL	String	The listing's webpage URL
RegionID	Integer	The Unique Identifier for the region associated with this listing's account
Region	String	The name of the region associated with the listing's account
AcctStatus	String	The account's status (generally not shown on web)
PrimaryContactFullName	String	The name of the contact associated with this listing

PrimaryContactTitle	String	The title of the contact associated with this listing
RankID	Integer	The ID of the membership level associated with this listing
RankName	String	The name of the membership level associated with this listing (no rank signified by a value of 'Default')
Description	String	Description of the listing
LogoFile	String	Name of the file to display the listing's logo
PhotoFile	String	Name of the file to display the listing's primary photo
ImgPath	String	The URL of the folder where the LogoFile and PhotoFile are located.
Images	Array	An Array that holds all images associated to the Listing
Amenities	Array	An array of Amenity structures.
Coupons	Array	An array of Coupon structures. (see the getCoupons Method)
AdditionalSubCats	Array	An Array that lists all additional subcats assigned to the Listing
FacilityInformation	Array	An array that holds all of the Property Information
AdditionalInformation	Array	An array that contains key value pairs of Listing UDF data
Created	Date	The date the Listing was created.
UpdateDate	Date	The date the Listing was last updated.
SocialMedia	Array	List of the Social media fields with related information

The Amenities array consists of a series of structures which contain the following fields by default:

Field	Type	Description
Name	String	Name of the amenity
MultipleValues	Bit	If 1, the value of this amenity is itself an array
Value	Any	The value of the amenity, if MultipleValues is 0
ValueArray	Array	The array of values which make up the value of this amenity, if MultipleValues is 1.

DATA OBJECT (XML):

An XML Sample of the default fieldset is provided below.

GETLISTING RESPONSE (XML REPRESENTATION)

```
<?xml version="1.0" encoding="UTF-8"?>
  <RESPONSE>
    <LISTING>
      <LISTINGID>41661</LISTINGID>
      <ACCTID>119764</ACCTID>
      <COMPANY>Simpleview Inc.</COMPANY>
      <CATID>1574</CATID>
      <CATNAME>Convention & Group Services</CATNAME>
      <COUPONS>
        <COUPON>
          <COUPONID>46</COUPONID>
          <OFFERTITLE>Title and or short description goes here</OFFERTITLE>
          <OFFERTEXT>Long Description Goes here</ OFFERTEXT>
          <OFFERLINK>http://www.someurl.com</ OFFERLINK>
        </COUPON>
      </COUPONS>
    </LISTING>
  </RESPONSE>
```



```

        <REDEEMSTART>9/1/2009</REDEEMSTART>
        <REDEEMEND>10/1/2009</REDEEMEND>
        <POSTSTART>9/1/2009</ POSTSTART>
        <POSTEND>10/1/2009</ POSTEND>
        <COUPONCATS>
            <ITEM>
                <COUPONCATID>3</ COUPONCATID>
                <COUPONCATNAME>Hotels</COUPONCATNAME>
            </ITEM>
        </ COUPONCATS>
    </ COUPON>
</COUPONS>
<SUBCATID>48</SUBCATID>
<SUBCATNAME>Destination Management Companies (DMCs)</SUBCATNAME>
<ADDITIONALSUBCATS>
    <ITEM>
        <CATID>5</CATID>
        <CATNAME>A Category</ CATNAME>
        <SUBCATID>34</SUBCATID>
        <SUBCATNAME>A Sub-category</SUBCATNAME>
    </ ITEM>
</ADDITIONALSUBCATS>
<ADDR1>7430 N La Cholla Blvd</ADDR1>
<ADDR2/>
<ADDR3/>
<CITY>Tucson</CITY>
<STATE>AZ</STATE>
<ZIP>85741</ZIP>
<LONGITUDE />
<LATITUDE />
<TOLLFREE/>
<PHONE>(520) 575-1151</PHONE>
<ALTPHONE/>
<FAX>(520) 575-1171</FAX>
<EMAIL>info@simpleviewinc.com</EMAIL>
<WEBURL>www.simpleviewinc.com</WEBURL>
<REGIONID>2</REGIONID>
<REGION>North</REGION>
<ACCTSTATUS>Active</ACCTSTATUS>
<PRIMARYCONTACTFULLNAME>Dr. Ryan George, III</PRIMARYCONTACTFULLNAME>
<PRIMARYCONTACTTITLE/>
<RANKID>1</RANKID>
<RANKNAME>Default</RANKNAME>
<DESCRIPTION>Listing description...</DESCRIPTION>

```

```

<LOGOFILE>sv_signature2.gif</LOGOFILE>
<PHOTOFILE/>
<IMGPATH>http://www.simpleviewinc.com/images/admin/logos/</IMGPATH>
<AMENITIES>
  <ITEM>
    <NAME>Total Rooms</NAME>
    <MULTIPLEVALUES>0</MULTIPLEVALUES>
    <VALUE>650</VALUE>
    <VALUEARRAY/>
  </ITEM>
  <ITEM>
    <NAME>Room Service Offered</NAME>
    <MULTIPLEVALUES>1</MULTIPLEVALUES>
    <VALUE/>
    <VALUEARRAY>
      <ITEM>24-Hour</ITEM>
      <ITEM>Breakfast</ITEM>
      <ITEM>Dinner</ITEM>
    </VALUEARRAY>
  </ITEM>
</AMENITIES>
<FACILITYINFORMATION />
<PROPERTYROOMS />
<ADDITIONALINFORMATION />
<IMAGES />
<CREATED>10-12-2002</CREATED>
<LASTUPDATED>12-02-2009</LASTUPDATED>
</LISTING>
<REQUESTSTATUS>
  <ERRORS></ERRORS>
  <HASERRORS>0</HASERRORS>
  <RESULTS>1</RESULTS>
  <PAGESIZE>2</PAGESIZE>
  <PAGENUM>1</PAGENUM>
</REQUESTSTATUS>
</RESPONSE>

```

GETLISTINGTYPES – RETURNS STRUCT**ACTION: GETLISTINGTYPES****METHOD NAME:**

getListingTypes

PARAMETERS:

STRING USERNAME

STRING PASSWORD

BIT WEBONLY

DESCRIPTION:

This method is used to retrieve all listing types defined in the database. The results can optionally be restricted to only those types meant to display on the client's website.

GETLISTINGTYPES DATA OBJECT (SOAP):

The data is sent back as a map, containing the following fields by .

Field	Type	Description
TypeID	Integer	Unique Identifier for the Listing Type
TypeName	String	Name of the Listing Type
ShowWeb	Boolean	Whether or not this type is meant to display on the client's website

DATA OBJECT (XML):

An XML Sample of the default fieldset is provided below.

GETLISTINGTYPES RESPONSE (XML REPRESENTATION)

```
<?xml version="1.0" encoding="UTF-8"?>
  <RESPONSE>
    <LISTINGTYPES>
      <LISTINGTYPE>
        <TYPEID>1</TYPEID>
        <TYPENAME>Website</TYPENAME>
        <SHOWWEB>1</SHOWWEB>
      </LISTINGTYPE>
      <LISTINGTYPE>
        <TYPEID>2</TYPEID>
        <TYPENAME>Guide</TYPENAME>
        <SHOWWEB>0</SHOWWEB>
      </LISTINGTYPE>
    </LISTINGTYPES>
  </RESPONSE>
</xml>
```

```

</LISTINGTYPES>
<REQUESTSTATUS>
  <ERRORS></ERRORS>
  <HASERRORS>0</HASERRORS>
  <RESULTS>2</RESULTS>
</REQUESTSTATUS>
</RESPONSE>

```

GETLISTINGCATS – RETURNS STRUCT

ACTION: GETLISTINGCATS

METHOD NAME:

getListingCats

PARAMETERS:

STRING USERNAME

STRING PASSWORD

INTEGER LISTINGTYPEID

DESCRIPTION:

Returns list of all listing categories within the specified listing type. If the listing type is an empty string or set to 0, the function will return all active listing categories (whose listing types are active and set to show on the web).

GETLISTINGCATS DATA OBJECT (SOAP):

The data is sent back as a map, containing the following fields by default.

Field	Type	Description
CatID	Integer	Unique Identifier for the Listing Category
Name	String	Name of the Listing Category
ListingType	String	Name of the Listing Type
ListingTypeID	Integer	Unique Identifier for the Listing Type

DATA OBJECT (XML):

An XML Sample of the default fieldset is provided below.

GETLISTINGCATS RESPONSE (XML REPRESENTATION)

```
<?xml version="1.0" encoding="UTF-8"?>
  <RESPONSE>
    <LISTINGCATEGORIES>
      <LISTINGCATEGORY>
        <CATID>1571</CATID>
        <NAME>Accommodations</NAME>
        <LISTINGTYPE>Website</LISTINGTYPE>
        <LISTINGTYPEID>1</LISTINGTYPEID>
      </LISTINGCATEGORY>
      <LISTINGCATEGORY>
        <CATID>1573</CATID>
        <NAME>Attractions</NAME>
        <LISTINGTYPE>Website</LISTINGTYPE>
        <LISTINGTYPEID>1</LISTINGTYPEID>
      </LISTINGCATEGORY>
      <LISTINGCATEGORY>
        <CATID>1574</CATID>
        <NAME>Convention & Group Services</NAME>
        <LISTINGTYPE>Website</LISTINGTYPE>
        <LISTINGTYPEID>1</LISTINGTYPEID>
      </LISTINGCATEGORY>
    </LISTINGCATEGORIES>
    <REQUESTSTATUS>
      <ERRORS></ERRORS>
      <HASERRORS>0</HASERRORS>
      <RESULTS>3</RESULTS>
    </REQUESTSTATUS>
  </RESPONSE>
```

GETLISTINGSUBCATS – RETURNS STRUCT**ACTION: GETLISTINGSUBCATS****METHOD NAME:**

getListingSubCats

PARAMETERS:

STRING USERNAME

STRING PASSWORD

INTEGER LISTINGCATID

INTEGER LISTINGTYPEID

DESCRIPTION:

Returns list of all listing subcategories within the specified listing category and/or listing type. If the listing category/type is an empty string or set to 0, the function will return all active listing subcategories (whose listing categories and types are active, and the types are set to show on the web).

GETLISTINGSUBCATS DATA OBJECT (SOAP):

The data is sent back as a map, containing the following fields by default.

Field	Type	Description
SubCatID	Integer	Unique Identifier for the Listing Subcategory
SubCatName	String	Name of the Listing Subcategory
CatID	Integer	Unique Identifier for the Listing Category to which this subcategory belongs
CatName	String	Name of the Listing Category to which this subcategory belongs
ListingTypeID	Integer	Unique Identifier for the Listing Type to which this category belongs
ListingTypeName	String	Name of the Listing Type to which this category belongs

DATA OBJECT (XML):

An XML Sample of the default fieldset is provided below.

GETLISTINGSUBCATS RESPONSE (XML REPRESENTATION)

```
<?xml version="1.0" encoding="UTF-8"?>
  <RESPONSE>
    <LISTINGSUBCATEGORIES>
      <SUBCATEGORY>
        <SUBCATID>23</CATID>
        <SUBCATNAME>Apartments & Condos</SUBCATNAME>
        <CATID>1571</CATID>
        <CATNAME>Accommodations</CATNAME>
      </SUBCATEGORY>
    </LISTINGSUBCATEGORIES>
  </RESPONSE>
```

```
        <LISTINGTYPEID>1</LISTINGTYPEID>
        <LISTINGTYPENAME>Website</LISTINGTYPENAME>
    </SUBCATEGORY>
    <SUBCATEGORY>
        <SUBCATID>50</CATID>
        <SUBCATNAME>Guest Ranches</SUBCATNAME>
        <CATID>1571</CATID>
        <CATNAME>Accommodations</CATNAME>
        <LISTINGTYPEID>1</LISTINGTYPEID>
        <LISTINGTYPENAME>Website</LISTINGTYPENAME>
    </SUBCATEGORY>
    <SUBCATEGORY>
        <SUBCATID>71</CATID>
        <SUBCATNAME>Hotels & Resorts</SUBCATNAME>
        <CATID>1571</CATID>
        <CATNAME>Accommodations</CATNAME>
        <LISTINGTYPEID>1</LISTINGTYPEID>
        <LISTINGTYPENAME>Website</LISTINGTYPENAME>
    </SUBCATEGORY>
</LISTINGSUBCATEGORIES>
<REQUESTSTATUS>
    <ERRORS></ERRORS>
    <HASERRORS>0</HASERRORS>
    <RESULTS>3</RESULTS>
</REQUESTSTATUS>
</RESPONSE>
```

GETLISTINGREGIONS – RETURNS STRUCT**ACTION: GETLISTINGREGIONS****METHOD NAME:**

getListingRegions

PARAMETERS:

STRING USERNAME

STRING PASSWORD

NUMERIC CATID

DESCRIPTION:

Returns a list of all regions of listings which belong to the specified category. If the listing category is an empty string or set to 0, the function will return all active regions.

GETLISTINGREGIONS DATA OBJECT (SOAP):

The data is sent back as a map, containing the following fields by default.

Field	Type	Description
RegionID	Integer	Unique Identifier for the Listing Region
Region	String	Name of the Listing Region

DATA OBJECT (XML):

An XML Sample of the default fieldset is provided below.

GETLISTINGREGIONS RESPONSE (XML REPRESENTATION)

```
<?xml version="1.0" encoding="UTF-8"?>
  <RESPONSE>
    <LISTINGREGIONS>
      <LISTINGREGION>
        <REGIONID>1</REGION>
        <REGION>Downtown</REGION>
      </LISTINGREGION>
      <LISTINGREGION>
        <REGIONID>2</REGION>
        <REGION>North</REGION>
      </LISTINGREGION>
      <LISTINGREGION>
        <REGIONID>3</REGION>
        <REGION>South</REGION>
      </LISTINGREGION>
    </LISTINGREGIONS>
  </RESPONSE>
</xml>
```



```

    <REQUESTSTATUS>
      <ERRORS></ERRORS>
      <HASERRORS>0</HASERRORS>
      <RESULTS>3</RESULTS>
    </REQUESTSTATUS>
  </RESPONSE>

```

GETLISTINGAMENITIES – RETURNS STRUCT

ACTION: GETLISTINGAMENITIES

METHOD NAME:

getListingAmenities

PARAMETERS:

STRING USERNAME

STRING PASSWORD

DESCRIPTION:

Returns a list of all amenities

GETLISTINGAMENITIES DATA OBJECT (SOAP):

The data is sent back as a map, containing the following fields by default.

Field	Type	Description
FieldID	Integer	Unique Identifier for the amenity field
UDFTypeID	Integer	The identifier for the data type of this amenity field. See Appendix B: UDFTypes for a list of possible values.
TypeName	String	The name of the data type of this amenity field. See Appendix B: UDFTypes for a list of possible values.
Label	String	The screen label for this amenity field
Decimal	Integer	The number of positions after the decimal (only if the UDFTypeID indicates that this amenity is a number).
AmenityTypeName	String	The name of the Amenity Type to which this amenity belongs. In the CRM UI, this corresponds to the heading bar underneath which the amenity appears.
SubTypeName	String	The name of the Subtype to which this amenity belongs. In the CRM UI, this corresponds to the tab within the Property tab on which the amenity appears.

DATA OBJECT (XML):

An XML Sample of the default fieldset is provided below.

GETLISTINGAMENITIES RESPONSE (XML REPRESENTATION)

```
<?xml version="1.0" encoding="UTF-8"?>
  <RESPONSE>
    <AMENITIES>
      <AMENITY>
        <FIELDID>1</FIELDID>
        <UDFTYPEID>10</UDFTYPEID>
        <TYPENAME>URL</TYPENAME>
        <LABEL>Map URL</LABEL>
        <DECIMAL />
        <AMENITYGROUPNAME>General</AMENITYGROUPNAME>
        <AMENITYTABNAME>General</AMENITYTABNAME>
      </AMENITY>
      <AMENITY>
        <FIELDID>2</FIELDID>
        <UDFTYPEID>4</UDFTYPEID>
        <TYPENAME>Number</TYPENAME>
        <LABEL>Total Rooms</LABEL>
        <DECIMAL>0</DECIMAL>
        <AMENITYGROUPNAME>General</AMENITYGROUPNAME>
        <AMENITYTABNAME>Accommodations</AMENITYTABNAME>
      </AMENITY>
      <AMENITY>
        <FIELDID>8</FIELDID>
        <UDFTYPEID>12</UDFTYPEID>
        <TYPENAME>Multi-Select</TYPENAME>
        <LABEL>Room Service Offered</LABEL>
        <DECIMAL />
        <AMENITYGROUPNAME>General</AMENITYGROUPNAME>
        <AMENITYTABNAME>Accommodations</AMENITYTABNAME>
      </AMENITY>
    </AMENITIES>
    <REQUESTSTATUS>
      <ERRORS></ERRORS>
      <HASERRORS>0</HASERRORS>
      <RESULTS>3</RESULTS>
    </REQUESTSTATUS>
  </RESPONSE>
```

GETCOUPONCATS – RETURNS STRUCT**ACTION: GETCOUPONCATS****METHOD NAME:**

getCouponCats

PARAMETERS:

STRING USERNAME

STRING PASSWORD

DESCRIPTION:

Returns a list of all coupon categories

GETLISTINGREGIONS DATA OBJECT (SOAP):

The data is sent back as a map, containing the following fields by default.

Field	Type	Description
CouponCatId	Integer	Unique Identifier for the Coupon Category
CouponCatName	String	Name of the Coupon Category

DATA OBJECT (XML):

An XML Sample of the default fieldset is provided below.

GETCOUPONCATS RESPONSE (XML REPRESENTATION)

```
<?xml version="1.0" encoding="UTF-8"?>
  <RESPONSE>
    <COUPONCATEGORIES>
      <COUPONCATEGORY>
        <COUPONCATNAME>Hotel Offers</COUPONCATNAME>
        <COUPONCATID>1</ COUPONCATID >
      </COUPONCATEGORY>
      <COUPONCATEGORY>
        <COUPONCATNAME>Restaurant Offers</COUPONCATNAME>
        <COUPONCATID>2</ COUPONCATID >
      </COUPONCATEGORY>
    </ COUPONCATEGORIES>
  </RESPONSE>
```

GETCOUPONS – RETURNS STRUCT**ACTION: GETCOUPONS****PARAMETERS:***STRING USERNAME**STRING PASSWORD**INTEGER PAGENUM**INTEGER PAGESIZE**STRUCT FILTERS (SEE APPENDIX A FOR FILTERS)***DESCRIPTION:**

This is the method used to search the coupons and return back your result set. It uses the same mechanic a getListings to return a datasets based on a filter. Paging is supported. A default page size of 15 is used when no page size is specified or when the page size is sent as 0 and the maximum page size allowed is 50. The API only exposes coupons that are NOT pending and do not have expired postend date, coupons with a null postend date are treated as infinitely active.

GETCOUPONS DATA OBJECT (SOAP):

The data is sent back as an array of maps (also referred to hereafter as structs or structures) of the type coupons. Please notice the difference between Post Start / End date and Redeem Start / End Date. Ideally coupons should be displayed and filtered on based on the Post Dates NOT the Redeem Dates. Each map by default contains the following fields.

Field	Type	Description
CouponID	Integer	The unique identifier of the coupon
Acctid	Integer	The foreign key reference to the Account
ListingID	Integer	The foreign key reference to the Listing, Note an account can have many listings, an account can also have many coupons. The coupons relationship to the listing is inferred by the listings and coupons association to the account.
CouponCatName	String	The name of the couponcategory that the coupon belongs to a coupon can belong to 0 to many categories.
CouponCatID	Integer	The foreign key reference to the coupon category, a coupon can belong to many categories, its recommended that filter includes a CouponCatID.
CreateDate	DateTime	The date the coupon was created, typically not displayed
UpdateDate	DateTime	The last update date of the coupons record.
OfferLink	String	An optional link to a website
OfferText	String	The long description of the coupon
OfferTitle	String	A title / short description of the coupon
PostStart	DateTme	The date on which the coupons should start appearing to the public.
PostEnd	DateTime	The date on which the coupons should stop appearing to the public
RedeemStart	DateTime	The starting date on which the coupon can be redeemed

RedeemEnd	DateTime	The ending date on which the coupon can be redeemed
MediaID	Integer	The ID of the coupon's image
MediaFile	String	The name of the coupon's media file
MediaName	String	The user-friendly name of the coupon's media file
MediaTypeID	Integer	The type ID of the coupon's media type
MediaType	String	The name of the coupon's media type
ImgPath	String	The URL of the folder where the MediaFile is located

DATA OBJECT (XML):

An XML Sample of the default fieldset is provided below.

```
<?xml version="1.0" encoding="UTF-8"?>
  <RESPONSE>
    <COUPONS>
      <COUPON>
        <COUPONCATNAME>Hotel Special Offers</COUPONCATNAME>
        <POSTSTART>10/1/2009</POSTSTART>
        <OFFERTEXT>$10 Off Rooms when booking a room</OFFERTEXT>
        <UPDATEDATE>12/1/2009</UPDATEDATE>
        <POSTEND>11/1/2009</POSTEND>
        <OFFERLINK>http://www.somelink.com</OFFERLINK>
        <ACCTID>1</ACCTID>
        <COMPANY>Simpleview</COMPANY>
        <COUPONID>1</ COUPONID>
        <CREATEDATE>07/15/2009</CREATEDATE>
        <REDEEMEND>11/15/2009</ REDEEMEND>
        <REDEEMSTART>10/15/2009</ REDEEMSTART>
        <OFFERTITLE>$10 Off </ OFFERTITLE>
        <MEDIAID>1</ MEDIAID >
        <MEDIAFILE> LSTNG-HotelCoupon.jpg</MEDIAFILE>
        <MEDIANAME>Hotel on Coupon</MEDIANAME>
        <MEDIATYPEID>2</MEDIATYPEID>
        <MEDIATYPE>Image</MEDIATYPE>
        <IMGPATH> http://www.simpleviewinc.com/images/admin/logos/
          </IMGPATH>
      </COUPON>
    </COUPONS>
  </RESPONSE>
```

GETCOUPONSBYCATS – RETURNS STRUCT**ACTION: GETCOUPONSBYCATS****PARAMETERS:***STRING USERNAME**STRING PASSWORD**INTEGER COUPONCATID**INTEGER PAGENUM**INTEGER PAGESIZE**STRUCT FILTERS (SEE APPENDIX A FOR FILTERS)***DESCRIPTION:**

This is the method used to search the coupons and return back your result set by CouponCategoryID. It uses the same mechanic a `getCoupons` to return a datasets based on a filter. Paging is supported. A default page size of 15 is used when no page size is specified or when the page size is sent as 0 and the maximum page size allowed is 50. The API only exposes coupons that are NOT pending and do not have expired postend date, coupons with a null postend date are treated as infinitely active.

GETCOUPONSBYCATS DATA OBJECT (SOAP):

Refer to the method `GetCoupons` for an example of what this method returns.

DATA OBJECT (XML):

Refer to the method `GetCoupons` for an example of what this method returns.

GETCOUPONSBYLISTINGID – RETURNS STRUCT**ACTION: GETCOUPONSBYLISTINGID****PARAMETERS:***STRING USERNAME**STRING PASSWORD**INTEGER LISTINGID**INTEGER PAGENUM**INTEGER PAGESIZE**DATE REDEEMSTART**DATE REDEEMEND**STRING KEYWORDS*

DESCRIPTION:

This is the method used to search the coupons and return back your result set by listingID. It differs from all other methods of this type and allows you to filter only on redeem start and end dates as well as a keyword. The returned coupon's redeem start date must be on or after the parameter provided, and the redeem end date must be on or prior to the provided date. The keyword searches in the offer text and title. Redeem start and end dates and keywords are optional and can also be empty strings. Paging is supported. A default page size of 15 is used when no page size is specified or when the page size is sent as 0 and the maximum page size allowed is 50. The API only exposes coupons that are NOT pending and do not have expired postend date, coupons with a null postend date are treated as infinitely active.

GETCOUPONSBYLISTINGID DATA OBJECT (SOAP):

Refer to the method GetCoupons for an example of what this method returns.

DATA OBJECT (XML):

Refer to the method GetCoupons for an example of what this method returns.

GETCOUPON – RETURNS STRUCT**ACTION: GETCOUPON****METHOD NAME:**

getCoupon

PARAMETERS:

STRING USERNAME

STRING PASSWORD

INTEGER COUPONID

INTEGER UPDATEHITS

DESCRIPTION:

Returns a single coupon structure.

GETCOUPON DATA OBJECT (SOAP):

The data is sent back as a map, containing the following fields. It has the same structure as the coupons returned by get getCoupon

Field	Type	Description
CouponID	Integer	The unique identifier of the coupon
Acctid	Integer	The foreign key reference to the Account
ListingID	Integer	The foreign key reference to the Listing, Note an account can have

		many listings, an account can also have many coupons. The coupons relationship to the listing is inferred by the listings and coupons association to the account.
CouponCatName	String	The name of the couponcategory that the coupon belongs to a coupon can belong to 0 to many categories.
CouponCatID	Integer	The foreign key reference to the coupon category, a coupon can belong to many categories, its recommended that filter includes a CouponCatID.
CreateDate	DateTime	The date the coupon was created, typically not displayed
UpdateDate	DateTime	The last update date of the coupons record.
OfferLink	String	An optional link to a website
OfferText	String	The long description of the coupon
OfferTitle	String	A title / short description of the coupon
PostStart	DateTme	The date on which the coupons should start appearing to the public.
PostEnd	DateTime	The date on which the coupons should stop appearing to the public
RedeemStart	DateTime	The starting date on which the coupon can be redeemed by the customer
RedeemEnd	DateTime	The ending date on which the coupon can be redeemed by the customer
MediaID	Integer	The ID of the coupon's image
MediaFile	String	The name of the coupon's media file
MediaName	String	The user-friendly name of the coupon's media file
MediaTypeID	Integer	The type ID of the coupon's media type
MediaType	String	The name of the coupon's media type
ImgPath	String	The URL of the folder where the MediaFile is located

DATA OBJECT (XML):

An XML Sample of the default fieldset is provided below.

GETCOUPON RESPONSE (XML REPRESENTATION)

```
<?xml version="1.0" encoding="UTF-8"?>
  <RESPONSE>
    <COUPON>
      <COUPONCATNAME>Hotel Special Offers</COUPONCATNAME>
      <POSTSTART>10/1/2009</POSTSTART>
      <OFFERTEXT>$10 Off Rooms when booking a room</OFFERTEXT>
      <UPDATEDATE>12/1/2009</UPDATEDATE>
      <POSTEND>11/1/2009</POSTEND>
      <OFFERLINK>http://www.somelink.com</OFFERLINK>
      <ACCTID>1</ACCTID>
      <COMPANY>Simpleview</COMPANY>
      <COUPONID>1</ COUPONID>
      <CREATEDATE>07/15/2009</CREATEDATE>
    </COUPON>
  </RESPONSE>
</xml>
```



```

<REDEEMEND>11/15/2009</ REDEEMEND>
<REDEEMSTART>10/15/2009</ REDEEMSTART>
<OFFERTITLE>$10 Off </ OFFERTITLE>
<MEDIAID>1</ MEDIAID >
<MEDIAFILE> LSTNG-HotelCoupon.jpg</MEDIAFILE>
<MEDIANAME>Hotel on Coupon</MEDIANAME>
<MEDIATYPEID>2</MEDIATYPEID>
<MEDIATYPE>Image</MEDIATYPE>
<IMGPATH> http://www.simpleviewinc.com/images/admin/logos/</IMGPATH>
<COUPON>
</RESPONSE>

```

UPDATEHITS – RETURNS BOOLEAN

ACTION: UPDATEHITS

METHOD NAME:

updateHits

PARAMETERS:

STRING USERNAME

STRING PASSWORD

INTEGER HITTYPEID (SEE APPENDIX C FOR HITTYPEIDS)

INTEGER RECID

DATE HITDATE

DESCRIPTION:

This method is used to update the hit count for listings, coupons, click-thrus etc. The RecID is the ID of the object whose hit counter needs to be incremented and it needs to match the type of the hit ID. So, if the hit type is Listing Views, the RecID must be the ListingID; if the hit type is Coupons then RecID must be the applicable CouponID. Some inactive hit types (for example, Categories and Subcategories) may require a customization in addition to being activated. Please see the list of all default hit types in Appendix C.

APPENDIX A: FILTER REFERENCE

FILTER AND FILTER GROUP OVERVIEW

A Filter is a data object used to specify which listings the `getListings` method should return. It represents a condition that the listing must meet in order to be returned in the Response structure's Listings structure.

A filter is defined by 4 fields – the field to check (broken out into a `fieldname` and a `fieldcategory`), the operator to apply, and the value to compare it to.

A Filter Group is a set of one or more filters, along with a logical operator that tells the `getListings` method how to combine these filters (if there is more than one). The `Filters` parameter to the `getListings` method should in fact be a Filter Group structure.

FILTERGROUP STRUCTURE

FILTERGROUP DATA OBJECT (SOAP):

The data is sent as a map, containing the following fields.

Field	Type	Description
Filters	Array	An array of Filter objects
AndOr	String	Either "And" or "Or". Indicates how the individual filters' conditions should be logically joined.
TypeName	String	The name of the data type of this amenity field. See Appendix B: UDFTypes for a list of possible values.

FILTERGROUP DATA OBJECT EXAMPLE (XML):

An XML fragment of a FilterGroup object is provided below.

```
<FILTERGROUP>
  <FILTERS>
    <ITEM>
      <FILTERTYPE>GREATER THAN</FILTERTYPE>
      <FIELDCATEGORY>LISTING</FIELDCATEGORY>
      <FILTERVALUE>20</FILTERVALUE>
      <FIELDNAME>LISTINGID</FIELDNAME>
    </ITEM>
    <ITEM>
      <FILTERTYPE>NOT EQUAL TO</FILTERTYPE>
      <FIELDCATEGORY>LISTING</FIELDCATEGORY>
      <FILTERVALUE>10</FILTERVALUE>
      <FIELDNAME>CATID</FIELDNAME>
    </ITEM>
  </FILTERS>
  <ANDOR>AND</ANDOR>
</FILTERGROUP>
```

ADVANCED FILTERGROUP DATA OBJECT EXAMPLE (XML):

```
<FILTERGROUP>
  <FILTERS>
    <ITEM>
      <FILTERTYPE>GREATER THAN</FILTERTYPE>
      <FIELDCATEGORY>LISTING</FIELDCATEGORY>
      <FILTERVALUE>20</FILTERVALUE>
      <FIELDNAME>LISTINGID</FIELDNAME>
    </ITEM>
    <ITEM>
      <FILTERTYPE>NOT EQUAL TO</FILTERTYPE>
      <FIELDCATEGORY>LISTING</FIELDCATEGORY>
      <FILTERVALUE>10</FILTERVALUE>
      <FIELDNAME>CATID</FIELDNAME>
    </ITEM>
    <ITEM>
      <FILTERS>
        <ITEM>
          <FILTERTYPE>LESS THAN</FILTERTYPE>
          <FIELDCATEGORY>LISTING</FIELDCATEGORY>
          <FILTERVALUE>6000</FILTERVALUE>
          <FIELDNAME>LISTINGID</FIELDNAME>
        </ITEM>
        <ITEM>
          <FILTERTYPE>NOT EQUAL TO</FILTERTYPE>
          <FIELDCATEGORY>LISTING</FIELDCATEGORY>
          <FILTERVALUE>25</FILTERVALUE>
          <FIELDNAME>CATID</FIELDNAME>
        </ITEM>
      </FILTERS>
      <ANDOR>OR</ANDOR>
    </ITEM>
  </FILTERS>
  <ANDOR>AND</ANDOR>
</FILTERGROUP>
```

FILTER STRUCTURE

FILTER DATA OBJECT (SOAP):

The data is sent as a map, containing the following fields.

Field	Type	Description
FieldCategory	String	DETERMINES THE FIELD CATEGORY OF THE FIELD WHICH THIS FILTER USES. SEE THE “ FILTER DATA OBJECT EXAMPLE (XML): <FILTERS> <ITEM>

		<pre> <FIELDNAME> CATID</FIELDNAME> <FIELDCATEGORY> LISTING</FIELDCATEGORY> <FILTERTYPE> EQUAL TO</FILTERTYPE> <FILTERVALUE>12</FILTERVALUE> </ITEM> </FILTERS> </pre> <p>Default FieldName & FieldCategory Values” section for a default list of valid FieldCategory/FieldName pairs.</p>
FieldName	String	<p><i>DETERMINES THE FIELD NAME OF THE FIELD WHICH THIS FILTER USES. SEE THE “</i></p> <p><i>FILTER DATA OBJECT EXAMPLE (XML):</i></p> <pre> <FILTERS> <ITEM> <FIELDNAME> CATID</FIELDNAME> <FIELDCATEGORY> LISTING</FIELDCATEGORY> <FILTERTYPE> EQUAL TO</FILTERTYPE> <FILTERVALUE>12</FILTERVALUE> </ITEM> </FILTERS> </pre> <p>Default FieldName & FieldCategory Values” section for a default list of valid FieldCategory/FieldName pairs.</p>
FilterType	String	Determines the type of comparison to be performed. See the “FilterType Values” section for a list of valid filtertypes
FilterValue	Any	The value against which the listing field will be compared.

FILTER DATA OBJECT EXAMPLE (XML):

```

<FILTERS>
<ITEM>
<FIELDNAME> CATID</FIELDNAME>
<FIELDCATEGORY> LISTING</FIELDCATEGORY>
<FILTERTYPE> EQUAL TO</FILTERTYPE>
<FILTERVALUE>12</FILTERVALUE>
</ITEM>
</FILTERS>

```

DEFAULT FIELDNAME & FIELDCATEGORY VALUES

The following table gives the default values which can be used for FieldCategory and FieldName in a Filter structure. To have any additional relevant fields allowed in your filters, please contact the project manager assigned to the project and let them know which additional client fields you will need.

FieldCategory	FieldName	FieldType	Description
Listing	ListingID	Numeric	The unique identifier for a listing

Listing	AcctID	Numeric	The unique identifier for the CRM account for the listing
Listing	Company	Text	The name of the company
Listing	TypeID	Numeric	The unique identifier for the type for the listing
Listing	TypeName	Text	The name of the type for the listing
Listing	CatID	Numeric	The unique identifier for the category for the listing
Listing	CatName	Text	The name of the category for the listing
Listing	SubCatID	Numeric	The unique identifier for the subcategory for the listing
Listing	SubCatName	Text	The name of the subcategory for the listing
Listing	Addr1	Text	The first line of the listed address
Listing	Addr2	Text	The second line of the listed address
Listing	City	Text	The city of the listed address
Listing	Stated	Text	The two-letter state or province abbreviation of the listed address
Listing	Zip	Text	The zip code or postal code of the listed address
Listing	StatusID	Numeric	The numeric status identifier for the CRM account for the listing
Listing	Status	Text	The status of the CRM account for the listing
Listing	FullName	Text	The full name of the contact for the listing
Listing	Title	Text	The job title of the contact for the listing
Listing	Region	Text	The region of the listing
Listing	RankID	Numeric	The rank identifier for the listing
Listing	RankName	Text	The rank name for the listing
Listing	Tollfree	Text	The toll-free phone number associated with the listing
Listing	Phone	Text	The phone number associated with the listing
Listing	AltPhone	Text	An alternate phone number associated with the listing
Listing	Fax	Text	The fax number associated with the listing
Listing	Email	Text	The email address associated with the listing
Listing	WebURL	Text	The URL of the website associated with the listing
Account	AcctID	Numeric	The unique identifier for the CRM account for the listing
Account	Company	Text	The name of the company
Account	SortCompany	Text	The name of the company as is used to sort. For example, a company named "The Hotel" might have a SortCompany of "Hotel, The".
Account	Formerly	Text	A previous name for the company
Coupon	CouponID	Numeric	The unique identifier for a coupon
Coupon	AcctID	Numeric	The unique identifier for the CRM account for a coupon
Coupon	ListingID	Numeric	The unique identifier for the CRM listing for a coupon.

Coupon	CouponCatID	Numeric	The unique identifier for the category for a coupon
Coupon	CouponCatName	Text	The name of the category for a coupon
Coupon	RedeemStart	Date	The redeem start date for a coupon
Coupon	RedeemEnd	Date	The redeem end date for a coupon
Coupon	PostStart	Date	The post start date for a coupon
Coupon	PostEnd	Date	The post end date for a coupon
Coupon	OfferTitle	Text	The title text for a coupon
Coupon	OfferText	Text	The associated body text for a coupon
Custom_Account	*	*	*Any non-Multi-Select Custom Field defined in the CRM application for Member Accounts will have a FieldCategory of "Custom_Account" and a FieldName of the label for that custom field.
Amenity	*	*	* Any non-Multi-Select amenity defined in the CRM application will have a FieldCategory of "Amenity" and a FieldName of the label for that amenity
Custom_Account_MS	*	*	*Any Multi-Select Custom Field defined in the CRM application for Member Accounts will have a FieldCategory of "Custom_Account_MS" and a FieldName of the label for that custom field.
Amenity_MS	*	*	* Any Multi-Select amenity defined in the CRM application will have a FieldCategory of "Amenity_MS" and a FieldName of the label for that amenity. If filtering by more than one of the multiple values, FilterValue must contain an array with the required values in each ITEM.
Facility	*	*	Any Facility amenity defined in the CRM application will have a FieldCategory of "Facility" and a FieldName of the column name (or a label) for that amenity

FILTERTYPE VALUES

The following table gives the values which can be used for FilterType in a Filter structure. When the FilterType indicates "Single Value", that means that the FilterValue allows for a single data value. When the FilterType indicates "Multiple Value", that means that the FilterValue allows for an array.

FilterType	Single Value	Multiple Value	Description
Equal To	Yes	No	The filter will be satisfied if the specified field is exactly equal to the specified value.

Not Equal To	Yes	No	The filter will be satisfied if the specified field is not exactly equal to the specified value.
Greater Than	Yes	No	The filter will be satisfied if the specified field is greater than the specified value.
Greater Than Or Equal To	Yes	No	The filter will be satisfied if the specified field is greater than or equal to the specified value.
Less Than	Yes	No	The filter will be satisfied if the specified field is less than the specified value.
Less Than Or Equal To	Yes	No	The filter will be satisfied if the specified field is less than or equal to the specified value.
Like	Yes	No	The filter will be satisfied if the specified field contains the specified value.
Starts With	Yes	No	The filter will be satisfied if the specified field starts with specified value. Used for strings in non-custom filter categories.
Ends With	Yes	No	The filter will be satisfied if the specified field ends with specified value. Used for strings in non-custom filter categories.
Any	No	Yes	The filter will be satisfied if any value within the specified field matches any of the specified values.
All	No	Yes	The filter will be satisfied if all values within specified field match one of the specified values.
None	No	Yes	The filter will be satisfied if no value within specified field matches one of the specified values.

APPENDIX B: UDFTYPES

The following values can be returned for the UDFTypeID and TypeName fields in the Amenity structure.

UDF Type ID	Type Name
1	Currency
2	Date
3	Email
4	Number
5	Percent
6	Phone
7	Dropdown
8	Text
9	Text Area
10	URL
11	Yes/No
12	Multi-Select
13	CC Number
14	CC Expiration
15	File Upload

APPENDIX C: HITTYPES

The following values are Listing Hit Types, which should be passed in the updateHits() function. Not all of them might be active at all times.

Hit Type ID	Hit Type	Active
1	Listings	1
2	Categories	0
3	Sub-Categories	0
4	Listing Website	1
5	Coupons	1
6	Locations	1
7	Reservations	0
12	Twitter Views	0
13	Twitter Click-thrus	0
14	Facebook Views	0
15	Facebook Click-thrus	0
16	Mobile Click-thrus	0
17	Mobile Views	0
18	Click to Call	0
58	Added to Itinerary	1
59	Map View	1