



Agoda YCS4 XML API

Version 4.0

Version Control

Version	Date	Description
0.1	23 August 2011	Initial Draft by Wayne
0.1.1	7 th October 2011	Final Draft by Wayne
0.2	11 th January 2012	Updated for new format XML
0.3	27 th March 2012	Updated for QA & UAT Feedback
0.4	27 th April 2012	Updated as per UAT review
0.5	7 th June 2012	Revised version
0.51	4 th July 2012	Add GetAllLanguages and change from Storefront ID to Language ID
0.6	26 th July 2012	Complete Promotion, Rate Matrix and Booking
0.7	17 th August 2012	XML tables examples added
0.8	22 nd August 2012	Definitions and Schemas Added
0.81	27 th August 2012	Rate Matrix requests added to Full version
0.90	3 rd September 2012	Added Full Descriptions for Get Requests Removed AutoTopUp from SetHotelInventory request

Table of Contents

AGODA COMPANY	5
COMPANY BACKGROUND	5
CONTACT INFORMATION	5
OVERVIEW	6
IMPORTANT NOTES	6
AGODA BEST PRACTICES	6
AGODA RECOMMENDATIONS	6
AGODA FAIR USE POLICY	7
GENERIC XML ERROR HANDLING	7
RESPONSE CODES	7
RESPONSE BODY	7
XML GET OPERATIONS	9
GETALLCXLPOICIES	9
GETHOTELCXLPOLICY	11
GETALLCURRENCIES	14
GETALLLANGUAGES	16
GETHOTELPROMOTIONS	18
GETHOTELROOMTYPES	19
GETHOTELROOMSETTINGS	21
GETHOTELRATEPLANS	23
GETHOTELINVENTORY	26
GETALLSURCHARGETYPES	31
GETHOTELSURCHARGE	33
GETALLVALUEADDS	35
GETHOTELVALUEADDS	37
XML SET OPERATIONS	39
SETHOTELINVENTORY	39
SETHOTELCXLPOLICY	44
SETHOTELPROMOTION	46
SETHOTELROOMSETTINGS	47
SETHOTELSURCHARGE	49
SETHOTELVALUEADDS	51
BOOKING API'S	53
GETBOOKINGLIST	53
GETBOOKINGDETAILS	55
APPENDIX A	59
YCS MAIN SCHEMA	59
YCS XML SCHEMAS	70

APPENDIX C	80
CHARGE TYPE LIST	80
APPENDIX D	81
ERROR CODES	81
APPENDIX E	83
LIST OF COUNTRY NAMES, CODES AND CURRENCIES	83

Agoda Company

Company Background

With hundreds of staff operating across the globe, Agoda is uniquely able to combine local knowledge and connections with world class technology to provide the best hotel deals on the web.

What makes Agoda unique is the pairing of technology and the human touch. We negotiate personally and directly with hotels to ensure the best possible value, and then use the latest technology to deliver these bargains to our customers. And our 24/7 multi-lingual customer service means assistance is quick at hand when it really counts.

Thousands of hotels have chosen Agoda to provide distribution and sales for their hotel because we are always accountable, reachable and connected. In recognition of this, Agoda was voted Favorite Online Travel Site at the TravelMole and EyeForTravel Online Travel Innovation Awards 2012.

At Agoda, we believe that travel opens minds to new ideas, cultures and ways of thinking. That is why we strive to make travel more accessible to everyone by making it fast, easy and affordable to find great deals on hotels worldwide. We sell globally but price locally, leaving you more time and money to enjoy the journey.

Contact Information

All support enquiries relating to Agoda YCS4 XML API should be sent to bd-connectivity@agoda.com to ensure they are handled efficiently by the Agoda Connectivity team.

For commercial enquiries please contact Andrew Smith - Connectivity Manager of Hotels Global for Agoda.

Contact Details

Support

bd-connectivity@agoda.com

Commercial

Andrew Smith
Connectivity Manager – Hotels, Global
Phone: +66 2 625 9278
Email: andrew.smith@agoda.com

Overview

The purpose of this document is to provide Agoda Yield Control System (YCS) partners with an XML based API. This API is only accessible to approved partners, and the data that is accessible via the API is limited to directly contracted Agoda hotels.

Important Notes

Backwards Compatibility

Agoda will support YCS API version 2.2 and 3.0 until **30 June 2013**. Partners are encouraged to develop to YCS API version 4 ahead of this deadline to take advantage of the new features and functional improvements in YCS4.

Agoda Best Practices

Partners should adhere to the following common standards/guidelines:

1. Partner servers should use server names and DNS to resolve server names into IP addresses, and should implement DNS "Time To Live". Partners should not connect to IP addresses nor cache IP addresses longer than the Agoda set Time To Live.
2. Partner should properly implement parsing of the XML that is received from Agoda. For example, Partner should not make assumptions on the order of elements, presence or absence of certain elements, other than what is provided by Agoda in the XML Schema.
3. Partners must read and action XML response messages and response codes. Should an error occur the response code and description will provide information as to the cause of the error, be it violation of the business rules, violation of our 'Fair Use Policy' (detailed below) or an error on Agoda's servers. It is Partners responsibility to correct any security or data related errors, for instance Failure to authenticate due to invalid API key, incorrect Room Type ID's etc.

Agoda Recommendations

Coming Soon

Agoda Fair Use Policy

Agoda Services Co Ltd reserves the right to temporarily suspend the service to partners who post an unreasonable number of requests per second. Should the service be suspended, Agoda will return the documented response code of 403. This policy is in place to ensure all of our partners can use the system without causing unnecessary disruption to other Agoda partners.

Generic XML Error Handling

Agoda YCS4 API has exception handling and communication about problems in the API validation or problems in our processing.

Response Codes

Every response XML root element or response object will have a "status" field as an attribute that identifies the general response type from Agoda's API servers.

Please see [Appendix D](#) for a full list of Error Codes

It is very important that YCS4 API partners review these error responses and action on their information. Failure to action error responses can cause significant operational and financial impact on Agoda and API partner Hotel customers. Agoda will not accept liability for any financial impact to API customers if error messages are returned to API partners and not actioned on.

Response Body

Success Responses:

In case you received a success response status, the response body will, if applicable for the request, contain the response object as defined in the conversation specification.

Example:

```
<StatusResponse status="200" AffectedRows="1"/>
```

Error Responses:

In case you receive an error response, the response body will contain an <StatusResponse> tag with a collection of <ErrorItem> elements that identify as best as we can the exact problems or validation problems that we have encountered processing your request.

Example:

```
<StatusResponse Status="200" AffectedRows="1">
  <ErrorItem ErrorID="201" ItemID="1" ErrorText="This is an example error item that
  can be included in the ProcessResult"/>
  <ErrorItem ErrorID="201" ItemID="2" ErrorText="This is an example error item that
  can be included in the ProcessResult"/>
</StatusResponse>
```

```
<StatusResponse Status="401" AffectedRows="0">
  <ErrorItem ErrorID="401" ItemID="0" ErrorText="Authentication failed. Invalid API
  Key." />
</StatusResponse>
```

```
<StatusResponse Status="500" AffectedRows="0">
  <ErrorItem ErrorID="157" ItemID="1" ErrorText="Invalid Surcharge Rate - [2012-06-
  07 - 2012-07-07] - CurrencyCode: Cannot set currency code for this surcharge
  type." />
  <ErrorItem ErrorID="157" ItemID="1" ErrorText="Invalid Surcharge Rate - [2012-06-
  07 - 2012-07-07] - SurchargeType: Surcharge type is not matched." />
  <ErrorItem ErrorID="157" ItemID="2" ErrorText="Invalid Surcharge Rate - [2012-06-
  07 - 2012-07-07] - SurchargeType: Surcharge type is not matched." />
</StatusResponse>
```


XML Get Operations

Agoda 'get' messages are used to request information from Agoda API and receive a response with this information contained. Get messages should be viewed as information only messages and posting these messages to Agoda API will result in no change to data stored within YCS4.

GetAllCxlPolicies

Description

This message type allows partner to request all cancellation policies that are available to hotels using YCS4. This message response will enable partner to then use contents to post [SetHotelCxlPolicy](#) to set new cancellation policy for property.



Helpful Hint: This message request must contain HotelID element for authentication however this does not mean that the response is specific to this property. This message type returns *all* Agoda cancellation policies regardless of hotel in request



Helpful Hint: This message request must contain RequestedLanguage element. The Policy Text returned will be in the language specified and a list of supported languages can be found using [GetAllLanguages](#) message

XML Elements and Attributes

GetAllCxlPolicies RQ					
Level	Name	Type	Occurs	Required	Comments
0	GetAllCxlPoliciesRequest		1	Y	
1	Authentication				
	@APIKey	String Length 32	1	Y	This is the primary authentication for connection to the Agoda API
	@HotelID	String Length 1-8	1	Y	Every request must contain a valid hotel ID that partner has access to
1	RequestedLanguage	Alpha Length 2	1	N	All Agoda languages can be found by requesting GetAllLanguages

GetAllCxlPolicies RS					
Level	Name	Type	Occurs	Required	Comments
0	GetAllCxlPoliciesResponse		1	Y	
1	StatusResponse		1	Y	For further information on status responses see Response Codes and for code reference see Appendix D
	@Status	Numeric Length 3	1	Y	
	@AffectedRows	Numeric Length 1 - 4	1	Y	Affected rows indicates how many rows of data are returned (get request) or updated (set request)
1	CancellationPolicyList		1	Y	Contains each cancellation policy
2	CancellationPolicy		1-n	Y	
3	CancellationPolicyID	Numeric Length 4-6	1	Y	Each Agoda cancellation policy has a unique ID. This ID must be used when setting hotel cancel policy

					using SetHotelCxlPolicy
3	PolicyParameters		1	Y	Encloses three policy parameters for each cancellation policy
4	Policies	Numeric NNN.NNNN	3	Y	All cancellation policies contain 3 policy elements. Policies are set in order and policy 3 is always no show policy and will list days as 0. The Policies value relates to a value or percentage
	@days	Numeric Length 1-3	1	Y	
	@charge	Alpha Length 1	1	Y	Will show either N for number of nights penalty or P for percentage of total cost penalty
3	PolicyText	CDATA	1	Y	Plain text policy description in language requested
	@language	Alpha Length 2	1	Y	

Request Example

```
<?xml version="1.0" encoding="UTF-8"?>
<GetAllCxlPoliciesRequest xmlns="http://xml.ycs.agoda.com">
  <Authentication APIKey="00000000-0000-0000-0000-000000000000" HotelID="00000"/>
  <RequestedLanguage>en</RequestedLanguage>
</GetAllCxlPoliciesRequest>
```

Response Example

```
<?xml version="1.0"?>
<GetAllCxlPoliciesResponse xmlns="http://xml.ycs.agoda.com">
  <StatusResponse status="200" AffectedRows="4" />
  <CancellationPolicyList>
    <CancellationPolicy>
      <CancellationPolicyID>13308</CancellationPolicyID>
      <PolicyParameters>
        <Policies days="365" charge="P">100.0000</Policies>
        <Policies days="0" charge="P">0.0000</Policies>
        <Policies days="0" charge="P">100.0000</Policies>
      </PolicyParameters>
      <PolicyText language="en"><![CDATA[This booking is non-Refundable and cannot be amended or modified. Failure to arrive at your hotel will be treated as a No-Show and no refund will be given. (Hotel policy)]]></PolicyText>
    </CancellationPolicy>
  </CancellationPolicyList>
</GetAllCxlPoliciesResponse>
```

GetHotelCxlPolicy

Description

This message type allows partner to request all cancellation policies related to a single hotel using YCS4.



Important Note: To retrieve hotel level cancellation policies, ensure that room type element is entered as 0 (zero). If you want to check for specific room type cancellation policies, then submit valid room type ID.

This message type is used to retrieve a hotel cancellation policy specific to:

- Rate Plan
- Room Type
- Dates Range

XML Elements and Attributes

GetHotelCxlPolicy RQ					
Level	Name	Type	Occurs	Required	Comments
0	GetHotelCxlPolicyRequest		1	Y	
1	Authentication				
	@APIKey	String Length 32	1	Y	This is the primary authentication for connection to the Agoda API
	@HotelID	String Length 1-8	1	Y	Every request must contain a valid hotel ID that partner has access to
1	RoomType		1	Y	
	@RoomTypeID	Numeric Length 1-8	1	Y	This is the Agoda room type ID that must be submitted to request Hotel Cancellation Policy. Use GetHotelRoomTypes to get IDs. To get hotel level cancellation policies set RoomType to 0
	@RatePlanID	Numeric Length 1-3	1	Y	This is the Agoda rate plan ID that must be submitted to request availability, rate and inventory information (ARI). Use GetHotelRatePlans to get IDs
1	DateRange		1	Y	Date Range information for request
	@Type	Alpha	1	Y	Type of date range. There is only one option currently which is Stay
	@Start	Date YYYY-MM-DD	1	Y	The inclusive start date for the query
	@End	Date YYYY-MM-DD	1	Y	The inclusive end date for the query
1	RequestedLanguage	Alpha Length 2	1	N	All Agoda languages can be found by requesting GetAllLanguages

GetHotelCxlPolicy RS					
Level	Name	Type	Occurs	Required	Comments
0	GetHotelCxlPolicyResponse		1	Y	
1	StatusResponse		1	Y	For further information on status responses see Response Codes and

					for code reference see Appendix D
	@Status	Numeric Length 3	1	Y	
	@AffectedRows	Numeric Length 1-4	1	Y	Affected rows indicates how many rows of data are returned (get request) or updated (set request)
1	HotelCancellationPolicyList		1	Y	Contains each cancellation policy
2	HotelCancellationPolicy		1-n	Y	
3	RoomType		1	Y	
	@RoomTypeID	Numeric Length 1-8	1	Y	This is the Agoda room type ID related to cancellation policy. Use GetHotelRoomTypes to get IDs
	@RatePlanID	Numeric Length 1-3	1	Y	This is the Agoda rate plan ID related to cancellation policy. Use GetHotelRatePlans to get IDs
3	DateRange		1	Y	Date Range information for request
	@Type	Alpha	1	Y	Type of date range. There is only one option currently which is Stay
	@Start	Date YYYY-MM-DD	1	Y	The inclusive start date for the response
	@End	Date YYYY-MM-DD	1	Y	The inclusive end date for the response
3	CancellationPolicy		1-n	Y	
3	CancellationPolicyID	Numeric Length 4-6	1	Y	Each Agoda hotel cancellation policy has a unique ID. This ID relates to this hotel cancel policy only. Use SetHotelCxlPolicy to set hotel policy
3	PolicyParameters		1	Y	Encloses three policy parameters for each cancellation policy
4	Policies	Numeric NNN.NNNN	3	Y	All cancellation policies contain 3 policy elements. Policies are set in order and policy 3 is always no show policy and will list days as 0 . The Policies value relates to a value or percentage
	@days	Numeric Length 1-3	1	Y	
	@charge	Alpha Length 1	1	Y	Will show either N for number of nights penalty or P for percentage of total cost penalty
3	PolicyText	CDATA	1	Y	Plain text policy description in language requested
	@language	Alpha Length 2	1	Y	

Request Example

```
<?xml version="1.0" encoding="utf-8" ?>
<GetHotelCxlPolicyRequest xmlns="http://xml.ycs.agoda.com">
  <Authentication APIKey="00000000-0000-0000-0000-000000000000" HotelID="00000"/>
  <RoomType RoomTypeID="0" RatePlanID="1"/>
  <DateRange Type="Stay" Start="2012-08-23" End="2012-09-23"/>
  <RequestedLanguage>en</RequestedLanguage>
</GetHotelCxlPolicyRequest>
```

Response Example

```
<?xml version="1.0"?>
<GetHotelCxlPolicyResponse xmlns="http://xml.ycs.agoda.com">
  <StatusResponse status="200" AffectedRows="1" />
  <HotelCancellationPolicyList>
    <HotelCancellationPolicy>
      <RoomType RoomTypeID="0" RatePlanID="1" />
      <DateRange Type="Stay" Start="2012-08-10" End="2013-08-10" />
      <CancellationPolicy>
        <CancellationPolicyID>678981</CancellationPolicyID>
        <PolicyParameters>
          <Policies days="7" charge="N">1.0000</Policies>
          <Policies days="0" charge="N">1.0000</Policies>
        </PolicyParameters>
        <PolicyText language="en"><![CDATA[Any cancellation received within 7 day/s prior to arrival will incur the first 1 night/s charge. Failure to arrive at your hotel will be treated as a No-Show and will incur the first 1 night/s charge (Hotel policy).]]></PolicyText>
      </CancellationPolicy>
    </HotelCancellationPolicy>
  </HotelCancellationPolicyList>
</GetHotelCxlPolicyResponse>
```

GetAllCurrencies

Description

This message can be used to retrieve all Agoda supported currencies.



Helpful Hint: Regardless of the currencies supported, hotel will always receive booking information for their local currency (or otherwise specified in their contract). Currencies are always set with US Dollar currency as the base currency.

XML Elements and Attributes

GetAllCurrencies RQ					
Level	Name	Type	Occurs	Required	Comments
0	GetAllCurrenciesRequest		1	Y	
1	Authentication				
	@APIKey	String Length 32	1	Y	This is the primary authentication for connection to the Agoda API
	@HotelID	String Length 1-8	1	Y	Every request must contain a valid hotel ID that partner has access to

GetAllCurrencies RS					
Level	Name	Type	Occurs	Required	Comments
0	GetAllCurrenciesResponse		1	Y	
1	StatusResponse		1	Y	For further information on status responses see Response Codes and for code reference see Appendix D
	@Status	Numeric Length 3	1	Y	
	@AffectedRows	Numeric Length 1-4	1	Y	Affected rows indicates how many rows of data are returned (get request) or updated (set request)
1	CurrencyList		1	Y	Container with all currencies
	@Currency Code	Alpha Length 3	1-n	Y	Lists each valid currency available for this hotel on Agoda.com. A list of all currencies with descriptions are in Appendix E
	@Exchange Rate	Numeric	1-n	Y	Lists the Agoda exchange rate that will be used if pricing on website is converted to currency other than local currency. Base currency is always listed as USD

Request Example

```
<?xml version="1.0" encoding="UTF-8"?>
<GetAllCurrenciesRequest xmlns="http://xml.ycs.agoda.com">
  <Authentication APIKey="00000000-0000-0000-0000-000000000000"
    HotelID="123456"/>
</GetAllCurrenciesRequest>
```

Response Example

```
<?xml version="1.0" encoding="UTF-8"?>
<GetAllCurrenciesResponse xmlns="http://xml.ycs.agoda.com">
  <StatusResponse Status="200" AffectedRows="3" />
  <CurrencyList>
    <Currency code="EUR" ExchangeRate="0.7482"/>
    <Currency code="THB" ExchangeRate="32.4020"/>
    <Currency code="USD" ExchangeRate="1.000"/>
  </CurrencyList>
</GetAllCurrenciesResponse>
```

GetAllLanguages

Description

This API is used to retrieve all of Agoda's supported languages that can be used by the Hotel. The use of this request type is that many other API message types allow response CDATA to be returned in languages other than English. To find out which languages are supported in the API use this GetAllLanguages request.

XML Elements and Attributes

GetAllLanguages RQ					
Level	Name	Type	Occurs	Required	Comments
0	GetAllLanguagesRequest		1	Y	
1	Authentication				
	@APIKey	String Length 32	1	Y	This is the primary authentication for connection to the Agoda API
	@HotelID	String Length 1-8	1	Y	Every request must contain a valid hotel ID that partner has access to
1	RequestedLanguage	Alpha Length 2	1	N	All Agoda languages can be found by requesting GetAllLanguages . Note that currently any language request will still return only English names for Language types

GetAllLanguages RS					
Level	Name	Type	Occurs	Required	Comments
0	GetAllLanguagesResponse		1	Y	
1	StatusResponse		1	Y	For further information on status responses see Response Codes and for code reference see Appendix D
	@Status	Numeric Length 3	1	Y	
	@AffectedRows	Numeric Length 1-4	1	Y	Affected rows indicates how many rows of data are returned (get request) or updated (set request)
1	Language List		1	Y	Container with all languages
2	Language		1-n	Y	Contains one language information
3	Language ID	Numeric Length 1-2	1	Y	Agoda ID for language
3	LanguageAbrv	Alpha Length 2	1	Y	Abbreviated Language Code
3	LanguageLocale	Alpha aa-aa	1	Y	
3	LanguageDisplayName	Alpha String	1	Y	CDATA text name in language requested
	@language	Alpha Length 2	1	Y	

Example Request

```
<?xml version="1.0" encoding="UTF-8"?>
<GetAllLanguagesRequest xmlns="http://xml.ysc.agoda.com">
  <Authentication APIKey="00000000-0000-0000-0000-000000000000" HotelID="123456"/>
  <RequestedLanguage>en</RequestedLanguage>
</GetAllLanguagesRequest>
```


Example Response

```
<?xml version="1.0" encoding="UTF-8"?>
<GetAllLanguagesResponse xmlns="http://xml.ycs.agoda.com">
  <StatusResponse Status="200" AffectedRows="2"/>
  <LanguageList>
    <Language>
      <LanguageID>1</LanguageID>
      <LanguageAbbrv>EN</LanguageAbbrv>
      <LanguageLocale>en-us</LanguageLocale>
      <LanguageDisplayName
language="en"><![CDATA[English]]></LanguageDisplayName>
    </Language>
    <Language>
      <LanguageID>2</LanguageID>
      <LanguageAbbrv>FR</LanguageAbbrv>
      <LanguageLocale>fr-fr</LanguageLocale>
      <LanguageDisplayName
language="en"><![CDATA[Français]]></LanguageDisplayName>
    </Language>
  </LanguageList>
</GetAllLanguagesResponse>
```

GetHotelPromotions

Description

This API is used to retrieve the Hotel's Promotion information by Dates.



Important Note: This API is not yet completed and information will be updated in future version of document.

GetHotelRoomTypes

Description

This API is used to retrieve Hotel Room Types for a single hotel in YCS.



Helpful Hint: This is an important request type as it allows for Agoda rooms to be mapped to Channel Manager rooms. Send this request and then use the response message to link Agoda room type IDs to Channel Manager Room Type IDs. Room Type ID is a mandatory field for setting rate and inventory.

XML Elements and Attributes

GetHotelRoomTypes RQ					
Level	Name	Type	Occurs	Required	Comments
0	GetHotelRoomTypesRequest		1	Y	
1	Authentication				
	@APIKey	String Length 32	1	Y	This is the primary authentication for connection to the Agoda API
	@HotelID	String Length 1-8	1	Y	Every request must contain a valid hotel ID that partner has access to
1	RequestedLanguage	Alpha Length 2	1	N	All Agoda languages can be found by requesting GetAllLanguages

GetHotelRoomTypes RS					
Level	Name	Type	Occurs	Required	Comments
0	GetHotelRoomTypesResponse		1	Y	
1	StatusResponse		1	Y	For further information on status responses see Response Codes and for code reference see Appendix D
	@Status	Numeric Length 3	1	Y	
	@AffectedRows	Numeric Length 1 - 4	1	Y	Affected rows indicates how many rows of data are returned (get request) or updated (set request)
1	HotelRoomTypesList		1	Y	Container with all hotel room types
2	RoomType		1-n	Y	
	@RoomTypeID	Numeric	1	Y	Agoda Room Type ID. This is a very important ID as it is used in many requests related to room type updates
3	@language	Alpha Length 2	1	Y	
3	@CDATA	Alpha String	1	Y	CDATA room type text name in language requested

Example Request

```
<?xml version="1.0" encoding="utf-8" ?>
<GetHotelRoomTypesRequest xmlns="http://xml.ycs.agoda.com">
  <Authentication APIKey="00000000-0000-0000-0000-000000000000" HotelID="123456"/>
  <RequestedLanguage>en</RequestedLanguage>
</GetHotelRoomTypesRequest>
```

Example Response

```
<?xml version="1.0" encoding="utf-8" ?>
<GetHotelRoomTypesResponse xmlns="http://xml.ycs.agoda.com">
  <StatusResponse Status="200" AffectedRows="2" />
  <HotelRoomTypesList>
    <RoomType RoomTypeID="12345" language="en"><![CDATA[Standard room]]></RoomType>
    <RoomType RoomTypeID="67890" language="en"><![CDATA[Deluxe room]]></RoomType>
  </HotelRoomTypesList>
</GetHotelRoomTypesResponse>
```

GetHotelRoomSettings

Description

This API is used to retrieve Hotel Room configuration settings. These settings can be important when mapping rooms for channel manager as the response will show how many persons the room can accommodate, and also how many extra beds and/or children can be accommodated.



Helpful Hint: The Max Children per room is an absolute amount. This means that even if room allows for one or more extra bed, the maximum children amount per room is fixed.

XML Elements and Attributes

GetHotelRoomSettings RQ					
Level	Name	Type	Occurs	Required	Comments
0	GetHotelRoomSettingsRequest		1	Y	
1	Authentication				
	@APIKey	String Length 32	1	Y	This is the primary authentication for connection to the Agoda API
	@HotelID	String Length 1-8	1	Y	Every request must contain a valid hotel ID that partner has access to
1	RequestedLanguage	Alpha Length 2	1	N	All Agoda languages can be found by requesting GetAllLanguages

GetHotelRoomSettings RS					
Level	Name	Type	Occurs	Required	Comments
0	GetHotelRoomSettingsResponse		1	Y	
1	StatusResponse		1	Y	For further information on status responses see Response Codes and for code reference see Appendix D
	@Status	Numeric Length 3	1	Y	
	@AffectedRows	Numeric Length 1 - 4	1	Y	Affected rows indicates how many rows of data are returned (get request) or updated (set request)
1	HotelRoomTypesList		1	Y	Container with all hotel room types
2	RoomType		1-n	Y	
	@RoomTypeID	Numeric	1	Y	Agoda Room Type ID. This is a very important ID as it is used in many requests related to room type updates
	@MaxOccupancy	Numeric Length 1-2	1	Y	Maximum number of persons that can be accommodated in the room using existing room bedding. Includes both adults and children
	@MaxExtraBed	Numeric Length 1	1	Y	Maximum number of extra beds that accommodated in room. Each bed allows for 1 extra person.
	@MaxChildren	Numeric Length 1	1	Y	Maximum number of children that can be accommodated in this room. This number is part of the total max occupancy number

	@TotalNORooms	Numeric Length 1-3	1	Y	Total number of rooms for this room type set in Agoda extranet
	@language	Alpha Length 2	1	Y	
	@CDATA	Alpha String	1	Y	CDATA room type text name in language requested

Example Request

```
<?xml version="1.0" encoding="utf-8" ?>
<GetHotelRoomSettingsRequest xmlns="http://xml.ycs.agoda.com">
  <Authentication APIKey="00000000-0000-0000-0000-000000000000" HotelID="123456"/>
  <RequestedLanguage>en</RequestedLanguage>
</GetHotelRoomSettingsRequest>
```

Example Response

```
<?xml version="1.0" encoding="utf-8" ?>
<GetHotelRoomSettingsResponse xmlns="http://xml.ycs.agoda.com">
  <StatusResponse Status="200" AffectedRows="2" />
  <HotelRoomTypesList>
    <RoomType RoomTypeID="12345" MaxOccupancy="5" MaxExtraBed="1" MaxChildren="1" MaxInfants="0"
TotalNORooms="74" language="en"><![CDATA[Standard room]]></RoomType>
    <RoomType RoomTypeID="67890" MaxOccupancy="7" MaxExtraBed="2" MaxChildren="2" MaxInfants="0"
TotalNORooms="52" language="en"><![CDATA[Deluxe room]]></RoomType>
  </HotelRoomTypesList>
</GetHotelRoomSettingsResponse>
```

GetHotelRatePlans

Description

This API is used to retrieve Hotel Rate Plan information. Hotel Rate Plan is a new feature for Agoda in YCS4.

Agoda Rate Plans are *not* the same as Hotel Rate Plans. Agoda Rate Plans are a separate rate strategy for different markets or rate types. Examples of Agoda Rate Plans are:

- **Retail:** The default rate plan (set as rate plan 1) for sale on Agoda public website
- **Agoda Private Sale:** Rate plan used for opaque offers which are sold through the Agoda network to such providers as Thai Airways and Jetstar

Agoda rate plans can be managed independently of each other *or* based off each other. For example, if hotel uses two above rate plans and has two room types then following would apply:

Room Type	Rate Plan
Superior	Retail
Deluxe	Retail
Superior	Private Sale
Deluxe	Private Sale

These room types can be mapped to four separate room types in channel manager system or two room types on two rate plans in channel manager system. When updates are made using SetHotelInventory message for the same date and same room types, four separate HotelInventory elements are required. One for each room type / rate plan combination.

If hotel have rate plan based off the retail plan (e.g. retail plan less 10% and shared inventory pool) then channel manager is only required to update retail plan. This setup should be discussed with hotel customers.

Another important piece of information in the GetHotelRatePlans response is the Rate Type. Agoda supports four different rate types:

- **NetExclusive:** Pricing is set in Agoda YCS4 excluding taxes, service charges and Agoda commission. Rate will be marked up with commission and taxes and service charges added prior to customer purchase
- **NetInclusive:** Pricing is set in Agoda YCS4 including taxes, service charges and excluding Agoda commission. Rate will be marked up with commission prior to customer purchase
- **SellExclusive:** Pricing is set in Agoda YCS4 excluding taxes and service charges and including Agoda commission. Rate will have taxes and service charges added to price prior to customer purchase
- **SellInclusive:** Pricing is set in Agoda YCS4 including taxes, service charges and Agoda commission. Rate will be sold directly to customer at this price.



Important Note: Regardless of the rate type set for the hotel, the pricing returned in the GetBookingDetails response will always be net of commission. The pricing will list exclusive of tax and service price, tax, service and inclusive of tax and service price in the response

XML Elements and Attributes

GetHotelRatePlans RQ					
Level	Name	Type	Occurs	Required	Comments
0	GetHotelRatePlansRequest		1	Y	
1	Authentication				
	@APIKey	String Length 32	1	Y	This is the primary authentication for connection to the Agoda API
	@HotelID	String Length 1-8	1	Y	Every request must contain a valid hotel ID that partner has access to
1	RequestedLanguage	Alpha Length 2	1	N	All Agoda languages can be found by requesting GetAllLanguages

GetHotelRatePlans RS					
Level	Name	Type	Occurs	Required	Comments
0	GetHotelRatePlansResponse		1	Y	
1	StatusResponse		1	Y	For further information on status responses see Response Codes and for code reference see Appendix D
	@Status	Numeric Length 3	1	Y	
	@AffectedRows	Numeric Length 1 - 4	1	Y	Affected rows indicates how many rows of data are returned (get request) or updated (set request)
1	RatePlanList		1-n	Y	
2	ID	Numeric	1	Y	Agoda Rate Plan ID. This is a very important ID as it is used in many requests related to room and rate type updates
2	Name		1	Y	
	@language	Alpha Length 2	1	Y	
	@CDATA	Alpha String	1	Y	CDATA rate plan text name in language requested
2	RateType	Alpha String	1	Y	Type of Rate Plan setup. See Rate Plan Description for more info
2	Currency Code	Alpha Length 3	1-n	Y	Lists currency for this rate plan. A list of all currencies with descriptions are in Appendix E
2	MinRateAdj	Numeric nnn.nn			Shows percentage adjustment for this rate plan for minimum rate setting
2	MaxRateAdj	Numeric nnn.nn			Shows percentage adjustment for this rate plan for maximum rate setting

Example Request

```
<?xml version="1.0" encoding="UTF-8"?>
<GetHotelRatePlansRequest xmlns="http://xml.ycs.agoda.com">
  <Authentication APIKey="00000000-0000-0000-0000-000000000000" HotelID="123456"/>
  <RequestedLanguage>en</RequestedLanguage>
</GetHotelRatePlansRequest>
```


Example Response

```
<?xml version="1.0" encoding="UTF-8"?>
<GetHotelRatePlansResponse xmlns="http://xml.ycs.agoda.com">
  <StatusResponse Status="200" AffectedRows="2" />
  <RatePlanList>
    <RatePlan>
      <ID>1</ID>
      <Name language="en"><![CDATA[Retail]]></Name>
      <RateType>Nett Inclusive</RateType>
      <Currency>THB</Currency>
      <MinRateAdj>10.00</MinRateAdj>
      <MaxRateAdj>500.00</MaxRateAdj>
      <MinLOS>1</MinLOS>
      <MaxLOS>5</MaxLOS>
    </RatePlan>
    <RatePlan>
      <ID>2</ID>
      <Name language="en"><![CDATA[Opaque]]></Name>
      <RateType>Nett Exclusive</RateType>
      <Currency>THB</Currency>
      <MinRateAdj>15.00</MinRateAdj>
      <MaxRateAdj>300.00</MaxRateAdj>
      <MinLOS>1</MinLOS>
      <MaxLOS>3</MaxLOS>
      <MaxAllotment>5</MaxAllotment>
    </RatePlan>
  </RatePlanList>
</GetHotelRatePlansResponse>
```

GetHotelInventory

Description

This API is used to retrieve Hotel ARI (Availability, Rates and Inventory) by Rate Plan, Room and Dates.



Helpful Hint: An Agoda Rate plan must be submitted with every GetHotelInventory request. The agoda.com default rate plan is '1' (Retail) however other rate plans may apply to property.



Important Note: There are major changes between YCS3 and YCS4 relating to both Inventory and Pricing configuration.

In YCS3 there was only one option for occupancy model (submit single, double and triple rates). This resulted in limitations for properties with more than three person occupancy pricing for room.

In YCS4 properties are set by default to 'Full Occupancy' model which is similar to YCS3. There is single, double and full occupancy pricing – plus extra bed pricing based on room settings.

For properties that require more detailed pricing, there are options for Additional Person model or Adult and Child model. In Additional Person model property can set Single, Double and Additional Person pricing – plus extra bed pricing based on room settings. With this model, properties can have pricing increments for 3rd person upwards to the maximum room occupancy.

For Adult and Child model, a similar process occurs to above where pricing is set in increments for 3rd adult or child upwards to the maximum room occupancy.



Helpful Hint: For each model different pricing elements are mandatory. Ensure that you review below Elements and Attributes table to check which elements are mandatory for each occupancy model.

For inventory there is an important change to consider which relates to Agoda Guaranteed Allocations. In YCS3 the AvailableAllotment value was a combined total of Regular Allotment and Guaranteed (or Contracted) Allotment. This caused some problems when the AvailableAllotment value was attempted to be reduced *below* the Guaranteed Allotment.

In YCS4, these two allocations are split in the GetInventory response. In the SetInventory request the inventory value set by API partners is the Regular Allotment and the Guaranteed Allotment is not considered.



Helpful Hint: Guaranteed Allotments are used by Agoda for marketing and promotional purposes. Properties that confirm guaranteed allotments to Agoda will receive email marketing opportunities and page promotion on agoda.com so it is important for partners to work with hotels and agoda to support this function.



Important Note: Guaranteed Allotments are not considered in any inventory update or restrictions applied. It is important to consider that even if CloseOutRegularAllotment is set to True and RegularAllotment is 0 (zero), bookings can still be confirmed if property has Guaranteed Allotment rooms available.



Helpful Hint: There are two new restriction types available in YCS4 – Maximum Length of Stay and Closed to Departure.

XML Elements and Attributes

GetHotelInventory RQ					
Level	Name	Type	Occurs	Required	Comments
0	GetHotelInventoryRequest		1	Y	
1	Authentication				
	@APIKey	String Length 32	1	Y	This is the primary authentication for connection to the Agoda API
	@HotelID	String Length 1-8	1	Y	Every request must contain a valid hotel ID that partner has access to
1	RoomType		1	Y	
	@RoomTypeID	Numeric Length 1-8	1	Y	This is the Agoda room type ID that must be submitted to request availability, rate and inventory information (ARI). Use GetHotelRoomTypes to get IDs
	@RatePlanID	Numeric Length 1-3	1	Y	This is the Agoda rate plan ID that must be submitted to request availability, rate and inventory information (ARI). Use GetHotelRatePlans to get IDs
1	DateRange		1	Y	Date Range information for request
	@Type	Alpha	1	Y	Type of date range. There is only one option currently which is Stay
	@Start	Date YYYY-MM-DD	1	Y	The inclusive start date for the inventory query
	@End	Date YYYY-MM-DD	1	Y	The inclusive end date for the inventory query
1	RequestedLanguage	Alpha Length 2	1	N	All Agoda languages can be found by requesting GetAllLanguages

GetHotelInventory RS					
Level	Name	Type	Occurs	Required	Comments
0	GetHotelInventoryResponse		1	Y	
1	StatusResponse		1	Y	For further information on status responses see Response Codes and for code reference see Appendix D
	@Status	Numeric Length 3	1	Y	
	@AffectedRows	Numeric Length 1 - 4	1	Y	Affected rows indicates how many rows of data are returned (get request) or updated (set request)
1	HotelInventoryList		1	Y	Container for all hotel inventory elements
2	HotelInventory		1-n	Y	This element contains all ARI for date range/room type/rate plan.
3	RoomType		1	Y	
	@RoomTypeID	Numeric Length 1-8	1	Y	This is the Agoda room type ID that must be submitted to request availability, rate and inventory information (ARI). Use GetHotelRoomTypes to get IDs
	@RatePlanID	Numeric Length 1-3	1	Y	This is the Agoda rate plan ID that must be submitted to request availability, rate and inventory information (ARI). Use

					GetHotelRatePlans to get IDs
3	DateRange		1	Y	Date Range information for request
	@Type	Alpha	1	Y	Type of date range. There is only one option currently which is Stay
	@Start	Date YYYY-MM-DD	1	Y	The inclusive start date for the inventory response
	@End	Date YYYY-MM-DD	1	Y	The inclusive end date for the inventory response
3	InventoryRate		1	Y	Container with all elements related to room type pricing
	@Currency Code	Alpha Length 3	1	Y	Lists each valid currency available for this hotel on Agoda.com. A list of all currencies with descriptions are in Appendix E
4	SingleRate	Numeric nnn.nn	1	Y	This is the single rate amount for the single person pricing for a room type and is mandatory for all hotel occupancy models
4	DoubleRate	Numeric nnn.nn	1	N	This is the double rate amount for the double person pricing for a room type. This is mandatory if the room type is set to 2 or more in Max Occupancy on existing bedding for all occupancy models
4	FullRate	Numeric nnn.nn	1	N	This rate is used as the full rate for the room type if the hotel occupancy model is set to Full Rate. For more information see GetHotelInventory description
4	ExtraPerson	Numeric nnn.nn	1	N	This rate is used as the extra person rate for the room type if the hotel occupancy model is set to Additional Person. For more information see GetHotelInventory description
4	ExtraAdult	Numeric nnn.nn	1	N	This rate is used as the extra adult rate for the room type if the hotel occupancy model is set to Additional Adult and Child. For more information see GetHotelInventory description
4	ExtraChild	Numeric nnn.nn	1	N	This rate is used as the extra child rate for the room type if the hotel occupancy model is set to Additional Adult and Child. For more information see GetHotelInventory description
4	ExtraBed	Numeric nnn.nn	1	N	This rate is used as the extra bed rate for the room type for all occupancy models where max number of extra beds for this room is more than 0. Not mandatory and if element not sent extra bed price in YCS4 remains same as current. For more information see GetHotelInventory description
4	BreakfastIncluded	Boolean	1	Y	Denotes if this room type offer includes breakfast.

4	PromotionBlackout	Boolean	1	Y	Denotes if promotions are applicable for this room type for this date. If true no promotion offers on this room type will be considered.
4	MinLOS	Numeric length 1-2	1	Y	Minimum length of stay that is required to search and book this room type
4	MaxLOS	Numeric length 1-2	1	Y	Maximum length of stay that is required to search and book this room type
3	InventoryAllotment		1	Y	Container with all elements related to room type availability and inventory
4	RegularAllotment	Numeric length 1-3	1	Y	Total amount of regular allotment rooms in this room type available in YCS4 for sale to customers
4	RegularAllotmentUsed	Numeric length 1-3	1	Y	Total amount of regular allotment rooms in this room type sold in YCS to Agoda customers
4	GuaranteedAllotment	Numeric length 1-3	1	Y	Total amount of guaranteed allotment rooms in this room type available in YCS4 for sale to customers. See GetHotelInventory description for more info on Guarantee Allotment
4	GuaranteedAllotmentUsed	Numeric length 1-3	1	Y	Total amount of guaranteed allotment rooms in this room type sold to Agoda customers. See GetHotelInventory description for more info on Guaranteed Allotment
4	AutoTopUp	Numeric length 1-3	1	Y	Total amount of auto top-up rooms in this room type sold to Agoda customers. See GetHotelInventory description for more info on Auto Top up and recommended to always set to 0
4	CutOffDayNormal	Numeric length 1-3	1	Y	The total number of days prior to the inventory date that rooms will be released from sale for normal allotment. See GetHotelInventory description for info and recommended to always set to 0
4	CutOffDayGuaranteed	Numeric length 1-3	1	Y	The total number of days prior to the inventory date that rooms will be released from sale for guaranteed allotment. See GetHotelInventory description for info.
4	CloseOutRegularAllotment	Boolean	1	Y	Controls if room type is open for sale or closed to sale. Close will stop room type being sold even if inventory is available but will still allow Guaranteed Allotments to be sold
4	ClosedToArrival	Boolean	1	Y	Controls if room type is open for arrival for this date or closed for arrival on this date. Customer can

					still stay through this date if they arrive on past date
4	ClosedToDeparture	Boolean	1	Y	Controls if room type is open for departure for this date or closed for departure on this date. Customer can still stay through this date if they depart on future date

Example Request

```
<?xml version="1.0" encoding="utf-8" ?>
<GetHotelInventoryRequest xmlns="http://xml.ycs.agoda.com">
  <Authentication APIKey="00000000-0000-0000-0000-000000000000" HotelID="123456"/>
  <RoomType RoomTypeID="12345" RatePlanID="22"/>
  <DateRange Type="Stay" Start="2012-01-01" End="2012-01-31"/>
  <RequestedLanguage>en</RequestedLanguage>
</GetHotelInventoryRequest>
```

Example Response

```
<?xml version="1.0" encoding="utf-8" ?>
<GetHotelInventoryResponse xmlns="http://xml.ycs.agoda.com">
  <StatusResponse Status="200" AffectedRows="1" />
  <HotelInventoryList>
    <HotelInventory>
      <RoomType RoomTypeID="11111" RatePlanID="1" />
      <DateRange Type="Stay" Start="2012-01-21"/>
      <InventoryRate Currency="THB">
        <SingleRate>1000.00</SingleRate>
        <DoubleRate>1500.00</DoubleRate>
        <FullRate>1500.00</FullRate>
        <ExtraPerson>700.00</ExtraPerson>
        <ExtraAdult>700.00</ExtraAdult>
        <ExtraChild>550.00</ExtraChild>
        <ExtraBed>650.00</ExtraBed>
        <BreakfastIncluded>true</BreakfastIncluded>
        <PromotionBlackout>true</PromotionBlackout>
        <MinLOS>1</MinLOS>
        <MaxLOS>5</MaxLOS>
      </InventoryRate>
      <InventoryAllotment>
        <RegularAllotment>3</RegularAllotment>
        <RegularAllotmentUsed>0</RegularAllotmentUsed>
        <GuaranteedAllotment>1</GuaranteedAllotment>
        <GuaranteedAllotmentUsed>0</GuaranteedAllotmentUsed>
        <AutoTopUp>10</AutoTopUp>
        <UsedAutoTopUp>0</UsedAutoTopUp>
        <CutOffDayNormal>2</CutOffDayNormal>
        <CutOffDayGuarenteed>1</CutOffDayGuarenteed>
        <CloseOutRegularAllotment>false</CloseOutRegularAllotment>
        <ClosedToArrival>false</ClosedToArrival>
        <ClosedToDeparture>false</ClosedToDeparture>
      </InventoryAllotment>
    </HotelInventory>
  </HotelInventoryList>
</GetHotelInventoryResponse>
```

GetAllSurchargeTypes

Description

This API is used to retrieve all Agoda surcharge types which can be used by the hotels.

XML Elements and Attributes

GetAllSurchargeTypes RQ					
Level	Name	Type	Occurs	Required	Comments
0	GetAllSurchargeTypesRequest		1	Y	
1	Authentication				
	@APIKey	String Length 32	1	Y	This is the primary authentication for connection to the Agoda API
	@HotelID	String Length 1-8	1	Y	Every request must contain a valid hotel ID that partner has access to
1	RequestedLanguage	Alpha Length 2	1	N	All Agoda languages can be found by requesting GetAllLanguages

GetAllSurchargeTypes RS					
Level	Name	Type	Occurs	Required	Comments
0	GetAllCxlPoliciesResponse		1	Y	
1	StatusResponse		1	Y	For further information on status responses see Response Codes and for code reference see Appendix D
	@Status	Numeric Length 3	1	Y	
	@AffectedRows	Numeric Length 1 - 4	1	Y	Affected rows indicates how many rows of data are returned (get request) or updated (set request)
1	SurchargeTypeList		1	Y	Contains each surcharge type
2	SurchargeType		1-n	Y	
3	SurchargeTypeID	Numeric Length 1-4	1	Y	Each Agoda surcharge type has a unique ID. This ID must be used when setting hotel surcharges using SetHotelSurcharge
3	SurchargeApplyType	Alpha Length 1	1	Y	Denotes type of surcharge. There are six options: M – Mandatory O – Optional E – Excluded I – Included C – Compulsory B – Breakfast See GetAllSurchargeTypes introduction for more information
3	SurchargeName	CDATA	1	Y	Plain text surcharge name in language requested
	@language	Alpha Length 2	1	Y	
3	SurchargeDescription	CDATA	1	Y	Plain text surcharge description in language requested
	@language	Alpha Length 2	1	Y	
3	ChargeTypeCode	Alpha Length 2-4	1	Y	Denotes charge type of surcharge (i.e. per person per night). See

					GetAllSurchargeTypes introduction for more information and full list
3	IsPercent	Boolean	1	Y	Denotes if surcharge is a percentage or a flat amount

Example Request

```
<?xml version="1.0" encoding="UTF-8"?>
<GetAllSurchargeTypesRequest xmlns="http://xml.ycs.agoda.com">
  <Authentication APIKey="00000000-0000-0000-0000-000000000000" HotelID="123456"/>
  <RequestedLanguage>en</RequestedLanguage>
</GetAllSurchargeTypesRequest>
```

Example Response

```
<?xml version="1.0" encoding="utf-8" ?>
<GetAllSurchargeTypesResponse xmlns="http://xml.ycs.agoda.com">
  <StatusResponse Status="200" AffectedRows="2" />
  <SurchargeTypeList>
    <SurchargeType>
      <SurchargeTypeID>5</SurchargeTypeID>
      <SurchargeApplyType>M</SurchargeApplyType>
      <SurchargeName language="en"><![CDATA[Gala Dinner]]></SurchargeName>
      <SurchargeDescription language="en"><![CDATA[Optional Gala Dinner with free flow wine and chocolate fountain]]></SurchargeDescription>
      <ChargeTypeCode>PGPN</ChargeTypeCode>
      <IsPercent>false</IsPercent>
    </SurchargeType>
    <SurchargeType>
      <SurchargeTypeID>6</SurchargeTypeID>
      <SurchargeApplyType>C</SurchargeApplyType>
      <SurchargeName language="en"><![CDATA[New Years Eve Gala Dinner]]></SurchargeName>
      <SurchargeDescription language="en"><![CDATA[Compulsory Seafood Gala Dinner with chocolate fountain]]></SurchargeDescription>
      <ChargeTypeCode>PGPN</ChargeTypeCode>
      <IsPercent>true</IsPercent>
    </SurchargeType>
  </SurchargeTypeList>
</GetAllSurchargeTypesResponse>
```


GetHotelSurcharge

Description

This API is used to retrieve the Hotel Surcharge information by Dates.

XML Elements and Attributes

GetHotelSurcharge RQ					
Level	Name	Type	Occurs	Required	Comments
0	GetHotelSurchargeRequest		1	Y	
1	Authentication				
	@APIKey	String Length 32	1	Y	This is the primary authentication for connection to the Agoda API
	@HotelID	String Length 1-8	1	Y	Every request must contain a valid hotel ID that partner has access to
1	DateRange		1	Y	Date Range information for request
	@Type	Alpha	1	Y	Type of date range. There is only one option currently which is Stay
	@Start	Date YYYY-MM-DD	1	Y	The inclusive start date for the query
	@End	Date YYYY-MM-DD	1	Y	The inclusive end date for the query
1	RequestedLanguage	Alpha Length 2	1	N	All Agoda languages can be found by requesting GetAllLanguages

GetHotelSurcharge RS					
Level	Name	Type	Occurs	Required	Comments
0	GetHotelSurchargeResponse		1	Y	
1	StatusResponse		1	Y	For further information on status responses see Response Codes and for code reference see Appendix D
	@Status	Numeric Length 3	1	Y	
	@AffectedRows	Numeric Length 1 - 4	1	Y	Affected rows indicates how many rows of data are returned (get request) or updated (set request)
1	HotelSurchargesList		1	Y	Contains each hotel surcharge
2	HotelSurcharge		1-n	Y	
3	RoomType	CDATA	1	Y	Plain text description of room type in language requested.
	@RoomTypeID	Numeric Length 1-8	1	Y	Room Type ID linked to surcharge. Note that RoomTypeID = 0 (zero) is for all room types
	@RatePlanID	Numeric Length 1-4	1	Y	Rate Plan ID linked to surcharge
	@Language	Alpha Length 2	1	Y	Language requested for response
3	DateRange		1	Y	Date Range information for request
	@Type	Alpha	1	Y	Type of date range. There is only one option currently which is Stay
	@Start	Date YYYY-MM-DD	1	Y	The inclusive start date for the query
	@End	Date YYYY-MM-DD	1	Y	The inclusive end date for the query
3	SurchargePolicy		1	Y	Container for surcharge information

4	SurchargeID	Numeric Length 1-8	1	Y	Unique surcharge ID for surcharge policy for this hotel.
4	SurchargeTypeID	Numeric Length 1-4	1	Y	Surcharge ID type. Use GetAllSurchargeTypes request for full list
4	Currency	Alpha Length 3	1-n	Y	Lists currency for this surcharge. A list of all currencies with descriptions are in Appendix E
4	Amount	Numeric Length 1-8	1	Y	Total cost of surcharge
4	IsPercentage	Boolean	1	Y	Denotes if Amount is percentage or flat amount
4	IsCommissionable	Boolean	1	Y	Denotes if surcharge is commissionable
4	Description	CDATA	1	Y	Plain text surcharge description in language requested
	@language	Alpha Length 2	1	Y	

Example Request

```
<?xml version="1.0" encoding="utf-8" ?>
<GetHotelSurchargeResponse xmlns="http://xml.ycs.agoda.com">
  <Authentication APIKey="00000000-0000-0000-0000-000000000000" HotelID="123456"/>
  <DateRange Type="Stay" Start="2012-01-01" End="2012-01-31"/>
  <RequestedLanguage>en</RequestedLanguage>
</GetHotelSurchargeResponse>
```

Example Response

```
<?xml version="1.0" encoding="utf-8" ?>
<GetHotelSurchargeResponse xmlns="http://xml.ycs.agoda.com">
  <StatusResponse Status="200" AffectedRows="2" />
  <HotelSurchargesList>
    <HotelSurcharge>
      <RoomType RoomTypeID="12345" RatePlanID="22" language="en"><![CDATA[Standard room]]></RoomType>
      <DateRange Type="Stay" Start="2012-01-01" End="2012-01-31"/>
      <SurchargePolicy>
        <SurchargeID>5627</SurchargeID>
        <SurchargeTypeID>5</SurchargeTypeID>
        <Currency>THB</Currency>
        <Amount>100.00</Amount>
        <IsPercent>false</IsPercent>
        <IsCommissionable>false</IsCommissionable>
        <Description language="en"><![CDATA[Boat Transfer (Adult x [ADULTS])]]></Description>
      </SurchargePolicy>
    </HotelSurcharge>
    <HotelSurcharge>
      <RoomType RoomTypeID="12345" RatePlanID="22" language="en"><![CDATA[Standard room]]></RoomType>
      <DateRange Type="Stay" Start="2012-01-01" End="2012-01-31"/>
      <SurchargePolicy>
        <SurchargeID>5628</SurchargeID>
        <SurchargeTypeID>42</SurchargeTypeID>
        <Currency />
        <Amount>10.00</Amount>
        <IsPercent>true</IsPercent>
        <IsCommissionable>true</IsCommissionable>
        <Description language="en"><![CDATA[Chinese New Year [NIGHTS] night/s (Adult x [ADULTS])]]></Description>
      </SurchargePolicy>
    </HotelSurcharge>
  </HotelSurchargesList>
</GetHotelSurchargeResponse>
```

GetAllValueAdds

Description

This API is used to retrieve the list of Value Add items that can be applied to Hotel Value Add packages.

XML Elements and Attributes

GetAllValueAdds RQ					
Level	Name	Type	Occurs	Required	Comments
0	GetAllValueAddsRequest		1	Y	
1	Authentication				
	@APIKey	String Length 32	1	Y	This is the primary authentication for connection to the Agoda API
	@HotelID	String Length 1-8	1	Y	Every request must contain a valid hotel ID that partner has access to
1	RequestedLanguage	Alpha Length 2	1	N	All Agoda languages can be found by requesting GetAllLanguages

GetAllValueAdds RS					
Level	Name	Type	Occurs	Required	Comments
0	GetAllValueAddsResponse		1	Y	
1	StatusResponse		1	Y	For further information on status responses see Response Codes and for code reference see Appendix D
	@Status	Numeric Length 3	1	Y	
	@AffectedRows	Numeric Length 1 - 4	1	Y	Affected rows indicates how many rows of data are returned (get request) or updated (set request)
1	ValueAddItemList		1	Y	Contains each Agoda value add
2	ValueAddItem		1-n	Y	
3	ValueAddItemID	Numeric Length 1-8	1	Y	Agoda Value Add Item ID. This is an ID as it is used in GetHotelValueAdds and SetHotelValueAdds
3	ValueAddItemName	CDATA	1	Y	Plain text value add description in language requested
	@language	Alpha Length 2	1	Y	

Example Request

```
<?xml version="1.0" encoding="UTF-8"?>
<GetAllValueAddsRequest xmlns="http://xml.ycs.agoda.com">
  <Authentication APIKey="00000000-0000-0000-0000-000000000000" HotelID="123456"/>
  <RequestedLanguage>en</RequestedLanguage>
</GetAllValueAddsRequest>
```

Example Response

```
<?xml version="1.0" encoding="UTF-8"?>
<GetAllValueAddsResponse xmlns="http://xml.ycs.agoda.com">
  <StatusResponse Status="200" AffectedRows="3" />
  <ValueAddItemList>
    <ValueAddItem>
      <ValueAddItemID>1</ValueAddItemID>
      <ValueAddItemName language="en"><![CDATA[Free spa]]></ValueAddItemName>
    </ValueAddItem>
    <ValueAddItem>
      <ValueAddItemID>2</ValueAddItemID>
      <ValueAddItemName language="en"><![CDATA[Fruit basket]]></ValueAddItemName>
    </ValueAddItem>
    <ValueAddItem>
      <ValueAddItemID>3</ValueAddItemID>
      <ValueAddItemName language="en"><![CDATA[Bottle of wine]]></ValueAddItemName>
    </ValueAddItem>
  </ValueAddItemList>
</GetAllValueAddsResponse>
```

GetHotelValueAdds

Description

This API is used to retrieve Hotel Value Add package information by dates.

XML Elements and Attributes

GetHotelValueAdds RQ					
Level	Name	Type	Occurs	Required	Comments
0	GetHotelValueAddsRequest		1	Y	
1	Authentication				
	@APIKey	String Length 32	1	Y	This is the primary authentication for connection to the Agoda API
	@HotelID	String Length 1-8	1	Y	Every request must contain a valid hotel ID that partner has access to
1	DateRange		1	Y	Date Range information for request
	@Type	Alpha	1	Y	Type of date range. There is only one option currently which is Stay
	@Start	Date YYYY-MM-DD	1	Y	The inclusive start date for the query
	@End	Date YYYY-MM-DD	1	Y	The inclusive end date for the query
1	RequestedLanguage	Alpha Length 2	1	N	All Agoda languages can be found by requesting GetAllLanguages

GetHotelValueAdds RS					
Level	Name	Type	Occurs	Required	Comments
0	GetHotelValueAddsResponse		1	Y	
1	StatusResponse		1	Y	For further information on status responses see Response Codes and for code reference see Appendix D
	@Status	Numeric Length 3	1	Y	
	@AffectedRows	Numeric Length 1 - 4	1	Y	Affected rows indicates how many rows of data are returned (get request) or updated (set request)
1	HotelValueAddsList		1	Y	Contains each hotel value add
2	ValueAdd		1-n	Y	
3	RoomType	CDATA	1	Y	Plain text description of room type in language requested.
	@RoomTypeID	Numeric Length 1-8	1	Y	Room Type ID linked to surcharge. Note that RoomTypeID = 0 (zero) is for all room types
	@RatePlanID	Numeric Length 1-4	1	Y	Rate Plan ID linked to surcharge
	@Language	Alpha Length 2	1	Y	Language requested for response
3	DateRange		1	Y	Date Range information for request
	@Type	Alpha	1	Y	Type of date range. There is only one option currently which is Stay
	@Start	Date YYYY-MM-DD	1	Y	The inclusive start date for the query
	@End	Date YYYY-MM-DD	1	Y	The inclusive end date for the query
3	Currency	Alpha Length 3	1-n	Y	Lists currency for this surcharge. A

					list of all currencies with descriptions are in Appendix E
3	TotalPackageValue	Numeric nnn.nn	1	Y	Total cost of value add
3	ApplyDaily	Boolean	1	Y	Denotes if surcharge is applied daily to to entire stay
4	ValueAddItemList		1	Y	Each Value Add can have multiple items. Item List will list each item that is contained in value add
5	ItemID	Alpha	1-n	Y	
	@ValueAddItemID	Numeric Length 1-8	1-2	Y	Agoda ID for value add. Use GetAllValueAdds request for full list of Value Adds

Example Request

```
<?xml version="1.0" encoding="utf-8" ?>
<GetHotelValueAddRequest xmlns="http://xml.ycs.agoda.com">
  <Authentication APIKey="00000000-0000-0000-0000-000000000000" HotelID="123456"/>
  <DateRange Type="Stay" Start="2012-01-01" End="2012-01-31"/>
  <RequestedLanguage>en</RequestedLanguage>
</GetHotelValueAddRequest>
```

Example Response

```
<?xml version="1.0" encoding="utf-8" ?>
<GetHotelValueAddResponse xmlns="http://xml.ycs.agoda.com">
  <StatusResponse Status="200" AffectedRows="1" />
  <HotelValueAddsList>
    <ValueAdd>
      <RoomType RoomTypeID="12345" RatePlanID="22" language="en"><![CDATA[Standard room]]></RoomType>
      <DateRange Type="Stay" Start="2012-01-01" End="2012-01-31"/>
      <Currency>USD</Currency>
      <TotalPackageValue>200.00</TotalPackageValue>
      <ApplyDaily>true</ApplyDaily>
      <ValueAddItemList>
        <ItemID ValueAddItemID="1">Free Spa</ItemID>
        <ItemID ValueAddItemID="2">Fruit basket</ItemID>
        <ItemID ValueAddItemID="3">Bottle of wine</ItemID>
      </ValueAddItemList>
    </ValueAdd>
  </HotelValueAddsList>
</GetHotelValueAddResponse>
```

XML Set Operations

SetHotelInventory

Description

This API is used to create or update the Hotel's Allotment by Rooms, Rate Plans and Dates.

XML Elements and Attributes

SetHotelInventory RQ					
Level	Name	Type	Occurs	Required	Comments
0	SetHotelInventoryRequest		1	Y	
	Authentication				
1	@APIKey	String Length 32	1	Y	This is the primary authentication for connection to the Agoda API
	@HotelID	String Length 1-8	1	Y	Every request must contain a valid hotel ID that partner has access to
	@AffectedRows	Numeric Length 1 - 4	1	Y	Affected rows indicates how many rows of data are returned (get request) or updated (set request)
1	HotelInventoryList		1	Y	Container for all hotel inventory elements
2	HotelInventory		1-n	Y	This element contains all ARI for date range/room type/rate plan.
3	RoomType		1	Y	
	@RoomTypeID	Numeric Length 1-8	1	Y	This is the Agoda room type ID related to the hotel inventory element. Agoda Room type IDs can be returned with GetHotelRoomTypes request
	@RatePlanID	Numeric Length 1-3	1	Y	This is the Agoda rate plan ID related to the hotel inventory element. Agoda Room type IDs can be returned with GetHotelRatePlans request
3	DateRange		1	Y	Date Range information for request
	@Type	Alpha	1	Y	Type of date range. There is only one option currently which is Stay
	@Start	Date YYYY-MM-DD	1	Y	The inclusive start date for the inventory response
	@End	Date YYYY-MM-DD	1	Y	The inclusive end date for the inventory response
3	InventoryRate		1	N	Container with all elements related to room type pricing. Entire container is not mandatory to message but if included then below mandatory logic applies.
	@Currency Code	Alpha Length 3	1	Y	Lists each valid currency available for this hotel on Agoda.com. A list of all currencies with descriptions are in Appendix E
4	SingleRate	Numeric nnn.nn	1	Y	This is the single rate amount for the single person pricing for a room

					type and is mandatory for all hotel occupancy models
4	DoubleRate	Numeric nnn.nn	1	N	This is the double rate amount for the double person pricing for a room type. This is mandatory if the room type is set to 2 or more in Max Occupancy on existing bedding for all occupancy models
4	FullRate	Numeric nnn.nn	1	N	This rate is used as the full rate for the room type if the hotel occupancy model is set to Full Rate. For more information see GetHotelInventory description
4	ExtraPerson	Numeric nnn.nn	1	N	This rate is used as the extra person rate for the room type if the hotel occupancy model is set to Additional Person. For more information see GetHotelInventory description
4	ExtraAdult	Numeric nnn.nn	1	N	This rate is used as the extra adult rate for the room type if the hotel occupancy model is set to Additional Adult and Child. For more information see GetHotelInventory description
4	ExtraChild	Numeric nnn.nn	1	N	This rate is used as the extra child rate for the room type if the hotel occupancy model is set to Additional Adult and Child. For more information see GetHotelInventory description
4	ExtraBed	Numeric nnn.nn	1	N	This rate is used as the extra bed rate for the room type for all occupancy models where max number of extra beds for this room is more than 0. Not mandatory and if element not sent extra bed price in YCS4 remains same as current. For more information see GetHotelInventory description
4	BreakfastIncluded	Boolean	1	N	Denotes if this room type offer includes breakfast. Not mandatory and if element not sent, breakfast flag in YCS4 remains same as current setting
4	PromotionBlackout	Boolean	1	N	Denotes if promotions are applicable for this room type for this date. If true no promotion offers on this room type will be considered. Not mandatory and if element not sent, promotion blackout in YCS4 remains same as current setting
4	MinLOS	Numeric length 1-2	1	N	Minimum length of stay that is required to search and book this room type. . Not mandatory and if element not sent, MLOS in YCS4 remains same as current setting

4	MaxLOS	Numeric length 1-2	1	N	Maximum length of stay that is required to search and book this room type. Not mandatory and if element not sent, MaxLOS in YCS4 remains same as current setting
3	InventoryAllotment		1	N	Container with all elements related to room type availability and inventory. Entire container is not mandatory to message
4	RegularAllotment	Numeric length 1-3	1	N	Set total amount of regular allotment rooms in this room type available in YCS4 for sale to customers. Not mandatory and if element not sent, regular allotment in YCS4 remains same as current
4	CutOffDayNormal	Numeric length 1-3	1	N	The total number of days prior to the inventory date that rooms will be released from sale for normal allotment. See GetHotelInventory description for info. Not mandatory and if element not sent, Cut Off in YCS4 remains same as current. Recommended to always set to 0
4	CloseOutRegularAllotment	Boolean	1	N	Controls if room type is open for sale or closed to sale. Close will stop room type regular inventory being sold even if inventory is available. Guaranteed inventory is still available for sale if close is applied. Not mandatory and if element not sent, close out status in YCS4 remains same as current.
4	ClosedToArrival	Boolean	1	Y	Set to make room type open for arrival for this date or closed for arrival on this date. Customer can still stay through this date if they arrive on past date. Not mandatory and if element not sent, Close to Arrival status in YCS4 remains same as current.
4	ClosedToDeparture	Boolean	1	Y	Set to make room type open for departure for this date or closed for departure on this date. Customer can still stay through this date if they depart on future date. Not mandatory and if element not sent, Close to Departure status in YCS4 remains same as current.

SetHotelInventory RS

Level	Name	Type	Occurs	Required	Comments
0	SetHotelInventoryResponse		1	Y	
1	StatusResponse		1	Y	For further information on status responses see Response Codes and for code reference see Appendix D
	@Status	Numeric Length 3	1	Y	
	@AffectedRows	Numeric Length 1 - 4	1	Y	Affected rows indicates how many rows of data are returned (get request) or updated (set request)

Example Request

```
<?xml version="1.0" encoding="UTF-8"?>
<SetHotelInventoryList xmlns="http://xml.ycs.agoda.com">
  <Authentication APIKey="00000000-0000-0000-0000-000000000000" HotelID="123456"/>
  <HotelInventoryList>
    <HotelInventory>
      <RoomType RoomTypeID="12345" RatePlanID="22"/>
      <DateRange Type="Stay" Start="2012-06-01" End="2012-06-30"/>
      <InventoryRate Currency="THB">
        <SingleRate>1000.00</SingleRate>
        <DoubleRate>1500.00</DoubleRate>
        <FullRate>1500.00</FullRate>
        <ExtraPerson>150.00</ExtraPerson>
        <ExtraAdult>150.00</ExtraAdult>
        <ExtraChild>150.00</ExtraChild>
        <ExtraBed>150.00</ExtraBed>
        <BreakfastIncluded>true</BreakfastIncluded>
        <PromotionBlackout>false</PromotionBlackout>
        <MinLOS>2</MinLOS>
        <MaxLOS>5</MaxLOS>
      </InventoryRate>
      <InventoryAllotment>
        <RegularAllotment>10</RegularAllotment>
        <AutoTopUp>10</AutoTopUp>
        <CutOffDayNormal>0</CutOffDayNormal>
        <CloseOutRegularAllotment>false</CloseOutRegularAllotment>
        <ClosedToArrival>false</ClosedToArrival>
        <ClosedToDeparture>false</ClosedToDeparture>
      </InventoryAllotment>
    </HotelInventory>
  </HotelInventoryList>
</SetHotelInventoryList>
```

Example Response

```
<?xml version="1.0" encoding="utf-8" ?>
<SetHotelInventoryResponse xmlns="http://xml.ycs.agoda.com">
  <StatusResponse Status="200" AffectedRows="1" />
</SetHotelInventoryResponse>
```

SetHotelCxlPolicy

Description

This API is used to create or update the Hotel's Cancellation Policies by Rooms, Rate Plans and Dates.

XML Elements and Attributes

SetHotelCxlPolicy RS					
Level	Name	Type	Occurs	Required	Comments
0	SetHotelCxlPolicyResponse		1	Y	
1	Authentication				
	@APIKey	String Length 32	1	Y	This is the primary authentication for connection to the Agoda API
	@HotelID	String Length 1-8	1	Y	Every request must contain a valid hotel ID that partner has access to
1	HotelCancellationPolicyList		1	Y	Contains each cancellation policy
2	HotelCancellationPolicy		1-n	Y	
3	RoomType		1	Y	
	@RoomTypeID	Numeric Length 1-8	1	Y	This is the Agoda room type ID related to cancellation policy. Use GetHotelRoomTypes to get IDs
	@RatePlanID	Numeric Length 1-3	1	Y	This is the Agoda rate plan ID related to cancellation policy. Use GetHotelRatePlans to get IDs
3	DateRange		1	Y	Date Range information for request
	@Type	Alpha	1	Y	Type of date range. There is only one option currently which is Stay
	@Start	Date YYYY-MM-DD	1	Y	The inclusive start date for the policy setting
	@End	Date YYYY-MM-DD	1	Y	The inclusive end date for the policy setting
3	CancellationPolicy		1-n	Y	
3	CancellationPolicyID	Numeric Length 4-6	1	Y	Each Agoda cancellation policy has a unique ID. This ID relates to this Agoda cancel policy. Use GetAllCxlPolicies to set hotel policy

SetHotelInventory RS					
Level	Name	Type	Occurs	Required	Comments
0	SetHotelInventoryResponse		1	Y	
1	StatusResponse		1	Y	For further information on status responses see Response Codes and for code reference see Appendix D
	@Status	Numeric Length 3	1	Y	
	@AffectedRows	Numeric Length 1 - 4	1	Y	Affected rows indicates how many rows of data are returned (get request) or updated (set request)

Example Request

```
<?xml version="1.0" encoding="utf-8" ?>
<SetHotelClxPolicyRequest xmlns="http://xml.ycs.agoda.com">
  <Authentication APIKey="00000000-0000-0000-0000-000000000000" HotelID="123456"/>
  <HotelCancellationsList>
    <HotelCancellationPolicy>
      <RoomType RoomTypeID="12345" RatePlanID="22"/>
      <DateRange Type="Stay" Start="2012-01-01" End="2012-01-31"/>
      <CancellationPolicy>
        <CancellationPolicyID>12</CancellationPolicyID>
      </CancellationPolicy>
    </HotelCancellationPolicy>
  </HotelCancellationsList>
</SetHotelClxPolicyRequest>
```

Example Response

```
<?xml version="1.0" encoding="utf-8" ?>
<SetHotelClxPolicyResponse xmlns="http://xml.ycs.agoda.com">
  <StatusResponse Status="200" AffectedRows="1" />
</SetHotelClxPolicyResponse>
```

SetHotelPromotion

Description

This API is used to create or update Hotel's Promotions by Rooms, Rate Plans, and Dates.



Important Note: This API is not yet completed and information will be updated in future version of document.

SetHotelRoomSettings

Description

This API is used to update the Hotel's Room Occupancy information by room.

XML Elements and Attributes

SetHotelRoomSettings RS					
Level	Name	Type	Occurs	Required	Comments
0	SetHotelRoomSettingsRequest		1	Y	
1	Authentication				
	@APIKey	String Length 32	1	Y	This is the primary authentication for connection to the Agoda API
	@HotelID	String Length 1-8	1	Y	Every request must contain a valid hotel ID that partner has access to
1	HotelRoomTypesList		1	Y	Container with all hotel room types
2	RoomType		1-n	Y	
	@RoomTypeID	Numeric	1	Y	Agoda Room Type ID. This is a very important ID as it is must be used for this request
	@MaxOccupancy	Numeric Length 1-2	1	Y	Maximum number of persons that can be accommodated in the room using existing room bedding. Includes both adults and children
	@MaxExtraBed	Numeric Length 1	1	Y	Maximum number of extra beds that accommodated in room. Each bed allows for 1 extra person.
	@MaxChildren	Numeric Length 1	1	Y	Maximum number of children that can be accommodated in this room. This number is part of the total max occupancy number
	@TotalNORooms	Numeric Length 1-3	1	Y	Total number of rooms for this room type set in Agoda extranet

SetHotelRoomSettings RS					
Level	Name	Type	Occurs	Required	Comments
0	SetHotelRoomSettingsResponse		1	Y	
1	StatusResponse		1	Y	For further information on status responses see Response Codes and for code reference see Appendix D
	@Status	Numeric Length 3	1	Y	
	@AffectedRows	Numeric Length 1 - 4	1	Y	Affected rows indicates how many rows of data are returned (get request) or updated (set request)

Example Request

```
<?xml version="1.0" encoding="utf-8" ?>
<SetHotelRoomOccupancyRequest xmlns="http://xml.ycs.agoda.com">
  <Authentication APIKey="00000000-0000-0000-0000-000000000000" HotelID="123456"/>
  <HotelRoomTypesList>
    <RoomType RoomTypeID="12345" MaxOccupancy="5" MaxExtraBed="1" MaxChildren="1" MaxInfants="0"
TotalNORooms="74"/>
    <RoomType RoomTypeID="67890" MaxOccupancy="7" MaxExtraBed="2" MaxChildren="2" MaxInfants="0"
TotalNORooms="52"/>
  </HotelRoomTypesList>
</SetHotelRoomOccupancyRequest>
```

Example Response

```
<?xml version="1.0" encoding="utf-8" ?>
<SetHotelRoomOccupancyResponse xmlns="http://xml.ycs.agoda.com">
  <StatusResponse Status="200" AffectedRows="2" />
</SetHotelRoomOccupancyResponse>
```


SetHotelSurcharge

Description

This API is used to create or update the Hotel's Surcharges by Rooms, Rate Plans and Dates.

XML Elements and Attributes

SetHotelSurcharge RS					
Level	Name	Type	Occurs	Required	Comments
0	SetHotelSurchargeResponse		1	Y	
1	Authentication				
	@APIKey	String Length 32	1	Y	This is the primary authentication for connection to the Agoda API
	@HotelID	String Length 1-8	1	Y	Every request must contain a valid hotel ID that partner has access to
1	HotelSurchargesList		1	Y	Contains each hotel surcharge
2	HotelSurcharge		1-n	Y	
3	RoomType				
	@RoomTypeID	Numeric Length 1-8	1	Y	Room Type ID linked to surcharge. Note that RoomTypeID = 0 (zero) is for all room types
	@RatePlanID	Numeric Length 1-4	1	Y	Rate Plan ID linked to surcharge
3	DateRange		1	Y	Date Range information for request
	@Type	Alpha	1	Y	Type of date range. There is only one option currently which is Stay
	@Start	Date YYYY-MM-DD	1	Y	The inclusive start date for the query
	@End	Date YYYY-MM-DD	1	Y	The inclusive end date for the query
3	SurchargePolicy		1	Y	Container for surcharge information
4	SurchargeTypeID	Numeric Length 1-4	1	Y	Surcharge ID type. Use GetAllSurchargeTypes request for full list
4	IsCommissionable	Boolean	1	Y	Denotes if surcharge is commissionable
4	Currency	Alpha Length 3	1-n	Y	Lists currency for this surcharge. A list of all currencies with descriptions are in Appendix E
4	Amount	Numeric Length 1-8	1	Y	Total cost of surcharge
4	DayApply	Numeric Length 7	1	Y	Sets which days of week this surcharge applies. In numeric format where 0 is false and 1 is true. Position 1 = Sunday Position 2 = Monday Position 3 = Tuesday Position 4 = Wednesday Position 5 = Thursday Position 6 = Friday Position 7 = Saturday

SetHotelSurcharge RS					
Level	Name	Type	Occurs	Required	Comments
0	SetHotelSurchargeResponse		1	Y	
1	StatusResponse		1	Y	For further information on status responses see Response Codes and for code reference see Appendix D
	@Status	Numeric Length 3	1	Y	
	@AffectedRows	Numeric Length 1 - 4	1	Y	Affected rows indicates how many rows of data are returned (get request) or updated (set request)

Example Request

```
<?xml version="1.0" encoding="utf-8" ?>
<SetHotelSurchargeRequest xmlns="http://xml.ycs.agoda.com">
  <Authentication APIKey="00000000-0000-0000-0000-000000000000" HotelID="123456"/>
  <HotelSurchargesList>
    <HotelSurcharge>
      <RoomType RoomTypeID="12345" RatePlanID="22"/>
      <DateRange Type="Stay" Start="2012-01-01" End="2012-01-31"/>
      <SurchargePolicy>
        <SurchargeTypeID>97</SurchargeTypeID>
        <IsCommissionable>false</IsCommissionable>
        <Currency>THB</Currency>
        <Amount>100.00</Amount>
        <DayApply>0000110</DayApply>
      </SurchargePolicy>
    </HotelSurcharge>
    <HotelSurcharge>
      <RoomType RoomTypeID="67890" RatePlanID="22"/>
      <DateRange Type="Stay" Start="2012-01-01" End="2012-01-31"/>
      <SurchargePolicy>
        <SurchargeTypeID>127</SurchargeTypeID>
        <IsCommissionable>false</IsCommissionable>
        <Amount>10.00</Amount>
        <DayApply>0000110</DayApply>
      </SurchargePolicy>
    </HotelSurcharge>
  </HotelSurchargesList>
</SetHotelSurchargeRequest>
```

Example Response

```
<?xml version="1.0" encoding="utf-8" ?>
<SetHotelSurchargeResponse xmlns="http://xml.ycs.agoda.com">
  <StatusResponse Status="200" AffectedRows="2" />
</SetHotelSurchargeResponse>
```

SetHotelValueAdds

Description

This API is used to create or update the Hotel's Value Add package by Rooms, Rate Plans and Dates.

XML Elements and Attributes

SetHotelValueAdds RS					
Level	Name	Type	Occurs	Required	Comments
0	SetHotelValueAddsRequest		1	Y	
1	Authentication				
	@APIKey	String Length 32	1	Y	This is the primary authentication for connection to the Agoda API
	@HotelID	String Length 1-8	1	Y	Every request must contain a valid hotel ID that partner has access to
1	HotelValueAddsList		1	Y	Contains each hotel value add
2	ValueAdd		1-n	Y	
3	RoomType				
	@RoomTypeID	Numeric Length 1-8	1	Y	Room Type ID linked to surcharge. Note that RoomTypeID = 0 (zero) is for all room types
	@RatePlanID	Numeric Length 1-4	1	Y	Rate Plan ID linked to surcharge
3	DateRange		1	Y	Date Range information for request
	@Type	Alpha	1	Y	Type of date range. There is only one option currently which is Stay
	@Start	Date YYYY-MM-DD	1	Y	The inclusive start date for the query
	@End	Date YYYY-MM-DD	1	Y	The inclusive end date for the query
3	Currency	Alpha Length 3	1-n	Y	Lists currency for this surcharge. A list of all currencies with descriptions are in Appendix E
3	TotalPackageValue	Numeric nnn.nn	1	Y	Total cost of value add
3	ApplyDaily	Boolean	1	Y	Denotes if surcharge is applied daily to to entire stay
4	ValueAddItemList		1	Y	Each Value Add can have multiple items. Item List will list each item that is contained in value add
5	ItemID	Alpha	1-n	Y	
	@ValueAddItemID	Numeric Length 1-8	1-2	Y	Agoda ID for value add. Use GetAllValueAdds request for full list of Value Adds

SetHotelValueAdds RS					
Level	Name	Type	Occurs	Required	Comments
0	SetHotelValueAdds		1	Y	
1	StatusResponse		1	Y	For further information on status responses see Response Codes and for code reference see Appendix D

	@Status	Numeric Length 3	1	Y	
	@AffectedRows	Numeric Length 1 - 4	1	Y	Affected rows indicates how many rows of data are returned (get request) or updated (set request)

Example Request

```
<?xml version="1.0" encoding="utf-8" ?>
<SetHotelValueAddsRequest xmlns="http://xml.ycs.agoda.com">
  <Authentication APIKey="00000000-0000-0000-0000-000000000000" HotelID="123456"/>
  <HotelValueAddsList>
    <ValueAdd>
      <RoomType RoomTypeID="12345" RatePlanID="22"/>
      <DateRange Type="Stay" Start="2012-01-01" End="2012-01-31"/>
      <Currency>USD</Currency>
      <TotalPackageValue>200</TotalPackageValue>
      <ApplyDaily>true</ApplyDaily>
      <ValueAddItemList>
        <ItemID ValueAddItemID="1"/>
        <ItemID ValueAddItemID="2"/>
        <ItemID ValueAddItemID="3"/>
      </ValueAddItemList>
    </ValueAdd>
  </HotelValueAddsList>
</SetHotelValueAddsRequest>
```

Example Response

```
<?xml version="1.0" encoding="utf-8" ?>
<SetHotelValueAddsResponse xmlns="http://xml.ycs.agoda.com">
  <StatusResponse Status="200" AffectedRows="1" />
</SetHotelValueAddsResponse>
```

Booking API's

GetBookingList

Description

This API is used to retrieve the Hotel's bookings from Agoda

XML Elements and Attributes

GetBookingList RQ					
Level	Name	Type	Occurs	Required	Comments
0	GetBookingListRequest		1	Y	
1	Authentication				
	@APIKey	String Length 32	1	Y	This is the primary authentication for connection to the Agoda API
	@HotelID	String Length 1-8	1	Y	Every request must contain a valid hotel ID that partner has access to
1	DateRange		1	Y	Date Range information for request
	@Type	Alpha	1	Y	Type of date range for booking request. There are two options for request: 'Stay' Date: To search for bookings by customer hotel arrival and departure 'Book' Date: To search for bookings by customer hotel booking start and end dates
	@Start	Date YYYY-MM-DD	1	Y	The inclusive start date for the booking query
	@End	Date YYYY-MM-DD	1	Y	The inclusive end date for the booking query
1	RequestType	Alpha	1	N	Denotes the type of bookings to query. Options are: Confirmed Amended Cancelled
1	RequestStatus	Alpha	1	N	Denotes the status of bookings to query. Options are: Acknowledgement Pending Acknowledged

GetBookingList RS					
Level	Name	Type	Occurs	Required	Comments
0	GetBookingListResponse		1	Y	
1	StatusResponse		1	Y	For further information on status responses see Response Codes and for code reference see Appendix D
	@Status	Numeric Length 3	1	Y	
	@AffectedRows	Numeric Length 1 - 4	1	Y	Affected rows indicates how many rows of data are returned (get

					request) or updated (set request)
1	BookingList		1	Y	Container that stores all hotel bookings
2	Booking		1-n	Y	Lists all booking information within
3	HotelID		1	Y	Agoda Hotel ID for booking
3	AckRequestID		1	Y	
3	BookingID	Numeric Length 1-15	1	Y	Agoda Booking ID for this booking. This must be used when sending GetBookingDetails request
3	BookingDate	Date YYYY-MM-DD	1	Y	Date that booking was created on Agoda YCS4
3	AckID	Alpha Numeric	1	Y	Acknowledgement ID. This is entered manually or via email link by hotel. This is not a mandatory requirement for hotels and booking is still confirmed even if no AckID
3	AckBy	Alpha	1	Y	Denotes name of user that manually acknowledged booking. If acknowledged within YCS4 this will show user login name. If acknowledged by email this will show 'AutoAcknowledged'
3	AckDate	Date YYYY-MM-DD	1	Y	Date that booking was acknowledged on Agoda YCS4
3	GuestName	Alpha	1	Y	The full name (First & Last name) of the guest for the booking
3	DateRange		1	Y	Date Range information for request
	@Type	Alpha	1	Y	Type of date range for booking request set to 'Stay' Date
	@Start	Date YYYY-MM-DD	1	Y	The inclusive start date for the booking
	@End	Date YYYY-MM-DD	1	Y	The inclusive end date for the booking
3	StatusChangeDate	Date YYYY-MM-DD	1	Y	The last date that the booking was changed/amended. If booking is amended multiple times only the last amended date is listed here
3	RatePlan	Numeric Length 1-3	1	Y	The Agoda Rate Plan that is listed against the booking. Use GetHotelRatePlans to get all hotel rate plans
3	PaymentModel				Payment Model for booking 1 – Merchant 2 – Merchant Commission See GetBookingList description

Example Request

Coming Soon

Example Response

Coming Soon

GetBookingDetails

Description

This API is used to retrieve the Hotel's bookings from Agoda

XML Elements and Attributes

GetBookingDetails RQ					
Level	Name	Type	Occurs	Required	Comments
0	GetBookingDetailsRequest		1	Y	
1	Authentication				
	@APIKey	String Length 32	1	Y	This is the primary authentication for connection to the Agoda API
	@HotelID	String Length 1-8	1	Y	Every request must contain a valid hotel ID that partner has access to
1	BookingIDList			Y	Container for all requested Booking IDs
2	BookingID	Numeric Length 1-15	1-25	Y	List of each Booking ID requested for Booking Details response. To get hotel Booking IDs use GetBookingList request

GetBookingDetails RS					
Level	Name	Type	Occurs	Required	Comments
0	GetBookingDetailsRequest		1	Y	
1	StatusResponse		1	Y	For further information on status responses see Response Codes and for code reference see Appendix D
	@Status	Numeric Length 3	1	Y	
	@AffectedRows	Numeric Length 1 - 4	1	Y	Affected rows indicates how many rows of data are returned (get request) or updated (set request)
1	BookingDetailData		1-25	Y	Lists all booking detailed information within
3	BookingID	Numeric Length 1-15	1	Y	Agoda Booking ID for this booking. This must be used when sending GetBookingDetails request
3	BookingDate	Date YYYY-MM-DD	1	Y	Date that booking was created on Agoda YCS4
3	HotelID		1	Y	Agoda Hotel ID for booking
3	Hotel	Alpha	1	Y	Full Name of Hotel as it is listed in Agoda YCS4
3	RoomType				
	@RoomTypeID	Numeric Length 1-8	1	Y	Room Type ID for booking. Room Type information can be found using GetHotelRoomTypes request
	@RatePlanID	Numeric Length 1-4	1	Y	Rate Plan ID linked to surcharge. Rate Plan information can be found using GetHotelRatePlans request
3	AckType	Alpha	1	Y	Type of Acknowledgement
3	AckStatus	Alpha	1	Y	Status of Acknowledgement
3	AckRequestID		1	Y	Acknowledgement ID

3	DateRange		1	Y	Date Range information for request
	@Type	Alpha	1	Y	Type of date range for booking request set to 'Stay' Date
	@Start	Date YYYY-MM-DD	1	Y	The inclusive start date for the booking
	@End	Date YYYY-MM-DD	1	Y	The inclusive end date for the booking
3	NoOfRoom	Numeric Length 1	1	Y	Number of rooms booked for this booking. Between 1 and 9 rooms can be booked on Agoda in a single booking
3	GuestCount		1	Y	Container for number of guests information
	@Adult	Numeric Length 1-2	1	Y	Number of Adults booked in booking. If multiple rooms are booked this number will be total adults for all bookings
	@Children	Numeric Length 1-2	1	Y	Number of Children booked in booking. If multiple rooms are booked this number will be total adults for all bookings
3	ExtraBed	Numeric Length 1-2	1	Y	Lists the number of extra beds that are booked in the booking. Any extra bed costs will be listed in the DailyRateDate element as ExtraBedRate
3	BookingID	Numeric Length 1-15	1	Y	Agoda Booking ID for this booking. This must be used when sending GetBookingDetails request
3	BreakfastIncluded	Boolean	1	Y	Denotes if breakfast is included in this booking
3	Promotion		1	Y	<i>Future Functionality does not list any information currently</i>
3	Rates		1	Y	Container for all booking rate/price information for booking
	@currency	Alpha Length 3	1	Y	Lists currency for this booking. A list of all currencies with descriptions are in Appendix E
	@exclusive	Numeric nnn.nn	1	Y	Total cost of booking excluding Agoda commission and excluding all taxes and service fees
	@tax	Numeric nnn.nn	1	Y	Total amount of tax for booking
	@fees	Numeric nnn.nn	1	Y	Total amount of fees for booking
	@inclusive	Numeric nnn.nn	1	Y	Total cost of booking excluding Agoda commission and including all taxes and service fees
4	DailyRateDate		1-n	Y	
	@type	Alpha	1	Y	Type of daily rate detail. Types are: RoomRate: Base room rate ExtraBedRate: Extra bed rate Surcharge: Total Surcharges rates
	@date	Date YYYY-MM-DD	1	Y	Date for rate detail
	@exclusive	Numeric nnn.nn	1	Y	Daily cost of booking excluding Agoda commission and excluding all taxes and service fees
	@tax	Numeric nnn.nn	1	Y	Daily amount of tax for booking
	@fees	Numeric nnn.nn	1	Y	Daily amount of fees for booking

	@inclusive	Numeric nnn.nn	1	Y	Daily cost of booking excluding Agoda commission and including all taxes and service fees
3	IATAID	Numeric	1	Y	Lists Agoda IATA number
3	Payment		1	Y	Container with payment information
4	CreditCardType	Alpha	1	Y	Credit card type for booking – will always show as Mastercard for all bookings as this is the Agoda card type for UPC card. See GetBookingDetails description
4	CreditCardNumber	Numeric Length 16	1	Y	Credit card number for booking
4	CreditCardName	Alpha	1	Y	Name on credit card. For all bookings will show 'Agoda Pte Ltd.'
4	CreditCardExpiry	Date MM/YYYY	1	Y	Credit card expiry date
4	CreditCardCCV	Numeric	1	Y	Credit card CCV number
3	Guests		19	Y	Container for Guest information
4	Guest		1	Y	Guest Details
5	Name	Alpha	1	Y	The full name (First & Last name) of the guest for the booking
5	Nationality	Alpha Length 2	1	Y	Nationality of Guest. A list of all nationalities with descriptions are in Appendix E
3	SpecialRequests	Alphas	1	Y	Free text box for guest to send additional request information

Example Request

```
<?xml version="1.0" encoding="UTF-8"?>
<GetBookingDetailsRequest xmlns="http://xml.ycs.agoda.com">
  <Authentication APIKey="00000000-0000-0000-0000-000000000000" HotelID="123456" />
  <BookingIDList>
    <BookingID>123456</BookingID>
    <BookingID>123456</BookingID>
    <BookingID>123456</BookingID>
    <BookingID>123456</BookingID>
    <BookingID>123456</BookingID>
  </BookingIDList>
</GetBookingDetailsRequest>
```

Example Response

```

<?xml version="1.0" encoding="UTF-8"?>
<GetBookingDetailsResponse xmlns="http://xml.ycs.agoda.com">
  <StatusResponse Status="200" AffectedRows="1">
    <ErrorItem ErrorID="201" ItemID="1" ErrorText="This is an example error item that can be included
in the ProcessResult"/>
    <ErrorItem ErrorID="201" ItemID="2" ErrorText="This is an example error item that can be included
in the ProcessResult"/>
  </StatusResponse>
  <BookingDetailData>
    <BookingID>3600690</BookingID>
    <BookingDate>2012-03-15</BookingDate>
    <HotelID>71656</HotelID>
    <Hotel>X Hotel</Hotel>
    <Roomtype RoomTypeID="149722" RatePlanID="1">Deluxe</Roomtype>
    <AckType>ConfirmBooking</AckType>
    <AckStatus>Acknowledged</AckStatus>
    <DateRange Type="Stay" Start="2012-04-01" End="2012-04-04"/>
    <NoOfRoom>2</NoOfRoom>
    <GuestCount Adult="2" Children="0"/>
    <ExtraBed>0</ExtraBed>
    <BreakfastIncluded>N</BreakfastIncluded>
    <Promotion/>
    <Rates currency="THB" exclusive="120.00" tax="12.00" fees="55.00" inclusive="167.00">
      <DailyRates>
        <DailyRateDate type="RoomRate" date="2012-08-21" exclusive="120.00" tax="12.00" fees="55.00"
inclusive="167.00"/>
        <DailyRateDate type="ExtraBedRate" date="2012-08-22" exclusive="120.00" tax="12.00"
fees="55.00" inclusive="167.00"/>
        <DailyRateDate type="Surcharge" date="2012-08-22" exclusive="120.00" tax="12.00" fees="55.00"
inclusive="167.00"/>
      </DailyRates>
    </Rates>
    <IATAID>#96-6 2709 0</IATAID>
    <Payment>
      <CreditCardType>Mastercard</CreditCardType>
      <CreditcardNumber>000000000000</CreditcardNumber>
      <CreditCardName>Name on the card</CreditCardName>
      <CreditCardExpiry>10/2012</CreditCardExpiry>
      <CreditCardCCV>189</CreditCardCCV>
    </Payment>
    <Guests>
      <Guest>
        <Name>Abc</Name>
        <Nationality>CN</Nationality>
      </Guest>
    </Guests>
    <SpecialRequests>I want a bigger room</SpecialRequests>
  </BookingDetailData>
</GetBookingDetailsResponse>

```

Appendix A

YCS Main Schema

Please find below YCS Main Schema used in this document

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema id="XMLSchema1" targetNamespace="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd"
elementFormDefault="qualified" xmlns="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd"
xmlns:mstns="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd" xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <!-- Authentication Objects -->
  <xs:element name="Authentication">
    <xs:complexType>
      <xs:attribute name="APIKey" type="xs:string" use="required" />
      <xs:attribute name="HotelID" type="xs:integer" use="required" />
    </xs:complexType>
  </xs:element>

  <!-- Status Response Objects-->
  <xs:element name="StatusResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="unbounded" name="ErrorItem">
          <xs:complexType>
            <xs:attribute name="ErrorID" type="xs:integer" use="required" />
            <xs:attribute name="ItemID" type="xs:integer" use="required" />
            <xs:attribute name="ErrorText" type="xs:string" use="required" />
          </xs:complexType>
        </xs:element>
      </xs:sequence>
      <xs:attribute name="Status" type="xs:integer" use="required" />
      <xs:attribute name="AffectedRows" type="xs:integer" use="required" />
    </xs:complexType>
  </xs:element>

  <!-- Standard Objects -->
  <xs:element name="DateRange">
    <xs:complexType>
      <xs:attribute name="Type" type="xs:string" use="required" />
      <xs:attribute name="Start" type="xs:date" use="required" />
      <xs:attribute name="End" type="xs:date" use="optional" />
    </xs:complexType>
  </xs:element>
  <xs:element name="RoomType">
    <xs:complexType>
      <xs:simpleContent>
        <xs:extension base="xs:string">
          <xs:attribute name="RoomTypeID" type="xs:integer" use="required" />
          <xs:attribute name="RatePlan" type="xs:integer" use="required" />
          <xs:attribute name="Language" type="xs:string" use="optional" />
        </xs:extension>
      </xs:simpleContent>
    </xs:complexType>
  </xs:element>
```

```

<xs:element name="RequestedLanguage" type="xs:string" />

<!-- Currency Objects -->
<xs:element name="CurrencyList">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="Currency"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="Currency">
  <xs:complexType>
    <xs:attribute name="code" type="xs:string" use="required" />
    <xs:attribute name="ExchangeRate" type="xs:decimal" use="optional" />
  </xs:complexType>
</xs:element>

<!-- Cancellation Objects -->
<xs:element name="CancellationPolicyList">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="CancellationPolicy"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="CancellationPolicy">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="CancellationPolicyID" type="xs:integer" minOccurs="1" maxOccurs="1" />
      <xs:element name="PolicyParameters">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="unbounded" name="PolicyParameter"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="PolicyText" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="PolicyText">
  <xs:complexType>
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute name="language" type="xs:string" use="required" />
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
</xs:element>
<xs:element name="PolicyParameter">
  <xs:complexType>
    <xs:simpleContent>
      <xs:extension base="xs:unsignedByte">
        <xs:attribute name="days" type="xs:integer" use="required" />
        <xs:attribute name="charge" type="xs:string" use="required" />
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>

```

```

</xs:element>

<!-- Guest Objects -->
<xs:element name="GuestCount">
  <xs:complexType>
    <xs:attribute name="Adult" type="xs:integer" use="required" />
    <xs:attribute name="Children" type="xs:integer" use="required" />
  </xs:complexType>
</xs:element>
<xs:element name="Guests">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Guest" minOccurs="1" maxOccurs="unbounded">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="Name" type="xs:string" />
            <xs:element name="Nationality" type="xs:string" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<!-- Rate Objects -->
<xs:element name="Rates">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="DailyRates" minOccurs="1" maxOccurs="unbounded">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="1" maxOccurs="unbounded" name="DailyRateDate">
              <xs:complexType>
                <xs:attribute name="date" type="xs:date" use="required" />
                <xs:attribute name="type" type="xs:string" use="required" />
                <xs:attribute name="exclusive" type="xs:decimal" use="required" />
                <xs:attribute name="tax" type="xs:decimal" use="required" />
                <xs:attribute name="fees" type="xs:decimal" use="required" />
                <xs:attribute name="inclusive" type="xs:decimal" use="required" />
              </xs:complexType>
            </xs:element>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="currency" type="xs:string" use="required" />
    <xs:attribute name="exclusive" type="xs:decimal" use="required" />
    <xs:attribute name="tax" type="xs:decimal" use="required" />
    <xs:attribute name="fees" type="xs:decimal" use="required" />
    <xs:attribute name="inclusive" type="xs:decimal" use="required" />
  </xs:complexType>
</xs:element>

<!-- Rate Plan Objects -->
<xs:element name="RatePlanList">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="1" maxOccurs="unbounded" name="RatePlan"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

```

</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="RatePlan">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="ID" type="xs:integer" minOccurs="1" maxOccurs="1"/>
      <xs:element name="Name" minOccurs="1" maxOccurs="1">
        <xs:complexType>
          <xs:simpleContent>
            <xs:extension base="xs:string">
              <xs:attribute name="language" type="xs:string" use="required" />
            </xs:extension>
          </xs:simpleContent>
        </xs:complexType>
      </xs:element>
      <xs:element name="RateType" type="xs:string" minOccurs="1" maxOccurs="1"/>
      <xs:element name="Currency" type="xs:string" minOccurs="1" maxOccurs="1"/>
      <xs:element name="MinRate" type="xs:decimal" minOccurs="0" maxOccurs="1"/>
      <xs:element name="MaxRate" type="xs:decimal" minOccurs="0" maxOccurs="1"/>
      <xs:element name="MinNightStay" type="xs:integer" minOccurs="0" maxOccurs="1"/>
      <xs:element name="MaxNightStay" type="xs:integer" minOccurs="0" maxOccurs="1"/>
      <xs:element name="MaxAllotment" type="xs:integer" minOccurs="0" maxOccurs="1"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<!-- Payment Objects -->
<xs:element name="Payment">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="CreditCardType" type="xs:string" minOccurs="1" maxOccurs="1"/>
      <xs:element name="CreditcardNumber" type="xs:string" minOccurs="1" maxOccurs="1"/>
      <xs:element name="CreditCardName" type="xs:string" minOccurs="1" maxOccurs="1"/>
      <xs:element name="CreditCardExpiry" type="xs:string" minOccurs="1" maxOccurs="1"/>
      <xs:element name="CreditCardCCV" type="xs:integer" minOccurs="1" maxOccurs="1"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<!-- Promotion Objects -->
<xs:element name="PromotionID" type="xs:integer" />
<xs:element name="HotelPromotionList">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="HotelPromotion"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="HotelPromotion">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="PromotionConditions"/>
      <xs:element name="PromotionBenefit"/>
      <xs:element name="PromotionRestrictions"/>
    </xs:sequence>
    <xs:attribute name="PromotionID" type="xs:integer" use="optional" />
  </xs:complexType>

```

```

</xs:element>
<xs:element name="PromotionConditions">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="RoomTypeList" minOccurs="1" maxOccurs="1">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="1" maxOccurs="unbounded" name="RoomType"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="MinAdvPurchase" type="xs:integer" minOccurs="0" maxOccurs="1"/>
      <xs:element name="MaxAdvPurchase" type="xs:integer" minOccurs="0" maxOccurs="1"/>
      <xs:element name="BookOn" type="xs:string" minOccurs="0" maxOccurs="1"/>
      <xs:element name="StayOn" type="xs:string" minOccurs="0" maxOccurs="1"/>
      <xs:element name="CheckInOn" type="xs:string" minOccurs="0" maxOccurs="1"/>
      <xs:element name="MinNightsStay" type="xs:integer" minOccurs="0" maxOccurs="1"/>
      <xs:element name="MaxNightsStay" type="xs:integer" minOccurs="0" maxOccurs="1"/>
      <xs:element name="MinRooms" type="xs:integer" minOccurs="0" maxOccurs="1"/>
      <xs:element name="BookDate" minOccurs="0" maxOccurs="1">
        <xs:complexType>
          <xs:attribute name="Start" type="xs:date" use="required" />
          <xs:attribute name="End" type="xs:date" use="required" />
        </xs:complexType>
      </xs:element>
      <xs:element name="BookTime" minOccurs="0" maxOccurs="1">
        <xs:complexType>
          <xs:attribute name="Start" type="xs:time" use="required" />
          <xs:attribute name="End" type="xs:time" use="required" />
        </xs:complexType>
      </xs:element>
      <xs:element name="StayDate" minOccurs="0" maxOccurs="1">
        <xs:complexType>
          <xs:attribute name="Start" type="xs:date" use="required" />
          <xs:attribute name="End" type="xs:date" use="required" />
        </xs:complexType>
      </xs:element>
      <xs:element name="WebSiteList" minOccurs="0" maxOccurs="1">
        <xs:complexType>
          <xs:sequence>
            <xs:element maxOccurs="unbounded" minOccurs="1" name="Website">
              <xs:complexType>
                <xs:simpleContent>
                  <xs:extension base="xs:string">
                    <xs:attribute name="ID" type="xs:unsignedByte" use="required" />
                  </xs:extension>
                </xs:simpleContent>
              </xs:complexType>
            </xs:element>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="PromotionBenefit">
  <xs:complexType>
    <xs:sequence>

```

```

<xs:element name="DiscountType" type="xs:string" minOccurs="1" maxOccurs="1" />
<xs:element name="Value1" type="xs:decimal" minOccurs="1" maxOccurs="1" />
<xs:element name="Value2" type="xs:decimal" minOccurs="0" maxOccurs="1" />
<xs:element name="Value3" type="xs:decimal" minOccurs="0" maxOccurs="1" />
<xs:element name="Value4" type="xs:decimal" minOccurs="0" maxOccurs="1" />
<xs:element name="Value5" type="xs:decimal" minOccurs="0" maxOccurs="1" />
<xs:element name="Value6" type="xs:decimal" minOccurs="0" maxOccurs="1" />
<xs:element name="Value7" type="xs:decimal" minOccurs="0" maxOccurs="1" />
<xs:element name="RewardPointMultiplier" type="xs:integer" minOccurs="1" maxOccurs="1" />
<xs:element name="IsRecurringBenefit" type="xs:boolean" minOccurs="1" maxOccurs="1" />
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="PromotionRestrictions">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="CancellationPolicy" minOccurs="1" maxOccurs="1" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<!-- Surcharge Objects -->
<xs:element name="SurchargeID" type="xs:integer" />
<xs:element name="SurchargePolicy">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="SurchargeID" type="xs:integer" minOccurs="0" maxOccurs="1"/>
      <xs:element name="SurchargeTypeID" type="xs:integer" minOccurs="0" maxOccurs="1"/>
      <xs:element name="Currency" type="xs:string" minOccurs="1" maxOccurs="1"/>
      <xs:element name="Amount" type="xs:decimal" minOccurs="1" maxOccurs="1"/>
      <xs:element name="IsPercent" type="xs:boolean" minOccurs="1" maxOccurs="1"/>
      <xs:element name="Description" minOccurs="1" maxOccurs="1">
        <xs:complexType>
          <xs:simpleContent>
            <xs:extension base="xs:string">
              <xs:attribute name="language" type="xs:string" use="required" />
            </xs:extension>
          </xs:simpleContent>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:element>
<xs:element name="SurchargeType">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="SurchargeTypeID" type="xs:unsignedByte" />
      <xs:element name="SurchargeApplyType" type="xs:string" />
      <xs:element name="SurchargeName">
        <xs:complexType>
          <xs:simpleContent>
            <xs:extension base="xs:string">
              <xs:attribute name="language" type="xs:string" use="required" />
            </xs:extension>
          </xs:simpleContent>
        </xs:complexType>
      </xs:element>
      <xs:element name="SurchargeDescription">

```



```

<xs:complexType>
  <xs:simpleContent>
    <xs:extension base="xs:string">
      <xs:attribute name="language" type="xs:string" use="required" />
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
</xs:element>
<xs:element name="ChargeTypeCode" type="xs:string" />
<xs:element name="IsPercent" type="xs:string" />
<xs:element name="IsCommissionable" type="xs:string" />
</xs:sequence>
</xs:complexType>
</xs:element>

<!-- Value Add Objects-->
<xs:element name="ValueAddItemList">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="ValueAddItem"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ValueAddItem">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="ValueAddItemID" type="xs:integer" minOccurs="1" maxOccurs="1"/>
      <xs:element name="ValueAddItemName" minOccurs="1" maxOccurs="1">
        <xs:complexType>
          <xs:simpleContent>
            <xs:extension base="xs:string">
              <xs:attribute name="language" type="xs:string" use="required" />
            </xs:extension>
          </xs:simpleContent>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<!-- Website Objects -->
<xs:element name="WebsiteList">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="Website">
        <xs:complexType>
          <xs:simpleContent>
            <xs:extension base="xs:string">
              <xs:attribute name="ID" type="xs:integer" use="required"/>
            </xs:extension>
          </xs:simpleContent>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<!-- Hotel Objects -->

```

```

<xs:element name="HotelRoomTypesList">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="RoomType"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="HotelValueAddsList">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="ValueAdd" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="RoomType" minOccurs="1" maxOccurs="1"/>
            <xs:element name="DateRange" minOccurs="1" maxOccurs="1"/>
            <xs:element name="Currency" type="xs:string" minOccurs="1" maxOccurs="1"/>
            <xs:element name="PackageWorth" type="xs:decimal" minOccurs="1" maxOccurs="1"/>
            <xs:element name="ApplyPerDay" type="xs:string" minOccurs="1" maxOccurs="1"/>
            <xs:element name="ValueAddItemList" minOccurs="1" maxOccurs="1">
              <xs:complexType>
                <xs:sequence>
                  <xs:element minOccurs="0" maxOccurs="unbounded" name="ItemID">
                    <xs:complexType>
                      <xs:simpleContent>
                        <xs:extension base="xs:string">
                          <xs:attribute name="ID" type="xs:integer" use="required" />
                        </xs:extension>
                      </xs:simpleContent>
                    </xs:complexType>
                  </xs:element>
                </xs:sequence>
              </xs:complexType>
            </xs:element>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="HotelCancellationPolicyList">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="HotelCancellationPolicy" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="HotelCancellationPolicy">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="RoomType" minOccurs="1" maxOccurs="1"/>
      <xs:element name="DateRange" minOccurs="1" maxOccurs="1"/>
      <xs:element name="CancellationPolicy" minOccurs="1" maxOccurs="1"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="HotelSurchargesList">
  <xs:complexType>
    <xs:sequence>

```

```

<xs:element maxOccurs="unbounded" name="HotelSurcharge">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="RoomType" minOccurs="1" maxOccurs="1"/>
      <xs:element name="DateRange" minOccurs="1" maxOccurs="1"/>
      <xs:element name="SurchargePolicy" minOccurs="1" maxOccurs="1"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>

<!-- Hotel Inventory Objects -->
<xs:element name="HotelInventoryList">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="1" maxOccurs="unbounded" name="HotelInventory"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="HotelInventory">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="RoomType" minOccurs="1" maxOccurs="1"/>
      <xs:element name="DateRange" minOccurs="1" maxOccurs="1"/>
      <xs:element name="InventoryRate" minOccurs="0" maxOccurs="1"/>
      <xs:element name="InventoryAllotment" minOccurs="0" maxOccurs="1"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="InventoryRate">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="SingleRate" type="xs:decimal" minOccurs="0" maxOccurs="1"/>
      <xs:element name="DoubleRate" type="xs:decimal" minOccurs="0" maxOccurs="1"/>
      <xs:element name="FullRate" type="xs:decimal" minOccurs="0" maxOccurs="1"/>
      <xs:element name="ExtraGuest" type="xs:decimal" minOccurs="0" maxOccurs="1"/>
      <xs:element name="ExtraAdult" type="xs:decimal" minOccurs="0" maxOccurs="1"/>
      <xs:element name="ExtraChild" type="xs:decimal" minOccurs="0" maxOccurs="1"/>
      <xs:element name="ExtraBed" type="xs:decimal" minOccurs="0" maxOccurs="1"/>
      <xs:element name="Breakfast" type="xs:boolean" minOccurs="0" maxOccurs="1"/>
      <xs:element name="PromotionBlackout" type="xs:boolean" minOccurs="0" maxOccurs="1"/>
      <xs:element name="MinNightStay" type="xs:integer" minOccurs="0" maxOccurs="1"/>
      <xs:element name="MaxNightStay" type="xs:integer" minOccurs="0" maxOccurs="1"/>
    </xs:sequence>
    <xs:attribute name="Currency" type="xs:string" use="required" />
  </xs:complexType>
</xs:element>
<xs:element name="InventoryAllotment">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="RegularAllotment" type="xs:integer" minOccurs="0" maxOccurs="1"/>
      <xs:element name="RegularAllotmentUsed" type="xs:integer" minOccurs="0" maxOccurs="1"/>
      <xs:element name="GuarenteedAllotment" type="xs:integer" minOccurs="0" maxOccurs="1"/>
      <xs:element name="GuarenteedAllotmentUsed" type="xs:integer" minOccurs="0" maxOccurs="1"/>
      <xs:element name="AutoTopUp" type="xs:integer" minOccurs="0" maxOccurs="1"/>
      <xs:element name="AutoTopUpUsed" type="xs:integer" minOccurs="0" maxOccurs="1"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

```

<xs:element name="CutOffDayNormal" type="xs:integer" minOccurs="0" maxOccurs="1"/>
<xs:element name="CutOffDayGuarenteed" type="xs:integer" minOccurs="0" maxOccurs="1"/>
<xs:element name="CloseOut" type="xs:boolean" minOccurs="0" maxOccurs="1"/>
<xs:element name="NoArrival" type="xs:boolean" minOccurs="0" maxOccurs="1"/>
<xs:element name="NoCheckOut" type="xs:boolean" minOccurs="0" maxOccurs="1"/>
</xs:sequence>
</xs:complexType>
</xs:element>

<!-- Rate matrix Objects -->
<xs:element name="HotelMatrixList">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="1" maxOccurs="unbounded" name="HotelMatrix"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="HotelMatrix">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="RoomType" minOccurs="1" maxOccurs="1"/>
      <xs:element name="DateRange" minOccurs="1" maxOccurs="1"/>
      <xs:element name="LengthOfStay" type="xs:unsignedByte" minOccurs="0" maxOccurs="1"/>
      <xs:element name="InventoryAllotment" minOccurs="0" maxOccurs="1"/>
      <xs:element name="OccupancyRate" minOccurs="0" maxOccurs="1"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="OccupancyRate">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Guests">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="AdultCount" type="xs:integer" />
            <xs:element minOccurs="0" maxOccurs="1" name="ChildCount" type="xs:integer" />
            <xs:element minOccurs="0" maxOccurs="1" name="Occupancy" type="xs:integer" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="Total_Rate" type="xs:decimal" minOccurs="1" maxOccurs="1"/>
      <xs:element name="Daily_Rate" type="xs:decimal" minOccurs="1" maxOccurs="1"/>
      <xs:element name="HasCloseOutFlag" type="xs:boolean" minOccurs="1" maxOccurs="1"/>
      <xs:element name="IsBreakfastIncluded" type="xs:boolean" minOccurs="1" maxOccurs="1"/>
      <xs:element name="PromotionType" type="xs:integer" minOccurs="0" maxOccurs="1"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<!-- Booking Objects-->
<xs:element name="Booking">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="HotelID" type="xs:integer" minOccurs="1" maxOccurs="1"/>
      <xs:element name="AckRequestID" type="xs:integer" minOccurs="1" maxOccurs="1" />
      <xs:element name="BookingID" type="xs:integer" minOccurs="1" maxOccurs="1" />
      <xs:element name="BookingDate" type="xs:date" minOccurs="1" maxOccurs="1" />
      <xs:element name="AckID" type="xs:integer" minOccurs="1" maxOccurs="1" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

```
<xs:element name="AckBy" type="xs:string" minOccurs="1" maxOccurs="1" />
<xs:element name="AckDate" type="xs:date" minOccurs="1" maxOccurs="1" />
<xs:element name="GuestName" type="xs:string" minOccurs="1" maxOccurs="1" />
<xs:element name="DateRange" minOccurs="1" maxOccurs="1">
  <xs:complexType>
    <xs:attribute name="Type" type="xs:string" use="required" />
    <xs:attribute name="from" type="xs:date" use="required" />
    <xs:attribute name="to" type="xs:date" use="required" />
  </xs:complexType>
</xs:element>
<xs:element name="StatusChangeDate" type="xs:string" minOccurs="1" maxOccurs="1"/>
<xs:element name="RatePlan" type="xs:integer" minOccurs="1" maxOccurs="1"/>
<xs:element name="PaymentModel" type="xs:integer" minOccurs="1" maxOccurs="1"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>
```

YCS XML Schemas

Please find below YCS XML schemas used in this document

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://xml.ycs.agoda.com" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd"
schemaLocation="D:/XML/schema/YcsMainSchema.xsd"/>
  <xs:element name="GetAllCxlPoliciesRequest">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Authentication"/>
        <xs:element name="RequestedLanguage" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://xml.ycs.agoda.com" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd"
schemaLocation="D:/XML/schema/YcsMainSchema.xsd"/>
  <xs:element name="GetAllCxlPoliciesResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="StatusResponse"/>
        <xs:element name="CancellationPolicyList"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://xml.ycs.agoda.com" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd"
schemaLocation="D:/XML/schema/YcsMainSchema.xsd"/>
  <xs:element name="GetHotelCxlPolicyRequest">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Authentication"/>
        <xs:element name="RoomType"/>
        <xs:element name="DateRange"/>
        <xs:element name="RequestedLanguage" type="xs:string" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://xml.ycs.agoda.com" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd"
schemaLocation="D:/XML/schema/YcsMainSchema.xsd"/>
  <xs:element name="GetHotelClxPolicyResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="StatusResponse"/>
        <xs:element name="HotelCancellationPolicyList"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://xml.ycs.agoda.com" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd"
schemaLocation="D:/XML/schema/YcsMainSchema.xsd"/>
  <xs:element name="GetAllLanguagesRequest">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Authentication"/>
        <xs:element name="RequestedLanguage" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://xml.ycs.agoda.com" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd"
schemaLocation="D:/XML/schema/YcsMainSchema.xsd"/>
  <xs:element name="GetAllLanguagesResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="StatusResponse"/>
        <xs:element name="LanguageList"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://xml.ycs.agoda.com" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd"
schemaLocation="D:/XML/schema/YcsMainSchema.xsd"/>
  <xs:element name="GetHotelRoomTypesRequest">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Authentication"/>
        <xs:element name="RequestedLanguage" type="xs:string" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://xml.ycs.agoda.com" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd"
schemaLocation="D:/XML/schema/YcsMainSchema.xsd"/>
  <xs:element name="GetHotelRoomTypesResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="StatusResponse"/>
        <xs:element name="HotelRoomTypesList"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://xml.ycs.agoda.com" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd"
schemaLocation="D:/XML/schema/YcsMainSchema.xsd"/>
  <xs:element name="GetHotelRoomSettingsRequest">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Authentication"/>
        <xs:element name="RequestedLanguage" type="xs:string" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://xml.ycs.agoda.com" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd"
schemaLocation="D:/XML/schema/YcsMainSchema.xsd"/>
  <xs:element name="GetHotelRoomSettingsResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="StatusResponse"/>
        <xs:element name="HotelRoomTypesList"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://xml.ycs.agoda.com" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd"
schemaLocation="D:/XML/schema/YcsMainSchema.xsd"/>
  <xs:element name="GetHotelRatePlansRequest">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Authentication"/>
        <xs:element name="RequestedLanguage" type="xs:string" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```



```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://xml.ycs.agoda.com" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd"
schemaLocation="D:/XML/schema/YcsMainSchema.xsd"/>
  <xs:element name="GetHotelRatePlansResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="StatusResponse"/>
        <xs:element name="RatePlanList"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://xml.ycs.agoda.com" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd"
schemaLocation="D:/XML/schema/YcsMainSchema.xsd"/>
  <xs:element name="GetHotelInventoryRequest">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Authentication" minOccurs="1" maxOccurs="1"/>
        <xs:element name="RoomType" minOccurs="1" maxOccurs="1"/>
        <xs:element name="DateRange" minOccurs="1" maxOccurs="1"/>
        <xs:element name="RequestedLanguage" type="xs:string" minOccurs="1" maxOccurs="1"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://xml.ycs.agoda.com" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd"
schemaLocation="D:/XML/schema/YcsMainSchema.xsd"/>
  <xs:element name="GetHotelInventoryResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="StatusResponse" minOccurs="1" maxOccurs="1"/>
        <xs:element name="HotelInventoryList" minOccurs="1" maxOccurs="1"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://xml.ycs.agoda.com" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="GetAllSurchargeTypesRequest">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Authentication"/>
        <xs:element name="RequestedLanguage" type="xs:string" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```

<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://xml.ycs.agoda.com" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="GetAllSurchargeTypesResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="StatusResponse"/>
        <xs:element name="SurchargeTypeList">
          <xs:complexType>
            <xs:sequence>
              <xs:element maxOccurs="unbounded" name="SurchargeType"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

```

<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://xml.ycs.agoda.com" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd"
schemaLocation="D:/XML/schema/YcsMainSchema.xsd"/>
  <xs:element name="GetHotelSurchargeResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Authentication"/>
        <xs:element name="DateRange"/>
        <xs:element name="RequestedLanguage" type="xs:string" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

```

<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://xml.ycs.agoda.com" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd"
schemaLocation="D:/XML/schema/YcsMainSchema.xsd"/>
  <xs:element name="GetHotelSurchargeResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="StatusResponse"/>
        <xs:element name="HotelSurchargesList"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

```

<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://xml.ycs.agoda.com" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd"
schemaLocation="D:/XML/schema/YcsMainSchema.xsd"/>
  <xs:element name="GetHotelValueAddRequest">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Authentication"/>
        <xs:element name="DateRange"/>
        <xs:element name="RequestedLanguage" type="xs:string" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

```

<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd"
schemaLocation="D:/XML/schema/YcsMainSchema.xsd"/>
  <xs:element name="GetHotelValueAddResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="StatusResponse"/>
        <xs:element name="HotelValueAddsList"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

```

<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://xml.ycs.agoda.com" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd"
schemaLocation="D:/XML/schema/YcsMainSchema.xsd"/>
  <xs:element name="GetAllValueAddsRequest">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Authentication"/>
        <xs:element name="RequestedLanguage" type="xs:string" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

```

<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://xml.ycs.agoda.com" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd"
schemaLocation="D:/XML/schema/YcsMainSchema.xsd"/>
  <xs:element name="GetAllValueAddsResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="StatusResponse"/>
        <xs:element name="ValueAddItemList"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://xml.ycs.agoda.com" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd"
schemaLocation="D:/XML/schema/YcsMainSchema.xsd"/>
  <xs:element name="SetHotelInventoryList">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Authentication" minOccurs="1" maxOccurs="1"/>
        <xs:element name="HotelInventoryList" minOccurs="1" maxOccurs="1"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://xml.ycs.agoda.com" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd"
schemaLocation="D:/XML/schema/YcsMainSchema.xsd"/>
  <xs:element name="SetHotelInventoryResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="StatusResponse" minOccurs="1" maxOccurs="1"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://xml.ycs.agoda.com" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd"
schemaLocation="D:/XML/schema/YcsMainSchema.xsd"/>
  <xs:element name="SetHotelCxlPolicyRequest">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Authentication"/>
        <xs:element name="HotelCancellationsList">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="HotelCancellationPolicy">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element name="RoomType"/>
                    <xs:element name="DateRange"/>
                    <xs:element name="CancellationPolicy"/>
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://xml.ycs.agoda.com" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd"
schemaLocation="D:/XML/schema/YcsMainSchema.xsd"/>
  <xs:element name="SetHotelClxPolicyResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="StatusResponse"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://xml.ycs.agoda.com" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd"
schemaLocation="D:/XML/schema/YcsMainSchema.xsd"/>
  <xs:element name="SetHotelSurchargeRequest">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Authentication"/>
        <xs:element name="HotelSurchargesList"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://xml.ycs.agoda.com" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd"
schemaLocation="D:/XML/schema/YcsMainSchema.xsd"/>
  <xs:element name="SetHotelSurchargeResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="StatusResponse"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://xml.ycs.agoda.com" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd"
schemaLocation="D:/XML/schema/YcsMainSchema.xsd"/>
  <xs:element name="SetHotelValueAddsRequest">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Authentication"/>
        <xs:element name="HotelValueAddsList"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://xml.ycs.agoda.com" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd"
schemaLocation="D:/XML/schema/YcsMainSchema.xsd"/>
  <xs:element name="SetHotelValueAddsResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="StatusResponse"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://xml.ycs.agoda.com" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd"
schemaLocation="D:/XML/schema/YcsMainSchema.xsd"/>
  <xs:element name="GetBookingDetailsRequest">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Authentication"/>
        <xs:element name="BookingIDList">
          <xs:complexType>
            <xs:sequence>
              <xs:element maxOccurs="25" name="BookingID" type="xs:int" />
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://xml.ycs.agoda.com" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://ycs.xml.agoda.com/schema/YcsMainSchema.xsd"
schemaLocation="D:/XML/schema/YcsMainSchema.xsd"/>
  <xs:element name="GetBookingDetailsResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="StatusResponse" minOccurs="1" maxOccurs="1"/>
        <xs:element name="BookingDetailData" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="BookingID" type="xs:unsignedInt" minOccurs="1" maxOccurs="1"/>
              <xs:element name="BookingDate" type="xs:date" minOccurs="1" maxOccurs="1"/>
              <xs:element name="HotelID" type="xs:unsignedInt" minOccurs="1" maxOccurs="1"/>
              <xs:element name="Hotel" type="xs:string" minOccurs="1" maxOccurs="1"/>
              <xs:element name="Roomtype" minOccurs="1" maxOccurs="1"/>
              <xs:element name="AckType" type="xs:string" minOccurs="1" maxOccurs="1"/>
              <xs:element name="AckStatus" type="xs:string" minOccurs="1" maxOccurs="1"/>
              <xs:element name="DateRange" minOccurs="1" maxOccurs="1"/>
              <xs:element name="NoOfRoom" type="xs:unsignedByte" minOccurs="1" maxOccurs="1"/>
              <xs:element name="GuestCount" minOccurs="1" maxOccurs="1"/>
              <xs:element name="ExtraBed" type="xs:unsignedByte" minOccurs="1" maxOccurs="1"/>
              <xs:element name="BreakfastIncluded" type="xs:string" minOccurs="1" maxOccurs="1"/>
              <xs:element name="Promotion" minOccurs="1" maxOccurs="1"/>
              <xs:element name="Rates" minOccurs="1" maxOccurs="1"/>
              <xs:element name="IATAID" type="xs:string" minOccurs="1" maxOccurs="1"/>
              <xs:element name="Payment" minOccurs="1" maxOccurs="1"/>
              <xs:element name="Guests" minOccurs="1" maxOccurs="1"/>
              <xs:element name="SpecialRequests" type="xs:string" minOccurs="1" maxOccurs="1"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Appendix C

Charge Type List

Charge Type Code	Calculations
PAPB	Per Adult Per Booking
PAPD	Per Adult Per Day
PAPN	Per Adult Per Night
PB	Per Booking
PCPB	Per Child Per Booking
PCPD	Per Child Per Day
PCPN	Per Child Per Night
PGPB	Per Guest Per Booking
PGPD	Per Guest Per Day
PGPN	Per Guest Per Night
PN	Per Night
PP	Per Person
PRPN	Per Room Per Night

Appendix D

Error Codes

Code	Description
200	Request has been processed successfully
204	The request does not return any valid data
401	Authorization failed due to a wrong API key, IP address violation or other authorization failure
403	You have violated our terms of use and exceed the quota of requests.
406	When a verification request cannot be confirmed
416	The validation of your request failed
500	We have encountered an unrecoverable problem in the processing of the API call
503	We are in an planned or ad-hoc maintenance window and the service is temporary unavailable

101	Invalid hotel please check your hotel ID
102	Date range - From must be less than date range - To
103	Invalid rate plan please check your rate plan ID
104	Invalid room type please check your room type ID
105	Invalid date value
106	RoomTypeID is required
107	Date range - From cannot be less than today date
108	Minimum NightStay must greater than 0
109	NetRate should be set
110	SellRate should be set
111	Double rate cannot less than single rate
112	Full rate cannot less than double rate
113	Single Rate cannot less than Minimum Room Rate ({minimum_room_rate})
114	Double Rate cannot less than Minimum Room Rate ({minimum_room_rate})
115	Full Rate cannot less than Minimum Room Rate ({minimum_room_rate})
116	Date ({date}), Allotment cannot less than contracted allotment ({allotment})
117	Cut off day cannot greater than maximum cut off day ({cutoffday})
118	Date ({date}), Minimum night stay cannot greater than minimum nights limit ({minimum_night_limit})
119	Please set available allotment
120	Cannot set CloseOutRegularAllotment on '{date}' the Guarantee Allotment is set
121	Cannot set ClosedToArrival on '{date}' the Guarantee Allotment is set
122	Hotel service error
123	Rate daily service error
124	Allotment service error
125	Queue manager error
126	Available Allotment must greater or equal to 0
127	Allotment cannot more than maximum allotment ({maximum_min_stay})

128	Maximum NightStay must greater than 0
129	Minimum NightStay must less than Maximum NightStay
130	AutoRefill is required
131	DateFrom is required
132	DateTo is required
133	Invalid rate type '{input_rate_type}', the hotel has configure for '{config_rate_type}'
134	Auto Refill cannot more than maximum auto refill (10)
135	Agoda Data service error
136	Extra Adult cannot less than Minimum Room Rate ({minimum_room_rate})
137	Extra Child cannot less than Minimum Room Rate ({minimum_room_rate})
138	Extra Bed cannot less than Minimum Room Rate ({minimum_room_rate})
139	Single Rate cannot greater than Maximum Room Rate ({maximum_room_rate})
140	Double Rate cannot greater than Maximum Room Rate ({maximum_room_rate})
141	Full Rate cannot greater than Maximum Room Rate ({maximum_room_rate})
142	Extra Adult cannot greater than Maximum Room Rate ({maximum_room_rate})
143	Extra Child cannot greater than Maximum Room Rate ({maximum_room_rate})
144	Extra Bed cannot greater than Maximum Room Rate ({maximum_room_rate})
145	Date ({date}), Maximum night stay cannot greater than maximum nights limit ({maximum_night_limit})
146	Cannot set double rate because max occupancy is {max_occupancy}
147	Cannot set full rate because max occupancy is {max_occupancy}
148	Cannot set full rate because invalid extra rate type ({extra_rate_type})
149	Cannot set extra adult because invalid extra rate type ({extra_rate_type})
150	Cannot set extra child because invalid extra rate type ({extra_rate_type})
151	Cannot set extra bed because max extra beds is ({max_extra_beds})
152	Cancellation service error
153	Security service error
154	Promotion service error - {0}
155	Invalid cancellation policy code
156	BrokenRules - {0}
157	Invalid Surcharge Rate - {0}
158	Room Occupancy service error = {0}
159	Value Add service error - {0}
160	Invalid RateMatrix - {0}
700	No data changed.
800	Unknow Exception.
900	XML API Exception: {0}

Appendix E

List of Country Names, Codes and Currencies

Country Name	Country Code ISO	Country Code ISO 2	Currency Code
Afghanistan	AFG	AF	USD
Albania	ALB	AL	EUR
Algeria	DZA	DZ	EUR
American Samoa	ASM	AS	EUR
Andorra	AND	AD	EUR
Angola	AGO	AO	USD
Anguilla	AIA	AI	USD
Antigua & Barbuda	ATG	AG	USD
Argentina	ARG	AR	ARS
Armenia	ARM	AM	USD
Aruba	ABW	AW	USD
Australia	AUS	AU	AUD
Austria	AUT	AT	EUR
Azerbaijan	AZE	AZ	USD
Bahamas	BHS	BS	USD
Bahrain	BHR	BH	USD
Bangladesh	BGD	BD	USD
Barbados	BRB	BB	USD
Belarus	BLR	BY	RUB
Belgium	BEL	BE	EUR
Belize	BLZ	BZ	USD
Benin	BEN	BJ	USD
Bermuda	BMU	BM	USD
Bhutan	BTN	BT	INR
Bolivia	BOL	BO	USD
Bonaire Sint Eustatius and Saba	BES	BQ	USD
Bosnia Herzegovina	BIH	BA	EUR
Botswana	BWA	BW	USD
Bouvet Island	BVT	BV	NOK
Brazil	BRA	BR	BRL
British Indian Ocean Territory	IOT	IO	USD
Brunei Darussalam	BRN	BN	USD
Bulgaria	BGR	BG	EUR
Burkina Faso	BFA	BF	USD
Burundi	BDI	BI	USD

Cambodia	KHM	KH	USD
Cameroon	CMR	CM	USD
Canada	CAN	CA	CAD
Cape Verde	CPV	CV	USD
Cayman Islands	CYM	KY	USD
Central African Republic	CAF	CF	USD
Chad	TCO	TD	USD
Chile	CHL	CL	USD
China	CHN	CN	CNY
Christmas Island	CXR	CX	AUD
Cocos (Keeling) Islands	CCK	CC	AUD
Colombia	COL	CO	COP
Comoros	COM	KM	USD
Congo	COG	CG	USD
Congo, Democratic Republic of the	COD	CD	USD
Cook Islands	COK	CK	NZD
Costa Rica	CRI	CR	USD
Cote D'Ivoire	CIV	CI	USD
Croatia	HRV	HR	EUR
Curacao	CUW	CW	USD
Cyprus	CYP	CY	EUR
Czech Republic	CZE	CZ	EUR
Denmark	DNK	DK	DKK
Djibouti	DJI	DJ	USD
Dominica	DMA	DM	USD
Dominican Republic	DOM	DO	USD
Ecuador	ECU	EC	USD
Egypt	EGY	EG	EGP
El Salvador	SLV	SV	USD
Equatorial Guinea	GNQ	GQ	USD
Eritrea	ERI	ER	USD
Estonia	EST	EE	USD
Ethiopia	ETH	ET	USD
Falkland Islands	FLK	FK	USD
Faroe Islands	FRO	FO	DKK
Fiji	FJI	FJ	FJD
Finland	FIN	FI	EUR
France	FRA	FR	EUR
French Guiana	GUF	GF	EUR
French Polynesia	PYF	PF	XPF
French Southern and Antarctic Lands	ATF	TF	EUR

French West Indies	ATF	TF	EUR
Gabon	GAB	GA	USD
Gambia	GMB	GM	USD
Georgia	GEO	GE	EUR
Germany	DEU	DE	EUR
Ghana	GHA	GH	USD
Gibraltar	GIB	GI	GBP
Greece	GRC	GR	EUR
Greenland	GRL	GL	DKK
Grenada	GRD	GD	USD
Guadeloupe	GLP	GP	EUR
Guam	GUM	GU	USD
Guatemala	GTM	GT	USD
Guinea Republic	GIN	GN	USD
Guinea-Bissau	GNB	GW	USD
Guyana	GUY	GY	USD
Haiti	HTI	HT	USD
Holy See (Vatican City)	VAT	VA	EUR
Honduras	HND	HN	USD
Hong Kong	HKG	HK	HKD
Hungary	HUN	HU	EUR
Iceland	ISL	IS	USD
India	IND	IN	INR
Indonesia	IDN	ID	IDR
Iraq	IRQ	IQ	USD
Ireland	IRL	IE	EUR
Israel	ISR	IL	ILS
Italy	ITA	IT	EUR
Jamaica	JAM	JM	USD
Japan	JPN	JP	JPY
Jordan	JOR	JO	USD
Kazakhstan	KAZ	KZ	USD
Kenya	KEN	KE	USD
Kiribati	KIR	KI	AUD
Kuwait	KWT	KW	USD
Kyrgyzstan	KGZ	KG	USD
Laos	LAO	LA	USD
Latvia	LVA	LV	EUR
Lebanon	LBN	LB	USD
Lesotho	LSO	LS	ZAR
Liberia	LBR	LR	USD

Libya	LBY	LY	USD
Liechtenstein	LIE	LI	CHF
Lithuania	LTU	LT	EUR
Luxembourg	LUX	LU	EUR
Macau	MAC	MO	MOP
Macedonia	MKD	MK	EUR
Madagascar	MDG	MG	USD
Malawi	MWI	MW	USD
Malaysia	MYS	MY	MYR
Maldives	MDV	MV	USD
Mali	MLI	ML	USD
Malta	MLT	MT	EUR
Marshall Islands	MHL	MH	USD
Martinique	MTQ	MQ	EUR
Mauritania	MRT	MR	USD
Mauritius	MUS	MU	USD
Mayotte	MYT	YT	EUR
Mexico	MEX	MX	MXN
Micronesia	FSM	FM	USD
Moldova	MDA	MD	USD
Monaco	MCO	MC	EUR
Mongolia	MNG	MN	USD
Montenegro	MNE	ME	EUR
Montserrat	MSR	MS	USD
Morocco	MAR	MA	MAD
Mozambique	MOZ	MZ	USD
Myanmar	MMR	MM	USD
Namibia	NAM	NA	ZAR
Nauru	NRU	NR	AUD
Nepal	NPL	NP	USD
Netherlands	NLD	NL	EUR
Netherlands Antilles	ANT	AN	USD
New Caledonia	NCL	NC	USD
New Zealand	NZL	NZ	NZD
Nicaragua	NIC	NI	USD
Niger	NER	NE	USD
Nigeria	NGA	NG	USD
Niue	NIU	NU	NZD
Norfolk Island	NFK	NF	AUD
Northern Mariana Islands	MNP	MP	USD
Norway	NOR	NO	NOK

Oman	OMN	OM	OMR
Pakistan	PAK	PK	PKR
Palau	PLW	PW	USD
Panama	PAN	PA	USD
Papua New Guinea	PNG	PG	USD
Paraguay	PRY	PY	USD
Peru	PER	PE	USD
Philippines	PHL	PH	PHP
Pitcairn Islands	PCN	PN	NZD
Poland	POL	PL	EUR
Portugal	PRT	PT	EUR
Puerto Rico	PRI	PR	USD
Qatar	QAT	QA	QAR
Reunion	REU	RE	EUR
Reunion Island	REU	RE	EUR
Romania	ROU	RO	EUR
Russia	RUS	RU	RUB
Rwanda	RWA	RW	USD
Saint Barthelemy	BLM	BL	EUR
Saint Helena	SHN	SH	GBP
Saint Kitts And Nevis	KNA	KN	USD
Saint Lucia	LCA	LC	USD
Saint Martin (France)	MAF	MF	EUR
Saint Pierre and Miquelon	SPM	PM	USD
Samoa	WSM	WS	USD
San Marino	SMR	SM	EUR
Sao Tome and Principe	STP	ST	USD
Saudi Arabia	SAU	SA	SAR
Senegal	SEN	SN	USD
Senegal (deleted)	SEN	SN	USD
Serbia	SRB	RS	EUR
Serbia And Montenegro	SCG	RS	EUR
Seychelles	SYC	SC	USD
Sierra Leone	SLE	SL	USD
Singapore	SGP	SG	SGD
Sint Maarten (Netherlands)	SXM	SX	USD
Slovakia	SVK	SK	EUR
Slovenia	SVN	SI	EUR
Solomon Islands	SLB	SB	USD
Somalia	SOM	SO	USD
South Africa	ZAF	ZA	ZAR

South Georgia Islands	SGS	GS	GBP
South Korea	KOR	KR	KRW
South Sudan	SSD	SS	USD
Spain	ESP	ES	EUR
Sri Lanka	LKA	LK	USD
St. Vincent & Grenadines	VCT	VC	USD
Sultanate Of Oman	OMN	OM	USD
Suriname	SUR	SR	USD
Swaziland	SWZ	SZ	USD
Sweden	SWE	SE	SEK
Switzerland	CHE	CH	CHF
Taiwan	TWN	TW	TWD
Tajikistan	TJK	TJ	RUB
Tanzania	TZA	TZ	USD
Thailand	THA	TH	THB
Timor-Leste	TLS	TL	USD
Togo	TGO	TG	USD
Tokelau	TKL	TK	NZD
Tonga	TON	TO	USD
Trinidad & Tobago	TTO	TT	USD
Tunisia	TUN	TN	TND
Turkey	TUR	TR	TRY
Turkmenistan	TKM	TM	USD
Turks & Caicos Islands	TCA	TC	USD
Tuvalu	TUV	TV	AUD
Uganda	UGA	UG	USD
Ukraine	UKR	UA	USD
United Arab Emirates	ARE	AE	AED
United Kingdom	GBR	GB	GBP
United States	USA	US	USD
Uruguay	URY	UY	USD
Uzbekistan	UZB	UZ	USD
Vanuatu	VUT	VU	VUV
Venezuela	VEN	VE	USD
Vietnam	VNM	VN	VND
Virgin Islands (British)	VGB	VG	USD
Virgin Islands (U.S.)	VIR	VI	USD
Wallis and Futuna	WLF	WF	USD
Western Sahara	ESH	EH	USD
Yemen	YEM	YE	USD
Yugoslavia	YUG	YU	EUR

Zambia	ZMB	ZM	USD
Zimbabwe	ZWE	ZW	USD

