|  |  |
| --- | --- |
|  | wvrap_logo.jpg  **Privileges – Web Services**  **Technical Specification Document**  Version 0.9.9  *28/04/2015*  *Author: Andrew Crawford*  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  "LEGAL NOTICE: This document(s) and the information disclosed herein are the proprietary and confidential data of Wyndham Vacations and Resorts Asia Pacific Pty Ltd.  Neither this document(s) nor the information contained within shall be distributed or reproduced without the express written authorization of Wyndham Vacations and Resorts Asia Pacific Pty Ltd." |

**Document Revisions**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Change** |
| 0.1 | 11/11/2014 | Andrew Crawford | Initial Documentation Creation |
| 0.2 | 20/11/2014 | Andrew Crawford | After writing some unit tests made sure all request / response variables appear in lower case since that is how the final system will work |
| 0.3 | 21/11/2014 | Andrew Crawford | Added a description for the Retrieve Order Amount service called from the Wyndham payment gateway and added a note that the Final Checkout service gets called from the payment gateway. |
| 0.4 | 02/12/2014 | Andrew Crawford | Added some Webservice URLs  Added the section with the public URLs  Made some slight modification to the Verification services (to make it clear what each one does) |
| 0.5 | 02/01/2015 | Andrew Crawford | Added more Webservice URLs  Noted on the address related services the country key needs to be sent instead of free text.  Noted that the default Australian address is ‘AUS’. This should keep it consistent with the Australia Country key used for member’s addresses.  Added the Country Service to return all countries.  Added that the card request history service returns a card type (for when this starts to support multiple types) |
| 0.6 | 20/01/2015 | Andrew Crawford | Changed the Holiday Services to reflect the Shop Services.  Renamed these from Deal Service to Holiday Service to avoid confusion with the hot deals service.  Added Shop and Holiday Service URLs  Added the Membership Purchase Amount Service  Added the Update Membership Purchase Service  Added Shop Order History Service  Added Countdown Calendar Service  Added Welcome Message Service  Added Promo Banner Service  Added Primary Banner Service  Added Frequent Values URL Service  Added Page Tracking Service  Added the MyAccount Webservices  Removed a few fields from the Member Read Service. There were individual services to look at calling these and some of them may need to be turned off anyway. |
| 0.7 | 22/01/2015 | Andrew Crawford | Some updates mostly concerned with the Payment Gateway and associated services. The Create Member service no longer takes a payment receipt, and a new service is created to handle updating the payment receipt against a member. |
| 0.8 | 12/02/2015 | Andrew Crawford | Added the Member Benefits Service.  Updated the Shop Get Category service to include information for the Shop Categories .  page.  Added the ‘NOTCREATED’ status to the Member Create to cater for an Owner number being valid, but the Privileges process not creating the record yet.  Added the VIEW field to the Holiday Service.  Added the MiniVac Eligibility Service. |
| 0.9 | 19/02/2015 | Andrew Crawford | Added the New Card Request  Dropped the WWID requirement.  Changed the public login details from an email to a Privileges White number (WHXXXX).  Updated the header numbering (since it seemed to be incorrect).  Added the Featured Product Service.  Added the Reward Form Service. |
| 0.9.1 | 06/03/2015 | Andrew Crawford | Added more URL’s for finished services |
| 0.9.2 | 11/03/2015 | Andrew Crawford | Update Password and TOC – Updated this for Old + New Password changes and added notes on the error conditions this returns.  Updated Featured Product Service to match implementation. |
| 0.9.3 | 13/03/2015 | Andrew Crawford | Updates the MyAccount service to better reflect the screens and functions.  Made sure the password is not mandatory on create member |
| 0.9.4 | 19/03/2015 | Andrew Crawford | Updated the General Enquiry Service to better reflect the screen.  Updated the Categories Service after checking the real mandatory fields (sub-category should not have been mandatory)  Added the MyAccount URL. |
| 0.9.5 | 23/03/2015 | Andrew Crawford | Update on the category service (with mandatory fields).  Dropped deal\_order from the Holiday Service (since it made no sense to have it here).  Update of the Holiday service to return all fields shown on the Prototype. This adds ‘rooms’ and ‘amenities’ to this.  Added subName and conditions to the Holiday Service fields.  Made Price in Holiday Service and Summary a text (since it’s not always a $ amount) |
| 0.9.6 | 26/03/2015 | Fernando Alves | * Updated URL for service “Order Amount Service” * Updated URL for “Initial Checkout Service” * Updated Address fields on the “Initial Checkout Service” |
| 0.9.7 | 30/03/2015 | Andrew Crawford | Made a change to the Password Recovery service to return a different flag when user needs to set a password AND agree to terms and conditions.  Added the ‘REDIRECT-URL’ to password recovery. Was mentioned in the login document, but forgotten from these screens.  Would rename this, but that may make things even more confusing. |
| 0.9.8 | 08/04/2015 | Andrew Crawford | Added the dateString to the countdown calendar service. |
| 0.9.9 | 28/04/2015 | Fernando Alves | Changed ‘type’ to ‘memberType’ for the verification service.  Put the ‘FORGOT USER’ back in on the forgotten service. Don’t think this version was distributed, so hopefully this doesn’t affect anyone.  Changed the Wyndham Verification Service to return the NOT FOUND as a 200 if successful to keep it in line with the other verification service. |

Contents

[1. Document Overview 6](#_Toc412042388)

[1.1 Related Documents 6](#_Toc412042389)

[2. Web Service Transport Details 7](#_Toc412042390)

[2.1 Format 7](#_Toc412042391)

[2.2 Security 7](#_Toc412042392)

[2.3 Mandatory Checks 8](#_Toc412042393)

[2.4 Text Values 8](#_Toc412042394)

[3. Privileges URLs 8](#_Toc412042395)

[4. Public Services 9](#_Toc412042396)

[3.1 Introduction 9](#_Toc412042397)

[3.2 Login Service 9](#_Toc412042398)

[3.3 Forgotten Service 10](#_Toc412042399)

[3.4 Wyndham Verification Service 11](#_Toc412042400)

[3.5 Verify Member Service 11](#_Toc412042401)

[3.6 Create Member Service 13](#_Toc412042402)

[3.7 Savings Calculator Service 16](#_Toc412042403)

[3.8 General Enquiry Service 17](#_Toc412042404)

[3.9 Message Service 17](#_Toc412042405)

[3.10 Country Service 18](#_Toc412042406)

[3.11 Privileges Purchase Amount Service 19](#_Toc412042407)

[3.12 Update Membership Receipt 19](#_Toc412042408)

[5. Member Services 20](#_Toc412042409)

[4.1 Introduction 20](#_Toc412042410)

[4.2 Logout Service 20](#_Toc412042411)

[4.3 Member Read Service 20](#_Toc412042412)

[4.4 Page Access Service 22](#_Toc412042413)

[4.5 Deal and Notification Service 23](#_Toc412042414)

[4.6 Update Member Service 24](#_Toc412042415)

[4.7 Update Password Service 25](#_Toc412042416)

[4.8 Request Card Service 25](#_Toc412042417)

[4.9 Card History Service 27](#_Toc412042418)

[4.10 Check Card Request History 28](#_Toc412042419)

[4.11 Update Membership Purchase Service 29](#_Toc412042420)

[4.12 Countdown Calendar Service 29](#_Toc412042421)

[4.13 Welcome Message Service 30](#_Toc412042422)

[4.14 Primary Banner Service 31](#_Toc412042423)

[4.15 Promotional Banner Service 31](#_Toc412042424)

[4.16 Frequent Values URL Service 32](#_Toc412042425)

[4.17 Page Tracking Service 33](#_Toc412042426)

[4.18 MyAccount Member Service 33](#_Toc412042427)

[4.19 Update Membership Receipt 35](#_Toc412042428)

[4.20 Member Benefits Service 36](#_Toc412042429)

[4.21 Wyndham Rewards Form 37](#_Toc412042430)

[5. Shop Services 37](#_Toc412042431)

[5.1 Introduction 37](#_Toc412042434)

[5.2 Categories Service 38](#_Toc412042435)

[5.3 Items Summary Service 38](#_Toc412042436)

[5.4 Items Summary by free text Service 40](#_Toc412042437)

[5.5 Item Service 40](#_Toc412042438)

[5.6 Check Delivery Area Service 42](#_Toc412042439)

[5.7 Initial Checkout Service 42](#_Toc412042440)

[5.8 Final Checkout Service 45](#_Toc412042441)

[5.9 Order Amount Service 46](#_Toc412042442)

[5.10 Order History Service 46](#_Toc412042443)

[5.11 Featured Product Service 47](#_Toc412042444)

[5. Holiday Deal Services 49](#_Toc412042445)

[5.1 Introduction 49](#_Toc412042448)

[5.2 Holidays Categories Service 49](#_Toc412042449)

[5.3 Holiday Summary Service 49](#_Toc412042450)

[5.4 Holiday Summary by free text Service 51](#_Toc412042451)

[5.5 Holiday Service 51](#_Toc412042452)

[5.6 Holiday Enquiry Service 53](#_Toc412042453)

[5.7 Minivac Eligbility Service 54](#_Toc412042454)

# Document Overview

This document is intended to provide a specification for the web services. This will detail the services available, the fields and types that these expect and the responses that will be returned.

## Related Documents

|  |  |  |
| --- | --- | --- |
| **Document Title** | **Location** | **Comments** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Web Service Transport Details

## Format

All web services will be in a JSON format. This was done because the services didn’t need the overhead or security from SOAP or similar transport standards. It also means that if need be the services can be called from JavaScript as well as back-end code.

All requests will look similar to:

*{*

*“username”:”123456”,*

*“password”:”654321”,*

*}*

Responses will also be in a JSON format but will also use a 200 error code to indicate a success and a 400 error code to indicate an error.

For example, a success message may look like:

***HTTP Code:*** *200*

***HTTP Body:*** *{*

*“sessionMemberID”:”1234567”*

*}*

Or even just a 200 response code with no body.

And an error message may look like:

***HTTP Code:*** *400*

***HTTP Body:*** *{*

*“error”:”Missing username”*

*}*

**Note:** The field names on the Request Values are case sensitive.

## Security

All services will be sent through HTTPS to comply with Wyndham’s standards on transporting personal information on public channels.

On the live server, requests will only be received from the public WCM IP Addresses to ensure a greater degree that requests originated from those servers.

After a member logs in, all services will be sent using a generated member session ID that will be sent back from the login service. This should ensure that in the event of a replay attack, requests will be rejected if the session expires.

A session is defined as being set 30 minutes between web service requests. If a user’s web service session times out, this will send back the text ‘EXPIRED’ as part of an error request.

As per Wyndham standards, these web services do not talk directly to the database from the DMZ. Instead they go through an intermediate step communicate to a server in the DMZ that then communicates to an internal server that communicates with the database.

## Mandatory Checks

It is expected that where a value is marked as Mandatory it will be checked for null and empty values before the web service is called.

The web service will repeat similar checks and return the text ‘MISSING $1’ where $1 is the field name that was empty.

## Text Values

In the description, this will show any text constants within a single bracket.

For example, if you see:

‘OWNER’ – Wyndham Owner Number

Then the text to pass for that value would be: OWNER

To make things simpler, these are also shown in upper case, but the service will accept any case for these values.

For the returned values, we will send back upper case text constants but it is up to your system if you want to cater for them being case sensitive or not.

# Privileges URLs

The main Privileges URLs are:

**Test / Stage:** <https://privileges-stage.wyndhamvrap.com/>

**Production:** <https://privileges.wyndahvrap.com/>

All URL’s references given will be relative to those base URLs.

For example, if the URL referenced is ‘/ws/login.json’ then the actual URL would be ‘https://privileges.wyndhamvrap.com/ws/login.json’

# Public Services

## Introduction

These are services are public in the sense that they do not require an internal member session ID to be called. This means that they can be called by the system before a member logs in.

## Login Service

**URL:** /ws/login.json

This is the main service used to log a user into the system.

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| Username | CHAR | Y | Username to log in with.  Username can be:   * 11 Digits – For Discovery / Owners * 6 Digits – For Staff * ‘WH’ + DIGITS – For Privileges White (Public) |
| Password | CHAR | Y | Password to log in with |

**Success Response**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Value** |
| Success | CHAR | This will contain either:   * ‘LOGIN’ if they were successfully authenticated * ‘REDIRECT’ if they are successful and need to be redirected to the change password page * ‘REDIRECT-TOC’ if they are successful and need to be redirected to the change password page and have the Terms and Conditions text and checkbox displayed. * ‘REDIRECT-DETAILS’ if they are successful and need to be redirected to the Member Details page (because we are missing some key information). |
| MemberNumber | CHAR | This will contain the session Member Number for use in other communications.  This uniquely identifies the member. |

**Error Response**

This returns a list of items.

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Value** |
| Error | CHAR | This will contain either:   * ‘MISSING $1’ where $1 is the missing mandatory field. * ‘NOT FOUND’ if the users credentials do not match our records. |

## Forgotten Service

This is used for the Forgotten Username / Forgotten Password Actions on the screen.

**URL:** /ws/forgotDetails.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| username | CHAR | Y – When type is ‘Password’  N – When type is ‘Username’ | Username to log in with.  Username can be:   * 11 Digits – For Discovery / Owners * 6 Digits – For Staff * Email Address (E.g. me@email.com) – Only for Public members * ‘WH’ + Digits – Only for Public Members |
| contact | CHAR | Y | Contact details (Phone, Mobile, Email) when asked.  This is asked on both a forgotten Username and Forgotten Password. |
| type | CHAR | Y | The type of information they have forgotten.  For now, values should only be:   * ‘PASSWORD’ – When triggered from the forgotten password screen. * ‘USERNAME’ – When triggered from the forgotten user screen |

**Success Response**

200 Response code with no body

**Error Response**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| Error | CHAR | This will contain either:   * ‘MISSING $1’ where $1 is the missing mandatory field. * ‘NOT FOUND’ if the details passed to not match any records. |

## Wyndham Verification Service

Used for returning Wyndham ownership information given some details. This is used primarily for verifying a Discovery Members details, but the plan is for this to also return Owners information if needed later on.

**URL:** /ws/ownerExists.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| memberNumber | INTEGER | Y | The membership number.  Membership number can be are just 11 Digits. |
| contact | CHAR | Y | Phone or Mobile Number.  Spaces and punctuation (i.e. Brackets, Dashes) should be removed from phone numbers before sending. |
| email | CHAR | Y | Email Address |
| memberType | CHAR | Y | This is to cater for different membership types.  Values are:   * ‘DISCOVERY’ – Wyndham Discovery Members * ‘OWNER’ – Wyndham Worldmark Owners |

**Success Response**

Returns a 200 code with no body if successful

**OR**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain ‘NOT FOUND’ if the users credentials do not match our records. |

**Error Response**

This returns a list of items.

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain either:   * ‘MISSING $1’ where $1 is the missing mandatory field. |

## Verify Member Service

This service is called as a pre-step to creating the membership.

This is used to verify that a privileges membership exists for the details passed.

**URL:** /ws/member/verifyMember.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| memberNumber | CHAR | Y – If type is ‘DISCOVERY’ ‘OWNER’ or ‘STAFF’ | This will contain either:   * 11 Digits – For Discovery / Owner * 6 Digits – For Staff * Email Address – For Public   This is set with the login that the person would use for the system.  As such, this will either be the owner number, Discovery Number, Staff Login or email address (for the Public) |
| memberType | CHAR | Y | This will contain either:   * ‘OWNER’ – Wyndham Worldmark Owners * ‘DISCOVERY’ – Wyndham Discovery Members * ‘STAFF’ – Wyndham Staff Members * ‘PUBLIC’ – For members of the Public |

**Success Response**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| success | CHAR | This returns the text:   * ‘EXISTS’ – If the membership details match an existing membership. * ‘NOTFOUND’ – If the membership details do not match an existing membership. |

**Error Response**

This returns a list of items.

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain either:   * ‘MISSING $1’ (where $1 is the missing value). |

## Create Member Service

Used for returning membership information given some details. This is used primarily for verifying a Discovery Members details, but the plan is for this to also return Owners information if needed later on.

For Discovery Members and Worldmark Owners this service just needs to pass through the Membership Details and Payment Information.

For staff members and members of the public, the service needs to pass back all information.

The reason for passing this back for staff members is they may have been issued with a WWID but may not have been created in our IT systems, so we need their full details to create an account on their behalf.

**URL:** /ws/member/createMember.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| memberNumber | CHAR | Y – If type is ‘DISCOVERY’ ‘OWNER’ or ‘STAFF’ | This will contain either:   * 11 Digits – For Discovery / Owner * 6 Digits – For Staff   This will be empty if the type is ‘PUBLIC’ |
| memberType | CHAR | Y | This will contain either:   * ‘OWNER’ – Wyndham Worldmark Owners * ‘DISCOVERY’ – Wyndham Discovery Members * ‘STAFF’ – Wyndham Staff Members * ‘PUBLIC’ – For members of the Public |
| password | CHAR | N – If the type is ‘OWNER’  Y – If the type is ‘DISCOVERY’ or ‘PUBLIC’ | Password to create the account with. |
| tocAgreeded | CHAR | Y | Contains the text ‘Y’  They are not permitted to create a membership through the public screens if they have not agreed to the Terms and Conditions |
| title | CHAR | Y – Only for ‘PUBLIC’ or ‘STAFF’  N – For ‘DISCOVERY’ or ‘OWNER’ | Persons title |
| firstName | CHAR | Y – Only for ‘PUBLIC’ or ‘STAFF’  N – For ‘DISCOVERY’ or ‘OWNER’ | Persons first name |
| lastName | CHAR | Y – Only for ‘PUBLIC’ or ‘STAFF’  N – For ‘DISCOVERY’ or ‘OWNER’ | Last Name or Surname |
| dob | CHAR | Y – Only for ‘PUBLIC’ or ‘STAFF’  N – For ‘DISCOVERY’ or ‘OWNER’ | Date of Birth in the format:  DD/MM/YYYY  E.g.  15/01/1990 |
| phone | CHAR | Y – Only for ‘PUBLIC’ or ‘STAFF’  N – For ‘DISCOVERY’ or ‘OWNER’  Either Phone or Mobile must be populated. | Home Phone Number |
| mobile | CHAR | Y – Only for ‘PUBLIC’ or ‘STAFF’  N – For ‘DISCOVERY’ or ‘OWNER’  Either Phone or Mobile must be populated. | Mobile Number |
| email | CHAR | Y – Only for ‘PUBLIC’ or ‘STAFF’  N – For ‘DISCOVERY’ or ‘OWNER’ | Email Address |
| addressLine1 | CHAR | Y – Only for ‘PUBLIC’ or ‘STAFF’  N – For ‘DISCOVERY’ or ‘OWNER’ | First Address Line |
| addressLine2 | CHAR | Y – Only for ‘PUBLIC’ or ‘STAFF’  N – For ‘DISCOVERY’ or ‘OWNER’ | Second Address Line |
| suburbCity | CHAR | Y – Only for ‘PUBLIC’ or ‘STAFF’  N – For ‘DISCOVERY’ or ‘OWNER’ | Suburb or City |
| state | CHAR | Y – Only for ‘PUBLIC’ or ‘STAFF’  N – For ‘DISCOVERY’ or ‘OWNER’ | Either an Australia State or alternate State / Provence / Locality. |
| postcode | CHAR | Y – Only for ‘PUBLIC’ or ‘STAFF’  N – For ‘DISCOVERY’ or ‘OWNER’ | Postcode or Zip Code |
| country | CHAR | Y – Only for ‘PUBLIC’ or ‘STAFF’  N – For ‘DISCOVERY’ or ‘OWNER’ | Country  This should correspond to the Country Key returned by the Country Service. |

**Success Response**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| success | CHAR | Session Member ID |

**Error Response**

This returns a list of items.

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain either:   * ‘MISSING $1’ where $1 is the missing mandatory field. * ‘EXISTS’ if the membership details match an existing member. * ‘NOTFOUND’ if the member type is owner or discovery and no matching records were found. * ‘NOTCREATED’ if the member type is owner or discovery, records were found, but the Privileges membership has not been created. |

## Savings Calculator Service

This is used for returning values needed by the Savings Calculator

**URL:** /ws/returnAllSavings.json

**Request Values**

There are no request values. The response will return everything when the URL is called.

**Success Response**

This returns a list of details

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| valueID | CHAR | A unique ID for the row |
| category | CHAR | Category to use |
| categoryOrder | INT | A value that can be used to order the categories.  The results will be ordered using this, but this is returned in case it is needed for the existing JavaScript. |
| question | CHAR | The question to ask |
| periodType | CHAR | The period the calculation applies for |
| points | CHAR | Appears to be used as the individual value a calculation is applied against. |
| limit | CHAR | Seems to be an upper limit applied to discounts. |

**Error Response**

**Note:** This still returns a list, but it will only contain 1 value with the error message in it

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This is catered for to keep up standards.  There is no expected error response, but in case there is anything unforeseen a custom message will be returned here.  We don’t expect this to be handled on the screens. This is being returned so we have some way to help debug what is failing in the service. |

## General Enquiry Service

Used for the General Enquiry form.

**URL:** /ws/contactUs.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| memberNumber | INTEGER | N | The membership number (if there is one entered) |
| enquiryRegarding | CHAR | Y | What the enquiry is about.  Expected values are:   * About Privileges * General Enquiry * Membership * Lost Membership Card * Forgot Username * Forgot Password * Request Additional Cards * Other   Based off a drop-down list on the Contact Us page. |
| firstName | CHAR | Y  (Firstname or Lastname are Mandatory) | Contact First Name  **Note:** They must enter either a firstname or a lastname. |
| lastName | CHAR | Y  (Firstname or Lastname are Mandatory) | Contact Last Name  **Note:** They must enter either a firstname or a lastname. |
| email | CHAR | Y  (Email or Phone are Mandatory) | Contact Email  **Note:** They must enter either a email or a phone. |
| Phone | CHAR | Y  (Email or Phone are Mandatory) | Contact Phone  **Note:** They must enter either a email or a phone. |
| text | CHAR | Y | The enquiry text |

**Success Response**

Returns a 200 code with no body

**Error Response**

This returns a list of items.

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain either:   * ‘MISSING $1’ where $1 is the missing mandatory field.   This returns a list since it is possible multiple mandatory fields are not set. |

## Message Service

Used for retrieving all messages back from the internal system.

**URL:** /ws/returnAllMessages.json

**Request Values**

Can be called with no parameters.

**Success Response**

(List of Values)

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| key | CHAR | Unique Message key (that can be referenced in the code) |
| title | CHAR | Message Title |
| text | CHAR | Message Text |

**Error Response**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | It is not expected that this will return an error, but if it does, this does not need to be handled / shown on the screen.  This returns an error more for debugging purposes. |

## Country Service

Used for retrieving all countries that can be selected for international addresses

**URL:** /ws/returnAllCountries.json

**Request Values**

Can be called with no parameters.

**Success Response**

(List of Values)

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| key | CHAR | Unique Country Key (E.g. AUS, IND) |
| name | CHAR | Display name for the country (E.g. Australia, India) |

**Error Response**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | It is not expected that this will return an error, but if it does, this does not need to be handled / shown on the screen.  This returns an error more for debugging purposes. |

## Privileges Purchase Amount Service

Used to return the current Privileges membership price.

**Note:** This service is only used by the Wyndham Payment Gateway. It uses this to get the amount to save the value being hard-coded within the Payment Gateway itself.

**URL:** /ws/privilegesAmount.json

**Request Values**

Can be called with no parameters.

**Success Response**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| amount | DECIMAL | The current cost of a Privileges Membership |
| reference | CHAR | Used to send back the payment reference information |

**Error Response**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | It is not expected that this will return an error, but if it does, this does not need to be handled / shown on the screen.  This returns an error more for debugging purposes. |

## Privileges Card Amount Service

Used to return the price of a membership card

**Note:** This service is only used by the Wyndham Payment Gateway. It uses this to get the amount to save the value being hard-coded within the Payment Gateway itself.

**URL:** /ws/privilegesCardAmount.json

**Request Values**

Can be called with no parameters.

**Success Response**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| amount | DECIMAL | The current cost of a Privileges Membership |
| reference | CHAR | Used to send back the payment reference information |

**Error Response**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | It is not expected that this will return an error, but if it does, this does not need to be handled / shown on the screen.  This returns an error more for debugging purposes. |

## Update Membership Receipt

Used to update a Privileges Membership Payment Receipt against a member record.

**Note:** This service is only used by the Wyndham Payment Gateway. This handles calling this service after a successful payment.

**URL:** /ws/payment/updateMemberReceipt.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | The sessionID |
| receiptNumber | CHAR | Y | The payment Receipt Number |
| receiptType | CHAR | Y | Set to ‘MEMBERSHIP’ |

**Success Response**

200 response with no body

**Error Response**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | It is not expected that this will return an error, but if it does, this does not need to be handled / shown on the screen.  This returns an error more for debugging purposes. |

## Update Card Receipt

Used to update a Privileges Membership Payment Receipt against a member record.

**Note:** This service is only used by the Wyndham Payment Gateway. This handles calling this service after a successful payment.

**URL:** /ws/payment/updateCardReceipt.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | The sessionID |
| receiptNumber | CHAR | Y | The payment Receipt Number |
| receiptType | CHAR | Y | Set to ‘CARD’ |

**Success Response**

200 response with no body

**Error Response**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | It is not expected that this will return an error, but if it does, this does not need to be handled / shown on the screen.  This returns an error more for debugging purposes. |

## Password Recovery Service

Used for the password recovery screen. Given a unique token, this will return if this is still valid (and return a valid SessionID) or if it’s expired because it was issued more than 72 hours ago.

**URL:** /ws/passwordRecovery.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| token | Char | Y | The unique password token passed from an email |

**Success Response Request Values**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| result | CHAR | This will return:  ‘EXPIRED’ – If the token is too old or not valid in the first place.  ‘REDIRECT’ – If the token is valid and the user can reset their password.  ‘REDIRECT-URL’ – If the token is valid but the user needs to redirect their password through a system external to Privileges.  ‘REDIRECT-PASS-TOC’ – If the token is valid and the user needs to not only reset their password but also to re-accept the Terms and Conditions. |
| url | CHAR | When the result is ‘REDIRECT-URL’, this is the URL to open. |
| SessionID | CHAR | If the result is ‘REDIRECT’ or ’REDIRECT-PASS-TOC’ |

**Error Response**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | Missing token – If the token cannot be found.  If the token has expired, it will send this back in the result field instead of using the error field. |

## Session Valid Service

Used to return if a session is still valid or not.

**URL:** /ws/sessionValid.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | Char | Y | The session ID to check. |

**Success Response Request Values**

Returns a 200 code with no body if successful

**OR**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will return:  ‘EXPIRED’ – If the session has expired or was not found. |

**Error Response**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | ‘Missing SessionID’ if the session ID is not passed in the body. |

# Member Services

## Introduction

These are services focused around changes that involve a member who is logged into the system.

## Logout Service

This is used to log a user and invalidate their web service session.

**URL:** /ws/member/logout.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | The Session Member ID |

**Success Response**

A 200 code with no body will be returned.

**Error Response**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain:   * ‘MISSING SessionID’. |

**Note:** This does not check if the SessionID has expired since the service would expire the session anyway.

## Member Read Service

Used to read a logged in members details.

This can be used to read either their full details (after they log in) or Partial details (if it’s just showing contact information).

**URL:** /ws/member/returnMemberDetails.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | The Session Member ID |
| type | CHAR | Y | Current values are:   * ‘FULL’ – Return all Details associated with a Membership. * ‘CONTACT’ – Return just the members contact details |

**Success Response**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| memberNumber | CHAR | Membership Number |
| memberType | CHAR | Membership Type.  Values are:   * ‘OWNER’ – Wyndham Worldmark Owner * ‘DISCOVERY’ – Wyndham Discovery Member * ‘STAFF’ – Wyndham Staff Member * ‘PUBLIC’ – Member of the Public |
| title | CHAR | The members title |
| firstName | CHAR | The members first name |
| partnerName | CHAR | The members partner name (if set) |
| dob | CHAR | The members date of birth.  This is a date returned as a string.  This will be in the format: DD/MM/YYYY |
| phone | CHAR | The members home phone number. |
| mobile | CHAR | The members mobile number. |
| email | CHAR | The members email address. |
| address\_Line1 | CHAR | The first address line |
| address\_Line2 | CHAR | The second address line |
| suburb\_City | CHAR | The members suburb or city |
| postcode | CHAR | The members postcode |
| country | CHAR | The members country  This will be the Country Name (as returned by the Country Service) |
| page\_Names | LIST  (Pages) | A list of page names the member has access to.  This will only be returned if the type was set to ‘FULL’ |
| push\_Notifications | LIST  (Notifications) | A list of push notifications to display for the member.  This will only be returned if the type was set to ‘FULL’ |

**Pages**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| page\_Name | CHAR | Constant name used to identify a page the member has access to. |

**Notifications**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| text | CHAR | Push Notifications to display for a member |

**Banners**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| image | CHAR | A path to the image banner to display. |
| url | CHAR | A url to open when the image banner is clicked. |

**Error Response**

This returns a list of items.

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain:   * ‘MISSING SessionID’. * ‘MISSING Type’ * ‘EXPIRED’ if the session ID is not found or has expired. |

## Page Access Service

Used for retrieving the page access details independently of the Member Read Service.

**URL:** /ws/member/pageAccess.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | N | The Session Member ID |

**Success Response**

This returns a list of values, each of which contain:

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| pageName | CHAR | Constant name used to identify a page the member has access to. |

**Error Response**

This returns a list of items.

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain:   * ‘MISSING SessionID’. * ‘EXPIRED’ if the session ID is not found or has expired. |

## Deal and Notification Service

Used for retrieving the push notification or Banner Details independently of the Member Read Service.

**URL:** /ws/member/returnNotification.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | N | The Session Member ID |
| type | CHAR | Y | The type of information to return.  Values are:   * ‘BANNER’ – To return current banners to display. * ‘NOTIFICATION’ – To return push notifications. * ‘HOTDEALS’ – Used for the Hot Deals page. This returns all deals whereas the Banner type only returns a sub-section. |

**Success Response**

This returns a list of values, each of which contain:

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| text | CHAR | Push notification text.  This is only returned if the type is ‘NOTIFICATION’ |
| image | CHAR | The banner image path to display.  This is only returned if the type is ‘BANNER’ or ‘HOTDEALS’. |
| url | CHAR | The URL to open for the image when clicked.  This is only returned if the type is ‘BANNER’ or ‘HOTDEALS’. |

**Error Response**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain:   * ‘MISSING SessionID’. * ‘MISSING Type’ * ‘EXPIRED’ if the session ID is not found or has expired. |

## Update Member Service

This is used to update the membership details of a logged in user.

**URL:** /ws/member/updateMember.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | Internal Member Session ID |
| title | CHAR | Y | Persons title |
| firstName | CHAR | Y | Persons first name |
| partnerName | CHAR | N | Partners or Spouse name |
| lastName | CHAR | Y | Last Name or Surname |
| dob | CHAR | Y | Date of Birth in the format:  DD/MM/YYYY  E.g.  15/01/1990 |
| phone | CHAR | Y  Either Phone or Mobile must be populated. | Home Phone Number |
| mobile | CHAR | Y  Either Phone or Mobile must be populated. | Mobile Number |
| email | CHAR | Y | Email Address |
| addressLine1 | CHAR | Y | First Address Line |
| addressLine2 | CHAR | Y | Second Address Line |
| suburbCity | CHAR | Y | Suburb or City |
| state | CHAR | Y | Either an Australia State or alternate State / Provence / Locality. |
| postcode | CHAR | Y | Postcode or Zip Code |
| country | CHAR | Y | Country  This should correspond to the Country Key returned by the Country Service. |

**Success Response**

A 200 code with no body will be returned.

**Error Response**

This returns a list of items.

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain:   * ‘MISSING $1’ where $1 is the missing mandatory field. * ‘EXPIRED’ if the session ID is not found or has expired. |

## Update Password Service

This is used to update the membership details of a logged in user.

This also records if the Terms and Conditions were agreed to (if they were displayed)

**Note:** Owners cannot update their password through Privileges. This is because their login relies on the US system. The should not have been able to see the change password area, and the service will reject these requests.

**URL:** /ws/member/updatePasswordToc.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | Internal Member Session ID |
| oldPassword | CHAR | N | Old Password |
| newPassword | CHAR | N | New Password  **Note:** When setting their resetting a password, just populate this field. |
| agreeTOC | CHAR | N | Records if they agreed to the Terms and Conditions displayed.  These only get displayed through a ‘REDIRECT-TOC’ request. |

**Success Response**

A 200 code with no body will be returned.

**Error Response**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain:   * ‘Missing $1’ where $1 is the missing mandatory field. * ‘EXPIRED’ if the session ID is not found or has expired. * ‘PASSWORD\_DOES\_NOT\_MATCH’ if the old password does not match the previously recorded password. * An error message if this is called on an Owner or Staff Member. * An error message a value of ‘N’ is passed to the agreeToc. You cannot use this service to unagree to the Terms and Conditions. |

## Request Card Service

Used for members requesting additional cards.

**URL:** /ws/member/requestCard.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | Internal Member Session ID |
| memberDetails | LIST  (MemberDetails) | Y | A list of the member details for additional cards.  Only owners and Discovery Members can send this through as a list.  Everyone else should send through their member name. |
| address\_Line1 | CHAR | Y | Address Line 1  Populated with either the members address or an alternate delivery address. |
| address\_Line2 | CHAR | Y | Address Line 2  Populated with either the members address or an alternate delivery address. |
| suburb\_City | CHAR | Y | Suburb or City.  Populated with either the members address or an alternate delivery address. |
| state | CHAR | Y | Either an Australia State or alternate State / Provence / Locality. |
| country | CHAR | Y | Country  This should correspond to the Country Key returned by the Country Service. |
| cardType | CHAR | Y | This is always set to: ‘MEMBERSHIP’ |

**MemberDetails**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| firstName | CHAR | Y  (When Supplied) | First Name on the additional card. |
| lastName | CHAR | Y  (When Supplied) | Last Name on the additional card. |

**Success Response**

A 200 code with no body will be returned.

**Error Response**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain:   * ‘Missing $1’ where $1 is the missing mandatory field. * ‘EXPIRED’ if the session ID is not found or has expired. |

## Card History Service

Returns the Card Request History

**URL:** /ws/member/cardHistory.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | Internal Member Session ID |

**Success Response**

This returns a list of values containing:

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| type | CHAR | Membership Card Type |
| name | CHAR | Membership Name requested on the card.  This will be first name and last name. |
| date | CHAR | The date the request was made.  This is a date in the format: DD/MM/YYYY |

**Error Response**

This returns a list of items.

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain:   * ‘MISSING $1’ where $1 is the missing mandatory field. * ‘EXPIRED’ if the session ID is not found or has expired. |

## Card Name Service

Returns all names a person can request cards for

**URL:** /ws/member/cardName.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | Internal Member Session ID |

**Success Response**

This returns a list of values containing:

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| type | CHAR | Membership Card Type |
| name | CHAR | Membership Name requested on the card.  This will be first name and last name. |
| date | CHAR | The date the request was made.  This is a date in the format: DD/MM/YYYY |

**Error Response**

This returns a list of items.

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain:   * ‘MISSING $1’ where $1 is the missing mandatory field. * ‘EXPIRED’ if the session ID is not found or has expired. |

## Check Card Request History

Returns if they member is eligible for a free card request or not

**URL:** /ws/member/cardRequestCheck.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | Internal Member Session ID |

**Success Response**

This returns a list of values containing:

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| status | CHAR | This returns if they can receive a free card request or if they have to purchase one.  This returns the values:   * ‘FREE’ – If they can receive a free card request * ‘PAID’ – If they must pay for a card request |

**Error Response**

This returns a list of items.

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain:   * ‘MISSING sessionID’ if the session is not set. * ‘EXPIRED’ if the session ID is not found or has expired. |

## Update Membership Purchase Service

Used by the Privileges Payment Gateway to update the membership record with the transaction ID from the Wyndham Payment Gateway

**Note:** This is only called from the Wyndham Payment Gateway

**URL:** /ws/payment/finalCheckout.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | Internal Member Session ID |
| transactionID | CHAR | Y | The Payment Gateway Transaction ID |

**Success Response**

A 200 code with no body will be returned.

**Error Response**

This returns a list of items.

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain:   * ‘MISSING $1’ where $1 is the missing mandatory field. * ‘EXPIRED’ if the session ID is not found or has expired. |

## Countdown Calendar Service

For some membership types, there is an expiry date that is set against them. This service is used to return the months and days left on the membership and the total days left (to support different visualisations of this data).

This service returns both a month + day group to display as well as a total day. This is done if there ever was a need to support different visualisations (Eg. Number vs Calendar)

For example, if there are 60 total days until the membership expires, this would return Month=2 Days=4 (28x2+4=60)

**URL:** /ws/member/countdownCalendar.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | Internal Member Session ID |

**Success Response**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| month | DECIMAL | How many Months until the membership expires (1 month = 28 days) |
| days | DECIMAL | How many days after the months until the membership expires |
| total\_days | DECIMAL | Total days until the membership expires |
| url | CHAR | URL to open when the user clicks on the Countdown Calendar. |
| dateStr | CHAR | The expiry date in the format mm/dd/yyyy |

**Error Response**

This returns a list of items.

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain:   * ‘MISSING sessionID’ if the sessionID is not set. * ‘EXPIRED’ if the session ID is not found or has expired. |

## Welcome Message Service

There was a requirement given for a custom welcome message to appear for the user.

This service was created with the necessary back-end rules to define what these messages should be and return them.

**URL:** /ws/member/welcomeMessage.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | Internal Member Session ID |

**Success Response**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| text | CHAR | The welcome message text to display based on the Member |

**Error Response**

This returns a list of items.

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain:   * ‘MISSING sessionID’ if the sessionID is not set. * ‘EXPIRED’ if the session ID is not found or has expired. |

## Primary Banner Service

There was a requirement given to show dynamic banner information on the main page. These banners have a URL associated with them that will open within the same window when clicked.

**URL:** /ws/returnPrimaryBanner.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | Internal Member Session ID |

**Success Response**

(This returns a list of items).

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| image | CHAR | The Image URL to use to show the image |
| url | CHAR | The URL to open when the image is clicked on |

**Error Response**

This returns a list of items.

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain:   * ‘MISSING sessionID’ if the sessionID is not set. * ‘EXPIRED’ if the session ID is not found or has expired. |

## Promotional Banner Service

There was a requirement given for 3 promotional banners showing on the primary page (after an owner logs in). This service was to be used for returning this information to the user.

**URL:** /ws/returnPromoBanner.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | Internal Member Session ID |

**Success Response**

(This returns a list of items).

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| image | CHAR | The Image URL to use to show the image |
| title | CHAR | The title to display |
| text | CHAR | The text to display |
| url | CHAR | The URL to open when the image is clicked on |

**Error Response**

This returns a list of items.

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain:   * ‘MISSING sessionID’ if the sessionID is not set. * ‘EXPIRED’ if the session ID is not found or has expired. |

## Frequent Values URL Service

There was a requirement given for 3 promotional banners showing on the primary page (after an owner logs in). This service was to be used for returning this information to the user.

**URL:** /ws/member/frequentValuesUrl.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | Internal Member Session ID |

**Success Response**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| url | CHAR | The Frequent Values URL to load for the member.  Note that these URL’s are unique to a Membership ID |

**Error Response**

This returns a list of items.

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain:   * ‘MISSING sessionID’ if the sessionID is not set. * ‘EXPIRED’ if the session ID is not found or has expired. |

## Page Tracking Service

This is used to track a Privileges page that the user has tried to open or a 3rd party page linked to from a Privileges page.

**URL:** /ws/member/recordPageVisit.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | Internal Member Session ID |
| url | CHAR | Y | The full URL that the user is opening |

**Success Response**

A 200 code with no body will be returned.

**Error Response**

This returns a list of items.

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain:   * ‘MISSING sessionID’ if the sessionID is not set. * ‘EXPIRED’ if the session ID is not found or has expired. |

## MyAccount Member Service

For the MyAccount page, we need to return the basic member details and additional details.

This web service was built to allow all details to be returned through a single call.

**URL:** /ws/member/myAccount.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | The Session Member ID |

**Success Response**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| memberNumber | CHAR | Membership Number |
| memberType | CHAR | Membership Type.  Values are:   * ‘OWNER’ – Wyndham Worldmark Owner * ‘DISCOVERY’ – Wyndham Discovery Member * ‘STAFF’ – Wyndham Staff Member * ‘PUBLIC’ – Member of the Public |
| privilegeTier | CHAR | Returns the Privileges Tier that is displayed. |
| benefitURL | CHAR | Privileges Tier Benefit URL to load in the Tier Benefits section |
| memberSince | CHAR | This is a date (in a string type) of when the Privileges membership started.  This will be in the format: DD/MM/YYYY |
| title | CHAR | The members title |
| firstName | CHAR | The members first name |
| partnerName | CHAR | The members partner name (if set) |
| dob | CHAR | The members date of birth.  This is a date returned as a string.  This will be in the format: DD/MM/YYYY |
| phone | CHAR | The members home phone number. |
| mobile | CHAR | The members mobile number. |
| email | CHAR | The members email address. |
| address\_Line1 | CHAR | The first address line |
| address\_Line2 | CHAR | The second address line |
| suburb\_City | CHAR | The members suburb or city |
| postcode | CHAR | The members postcode |
| country | CHAR | The members country  This will be the Country Name (as returned by the Country Service) |
| push\_Notifications | LIST  (Notifications) | A list of push notifications to display for the member.  This will only be returned if the type was set to ‘NOTIFICATION’  This information can be used for the Privileges News section. |
| membership | LIST  (Membership) | List of Membership numbers associated with the Privileges account |
| owner\_names | LIST  (Owner) | Owners names associated with the Privileges account.  This will only be populated if the member\_type is ‘OWNER’ or ‘DISCOVERY’ |

**Notifications**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| text | CHAR | Push Notifications to display for a member |

**Membership**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| name | CHAR | Membership Name (E.g. Privileges, Frequent Values, RCI, etc) |
| number | CHAR | Membership Number |
| Expiry | CHAR | Expiry Date in the format dd/mm/yyyy |

**Owner**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| title | CHAR | The Owners title |
| firstName | CHAR | Owners first name |
| lastName | CHAR | Owners last name |
| dob | CHAR | Date of birth in the format DD/MM/YYYY |

**Error Response**

This returns a list of items.

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain:   * ‘MISSING SessionID’. * ‘EXPIRED’ if the session ID is not found or has expired. |

## Member Benefits Service

It is used to return the Member Benefits against a member and category.

**Note:** The {categoryName} is a placeholder value for the category names.

These are:

* ACCOMODATION
* RESORT\_ BENEFITS
* TRAVEL\_RESERVATIONS
* MEMBERSHIPS
* DISCOUNTS
* WYNDHAM\_REWARDS

**URL:** /member/getBenefitCategory/**{categoryName}**.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | The Session Member ID |

**Success Response**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| categoryName | CHAR | The category name |
| type | CHAR | This will return either:   * ‘SINGLE’ – For the single column layout * ‘3COL’ – For the 3 column layout |
| benefits | List  (benefits) | List of benefits to show |

**Benefits:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| title | CHAR | The Benefit Title |
| image | CHAR | A path to the image to display |
| shortDesc | CHAR | The Short Description to display |
| longDesc | CHAR | The Long Description to display  **Note:** Only passed if the action is TEXT |
| url | CHAR | The URL to open when the item is clicked on  **Note:** Only passed if the action is URL |
| buttons | List  (imageButton) | A list of image buttons.  **Note:** There will be either 0, 1 or 2 items in this list |
| action | CHAR | Contains the text:   * ‘URL’ – If the action is to open a URL against the benefit * ‘TEXT’ – If the action is to show the long description against the benefit |
|  |  |  |

**imageButton:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| imageUrl | CHAR | The image path |
| linkUrl | CHAR | The URL to open when the image is clicked on |

**Error Response**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain:   * ‘MISSING $1’ where $1 is the missing value. * ‘EXPIRED’ if the session ID is not found or has expired. * ‘NOTFOUND’ if there are no benefits against the category (which would point to a larger configuration issue) |

## Wyndham Rewards Form

Used for the Wyndham Rewards Form

**URL:** /ws/rewardsForm.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | The Session Member ID |
| firstName | CHAR | Y | The first name from the form |
| lastName | CHAR | Y | The last name from the form |
| contactNumber | CHAR | Y | The contact number |
| email | CHAR | Y | The email |
| credits | INTEGER | Y | The amount of credits to exchange into points |

**Success Response**

Returns a 200 code with no body

**Error Response**

This returns a list of items.

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain either:   * ‘MISSING $1’ where $1 is the missing mandatory field.   This returns a list since it is possible multiple mandatory fields are not set. |

# Shop Services



## Introduction

These services are focused around interacting with the Shop and purchasing of items.

## Categories Service

Service to be used for retrieving categories and sub-categories without retrieving the shopping items associated to it. It should be used to build the category menu when loading the web site.

**URL:** /ws/shop/getCategories.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | Internal Member Session ID |
| country | CHAR | Y | Put in here for future support.  This should be set to the text ‘AUS’ |
| categoryId | CHAR | N | When passed, this will limit the menu items to level 1 only. The service will return that specific level 1 category and all the level 2 categories associated to it. |

**Success Response**

This returns a list of Categories:

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| id | Integer | Category ID |
| name | CHAR | Category Display Name |
| image | CHAR | The Path to the Image to display |
| subCategories | List (SubCategories) | List of sub-categories |

**Sub-category:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| id | Integer | Sub-category ID |
| name | CHAR | Sub-category Display Name |

## Items Summary Service

This is to be used for retrieving summaries for the shop items. As such, it would be used primarily when a user is navigating around the shop.

It filters the items by category or subcategory (One of them needs to be set) and post code.

**URL:** /ws/shop/getItemsSummary.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | Internal Member Session ID |
| categoryId | CHAR | N | When passed, this will limit items to a category passed.  It is not mandatory as long as subcategory is set. |
| subcategoryId | CHAR | N | When passed, this will limit items to a sub-category passed.  It is not mandatory as long as category is set. |
| country | CHAR | N | Put in here for future support.  This should be set to the text ‘AUS’ |
| postCode | CHAR | N | A postcode to limit items by (if entered). |

**Success Response**

This returns a list of shop items containing:

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| id | CHAR | Unique identifier for the item |
| name | CHAR | Item Name |
| supplier | CHAR | Item Supplier |
| brand | CHAR | Item Brand |
| description | CHAR | Item Short Description |
| price | DECIMAL | Item Sale Price |
| originalPrice | DECIMAL | Item Original Price |
| Images | LIST (Image) | A list of Images |

**Images:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| name | CHAR | Image name |
| location | CHAR | Location for that image |

**Error Response**

This returns a list of items.

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain:   * ‘MISSING $1’ where $1 is the missing mandatory field. * ‘NOTFOUND’ – If any of the other criteria were entered did not match against any records. * ‘EXPIRED’ if the session ID is not found or has expired. |

## Items Summary by free text Service

Used for searching items summary by free text.

**URL:** /ws/shop/getItemsSummaryByText.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | Internal Member Session ID |
| freeText | CHAR | Y | The free text to be used in the search |

**Success Response**

It is the same as “Item Summary Service”.

**Error Response**

It is the same as “Item Summary Service”.

## Item Service

Used for retrieving the full shop item details

**URL:** /ws/shop/getItem.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | Internal Member Session ID |
| itemID | CHAR | Y | The unique Shop Item ID |

**Success Response**

This returns a list of values containing:

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| category | Category | The item category |
| subCategory | SubCategory | The Item subcategory |
| id | CHAR | Unique identifier for the item |
| name | CHAR | Item Name |
| supplier | CHAR | Item Supplier |
| brand | CHAR | Item Brand |
| description | CHAR | Item Short Description – This is a general summary |
| longDescription | CHAR | The long or Full item – This is the full description and details of the item. |
| price | DECIMAL | Item Sale Price |
| originalPrice | DECIMAL | Item Original Price |
| maxQuantity | DECIMAL | The maximum number of this item that can be ordered. |
| shippingPrice | DECIMAL | The shipping price per item to charge. |
| options | List  (Option) | A list of options that can be selected for the item. |
| images | List  (Image) | A list of images to display with the item |
| disclaimer | CHAR | The disclaimer text to show with the Item |

**Category**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| Id | CHAR | Unique ID for the category row |
| name | CHAR | Category name |

**SubCategory**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| Id | CHAR | Unique ID for the sub-category row |
| name | CHAR | Sub-category name |

**Option**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| id | CHAR | Unique ID for the option row |
| name | CHAR | Grouping name for the option |
| value | CHAR | Value for the Name |

**Image**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| name | CHAR | Image name |
| location | CHAR | Location (Directory + File) relative to the root to show the image. |

**Error Response**

This returns a list of items.

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain:   * ‘MISSING $1’ where $1 is the missing mandatory field. * ‘NOTFOUND’ – If the ItemID does not match a known item * ‘EXPIRED’ if the session ID is not found or has expired. |

## Check Delivery Area Service

Used to check if an item is within the Delivery Area

**URL:** /ws/shop/checkDeliveryArea.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | Internal Member Session ID |
| itemID | CHAR | Y | The unique Shop Item ID |
| postcode | CHAR | Y | The Postcode to check |
| country | CHAR | Y | Added for future functions on country checking.  This should be set to ‘AUS’ |

**Success Response**

This returns a list of values containing:

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| inArea | CHAR | This returns:   * “true” – If the item can be delivered to that postcode * “false” – If the item cannot be delivered to that postcode. |
|  |  |  |

**Error Response**

This returns a list of items

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain:   * ‘MISSING $1’ where $1 is the missing mandatory field. * ‘NOTFOUND’ – If the ItemID does not match a known item. * ‘EXPIRED’ if the session ID is not found or has expired. |

## Initial Checkout Service

This service is intended to be called before the user goes to the NAB Payment gateway. It stores all the checkout details in our system before the gateway processes the order.

**URL:** /ws/shop/initialCheckout.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | Internal Member Session ID |
| order | Order | Y |  |

**Order**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| items | List (SelectedItem) | Y |  |
| address | Address | Y |  |

**SelectedItem**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| itemId | Integer | Y |  |
| quantity | Integer | Y |  |
| options | List  (Option) | N | Options for the items selected |

**Address**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| shippingName | CHAR | Y | The name to ship to.  This will either be populated from the member details or entered through the shipping address screen. |
| addressLine1 | CHAR | Y | The Address Line 1.  This will either be populated from the member details or entered through the shipping address screen. |
| addressLine2 | CHAR | Y | The Address Line 2.  This will either be populated from the member details or entered through the shipping address screen. |
| suburbCity | CHAR | Y | The Suburb or City.  This will either be populated from the member details or entered through the shipping address screen. |
| state | CHAR | Y | Either an Australia State or alternate State / Provence / Locality.  This will either be populated from the member details or entered through the shipping address screen |
| postcode | CHAR | Y | The postcode.  This will either be populated from the member details or entered through the shipping address screen |
| country | CHAR | Y | The country.  This will either be populated from the member details or entered through the shipping address screen. |
| dayContactNumber | CHAR | Y  Either Day\_Contact or Mobile is Mandatory. | The Daytime Contact Number.  This will either be populated from the Phone Number against the member details or entered through the shipping address screen. |
| mobile | CHAR | Y  Either Day\_Contact or Mobile is Mandatory. | The Mobile Contact Number.  This will either be populated from the Phone Number against the member details or entered through the shipping address screen. |

**Option**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| name | CHAR | Y | Option Name |
| value | CHAR | Y | Option Value |

**Note:** Adding options are not mandatory, but if there are added, both the option name and value needs to be set.

**Success Response**

This returns a list of values containing:

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| orderSummaryID | CHAR | A unique identifier for the order details submitted. |

**Error Response**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain:   * ‘MISSING $1’ where $1 is the missing mandatory field. * ‘NOTFOUND’ indicating some ItemID for an item that was not found in our system. Check notFoundItemsId field for a list of ids that were not found. * ‘OUT\_OF\_AREA’ indicating some item cannot be delivered to the requested post code. Check outOfAreaItemsId field for a list of ids that cannot be delivered. * ‘EXPIRED’ if the session ID is not found or has expired. |
| outOfAreaItemsId | List of item ID  (Integer) | A list of any item IDs that were out of area of the shipping postcode. |
| notFoundItemsId | List of item ID  (Integer) | A list of any item IDs that were out of area of the shipping postcode. |

## Final Checkout Service

This service is intended to be called after the user goes to the NAB Payment gateway. It stores all the checkout details in our system including the receipt returned by the Payment gateway.

**Note:** This service is called from the Wyndham Payment Gateway as part of the Payment Processing.

**URL:** /ws/order/finalCheckout.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | Internal Member Session ID |
| orderSummaryID | INTEGER | Y | Order Summary ID |
| paymentReceipt | CHAR | Y | The payment receipt that comes back from the NAB Payment gateway. |

**Success Response**

200 Response code with no body

**Error Response**

This returns a list of items

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| Error | CHAR | This will contain:   * ‘MISSING $1’ where $1 is the missing mandatory field. * ‘NOTFOUND’ if the OrderSummaryID does not match an order record. * ‘EXPIRED’ if the session ID is not found or has expired. |

## Order Amount Service

This service is used to retrieve the amount to pay given the payment information.

**Note:** This service is called from the Wyndham Payment Gateway as part of the Payment Processing.

**URL:** /ws/payment/getOrderAmount.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | Internal Member Session ID |
| orderSummaryID | CHAR | Y | Order Summary ID |

**Success Response**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| Amount | DECIMAL | The total amount to be paid against the summary |
| Reference | CHAR | A text reference that can appear with the payment |

**Error Response**

This returns a list of items

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| Error | CHAR | This will contain:   * ‘MISSING $1’ where $1 is the missing mandatory field. * ‘NOTFOUND’ if the OrderSummaryID does not match an order record. * ‘EXPIRED’ if the session ID is not found or has expired. |

## Order History Service

This service is used to retrieve order history for the member.

**URL:** /ws/order/getOrderHistory.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | Internal Member Session ID |

**Success Response**

(Returns a list of items)

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| summaryID | CHAR | Unique Order Summary ID |
| date | CHAR | Date the Order was made.  This is in the format: dd/mm/yyyy |
| total | DECIMAL | Total amount of the order |
| items | ITEM  (LIST) | List of Order Items |

**Item**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| name | CHAR | The Item Name |
| brand | CHAR | The Brand of the Item |
| amount | DECIMAL | The Purchase Amount of the Item |
| quantity | DECMIAL | How many of the item they purchased |

**Note:** If there are no orders placed for the member, this returns a 200 success code with no body

**Error Response**

This returns a list of items

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| Error | CHAR | This will contain:   * ‘MISSING $1’ where $1 is the missing mandatory field. * ‘EXPIRED’ if the session ID is not found or has expired. |

## Featured Product Service

Used for returning Featured / Top Items from the Shop.

**Note:** We may not need top items, but the service will cater for these for now.

It filters the items by category or subcategory (One of them needs to be set) and post code.

**URL:** /ws/shop/getFeaturedItemsSummary.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | Internal Member Session ID |
| categoryId | CHAR | N | When passed, this will limit items to a category passed.  It is not mandatory as long as subcategory is set.  This is passed a category Id. |
| subcategoryId | CHAR | N | When passed, this will limit items to a sub-category passed.  It is not mandatory as long as category is set.  This is passed a Sub Category Id. |
| country | CHAR | N | Put in here for future support.  This should be set to the text ‘AUS’ |
| postCode | CHAR | N | A postcode to limit items by (if entered). |
| type | CHAR | Y | This is passed the text:   * ‘FEATURED’ for featured items * ‘TOP’ for top items. |

**Success Response**

This returns a list of shop items containing:

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| id | CHAR | Unique identifier for the item |
| name | CHAR | Item Name |
| supplier | CHAR | Item Supplier |
| brand | CHAR | Item Brand |
| description | CHAR | Item Short Description |
| price | DECIMAL | Item Sale Price |
| originalPrice | DECIMAL | Item Original Price |
| Images | LIST (Image) | A list of Images |

**Images:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| name | CHAR | Image name |
| location | CHAR | Location for that image |

**Error Response**

This returns a list of items.

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain:   * ‘MISSING $1’ where $1 is the missing mandatory field. * ‘NOTFOUND’ – If any of the other criteria were entered did not match against any records. * ‘EXPIRED’ if the session ID is not found or has expired. |

# Holiday Deal Services



## Introduction

These are services related to the Holiday Deals and Resort Bookings.

## Holidays Categories Service

Service to be used for retrieving holiday’s categories and sub-categories without retrieving the holiday deals associated to it. It should be used to build the category menu when loading the web site.

**URL:** /ws/holiday/getCategories.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | Internal Member Session ID |
| country | CHAR | Y | Put in here for future support.  This should be set to the text ‘AUS’ |
| categoryId | CHAR | N | When passed, this will limit the menu items to level 1 only. The service will return that specific level 1 category and all the level 2 categories associated to it. |

**Success Response**

This returns a list of Categories:

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| id | Integer | Category ID |
| name | CHAR | Category Display Name |
| subCategories | List (SubCategories) | List of sub-categories |

**Sub-category:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| id | Integer | Sub-category ID |
| name | CHAR | Sub-category Display Name |

## Holiday Summary Service

This is to be used for retrieving summaries for the holiday deals. As such, it would be used primarily when a user is navigating around the menus.

It filters the items by category or subcategory.

**URL:** /ws/holiday/getSummary.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | Internal Member Session ID |
| categoryId | CHAR | N | When passed, this will limit items to a category passed.  It is not mandatory as long as subcategory is set. |
| subcategory | CHAR | N | When passed, this will limit items to a sub-category passed.  It is not mandatory as long as category is set. |
| country | CHAR | Y | Put in here for future support.  This should be set to the text ‘AUS’ |

**Success Response**

This returns a list of shop items containing:

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| id | CHAR | Unique identifier for the item |
| name | CHAR | Item Name |
| description | CHAR | Item Short Description |
| price | CHAR | Deal sales Price or deal discount.  This is a CHARACTER since it seems that we need to support showing the price as a $ amount (Eg. AUD $99) as well as a % off (E.g. 10% off for members). |
| originalPrice | DECIMAL | Item Original Price |
| Images | LIST (Image) | List (containing a single value) |

**Images:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| name | CHAR | Image name |
| location | CHAR | Location for that image |

**Error Response**

This returns a list of items.

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain:   * ‘MISSING $1’ where $1 is the missing mandatory field. * ‘NOTFOUND’ – If any of the other criteria were entered did not match against any records. * ‘EXPIRED’ if the session ID is not found or has expired. |

## Holiday Summary by free text Service

Used for searching holiday deal summaries by free text.

**URL:** /ws/holiday/getSummaryByText.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | Internal Member Session ID |
| freeText | CHAR | Y | The free text to be used in the search |

**Success Response**

It is the same as “Holiday Summary Service”.

**Error Response**

It is the same as “Holiday Summary Service”.

## Holiday Service

Used for retrieving the full shop item details

**URL:** /ws/holiday/getHoliday.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | Internal Member Session ID |
| itemID | CHAR | Y | The unique deal Item ID |

**Success Response**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| category | Category | The item category |
| subCategory | SubCategory | The Item subcategory |
| id | CHAR | Unique identifier for the deal |
| name | CHAR | Deal Name |
| subName |  | Name to show as a sub heading under the deal name |
| description | CHAR | Deal Short Description |
| longDescription | CHAR | Deal Long / Full Description |
| price | CHAR | Deal sales Price or deal discount.  This is a CHARACTER since it seems that we need to support showing the price as a $ amount (Eg. AUD $99) as well as a % off (E.g. 10% off for members). |
| conditions | CHAR | Shows the Price conditions.  E. g. Price Per Adult Twin Sharing Basis |
| images | List  (Image) | List containing just one image to show on the summary pages |
| bookingType | CHAR | The Booking Type to drive the booking actions.  Expected types are:   * ‘BOOKNOW’ – For 3rd party booking services * ‘MINIVAC’ – For Mini Vacation Deals |
| bookingURL | CHAR | When the type is ‘BOOKNOW’ this returns a URL for 3rd party booking sites. |
| amenities | CHAR | Contains text information to show about the amenities |
| rooms | CHAR | Contains text information to show about the rooms. |
| disclaimer | CHAR | Holds disclaimer text to show with the deal |
| view | CHAR | This contains the actions.   * SHOW – For when the booking button is shown. * PURCHASED – For when the booking button is greyed out. |
| alernateID | CHAR | The alternate booking ID when the bookingType is MiniVac |

**Category**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| Id | CHAR | Unique ID for the category row |
| name | CHAR | Category name |

**SubCategory**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| Id | CHAR | Unique ID for the sub-category row |
| name | CHAR | Sub-category name |

**Image**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| imageID | CHAR | Unique ID for the image row |
| imageLocation | CHAR | Location (Directory + File) relative to the root to show the image. |

**Error Response**

This returns a list of items

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain:   * ‘MISSING $1’ where $1 is the missing mandatory field. * ‘NOTFOUND’ – If the ItemID does not match a known item * ‘EXPIRED’ if the session ID is not found or has expired. |

## Holiday Enquiry Service

If the booking type is ‘ENQUIRENOW’ then an internal contact form is shown to the users to get additional information about the deal.

This service is intended to capture the values entered by the user.

**URL:** /ws/holiday/submitHolidayEnquiry.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | Internal Member Session ID |
| itemID | CHAR | Y | The ItemID for the deal that is being enquired about. |
| name | CHAR | Y | Name entered.  This is either populated from the Members details or overwritten on the Contact form. |
| contact | CHAR | Y | The contact details |
| date\_from | CHAR | N | The enquiry from date.  This is a date as a string in the format: DD/MM/YYYY |
| date\_to | CHAR | N | The enquiry to date.  This is a date as a string in the format: DD/MM/YYYY |
| noOfPeople | CHAR | N | The number of people to include for the deal. |
| text | CHAR | N | Any additional text with the enquiry. |
| qualification\_Answers | CHAR | N | If the bookingType is ‘MINIVAC’ then this is populated with the qualification questions and answers as comma separated values.  For example:  income=$75,000 pa and over, age=55 - 72 |

**Success Response**

Returns a 200 code with no body

**Error Response**

This returns a list of items

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain either:   * ‘MISSING $1’ where $1 is the missing mandatory field. * ‘NOTFOUND’ – If the ItemID does not match a known deal * ‘EXPIRED’ if the session ID is not found or has expired.   This returns a list since it is possible multiple mandatory fields are not set. |

## Minivac Eligbility Service

This service is used to determine if someone is eligible or not eligible for a MiniVac deal and to also generate the initial Not Eligible email for the user.

**URL:** /ws/holiday/minivacEligible.json

**Request Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value Type** | **Mandatory** | **Description** |
| sessionID | CHAR | Y | Internal Member Session ID |
| itemID | CHAR | Y | The Holiday Deal Item ID |
| marital | CHAR | Y | The Marital status.  This contains the values:   * Single * Married * Separated * Partner |
| income | CHAR | Y | The household income.  This contains the values:   * $0 - $54,999 * $55K - $64,999 * $65K - $74,999 * $75K - $99,999 * $100K+ |
| age | CHAR | Y | The age group.  This contains the values:   * 18 - 24 years * 25 - 39 years * 40 - 54 years * 55 - 72 years * 73 and over |

**Success Response**

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| Status | CHAR | The eligibility status.  This contains the values:   * ‘INELIGIBLE’ – If they do not qualify for a MiniVac * ‘ELIGIBLE’ – If they do qualify for a MiniVac |

**Error Response**

This returns a list of items

|  |  |  |
| --- | --- | --- |
| **Name** | **Value Type** | **Description** |
| error | CHAR | This will contain either:   * ‘MISSING $1’ where $1 is the missing mandatory field. * ‘NOTFOUND’ – If the ItemID does not match a known deal * ‘EXPIRED’ if the session ID is not found or has expired.   This returns a list since it is possible multiple mandatory fields are not set. |