

Electronic Ticketing Onboarding

SmartOrder Interface for Tickets





Revision History

Version #	Revision Date	Summary of Changes
1.0	11/2021	Document creation to support SmartOrder enhancements.
1.5	2/2022	Instant Fulfillment Requirements
2.0	4/2022	Add Activatetickets endpoint, updated delivery method changes, updated MyProductCatalog Requirement
2.1	7/2022	Added Reporting information pg10
2.2	10/2022	Updated changes to P@H delivery method, and updated daily refresh requirement
2.3	10/2022	Endpoint Examples and Error Code Updates
2.4	11/2022	Updated Daily Refresh Requirement pg25
2.5	03/2023	Added CheckUsage Endpoint, Removed LegalInfo Endpoint and Activateticket Endpoint, Added SmartOrder Booking flow, Added Purchase path and template links to delivery methods, Added fraud prevention language



Table of Contents

Revision History	2
Introduction to SmartOrder	5
About SmartOrder	5
Guest Fulfillment Options / Delivery Methods	6
Onboarding Process	7
Non-Disclosure Agreement.....	7
Financial Requirements	7
Contractual Requirements.....	8
Order Submission, Transmission & Validation Requirements.....	8
Certification & Quality Assurance.....	8
Pre-Launch	9
Price Change / New Product Requirements	9
Support	10
Post-Launch.....	10
Transmission Failures.....	10
Modifications	10
Reporting	10
Fraud Prevention	10
How to Use SmartOrder	11
Client Specific Call Details / Information	11
Object Descriptions.....	12
SmartOrder Booking Flow	14
Place Order Flow.....	14
Cancel Order Flow	15
API Endpoints and Examples	15
/(root) or /smartorder (GET)	15
/authorization (General) and /.well-known/openid-configuration.....	15
/connect/token (POST)	15
/smartorder/TestAuthorized (GET) (requires Bearer token)	17
/smartorder/FindEvents (POST) (requires Bearer token)	18



/smartorder/PlaceOrder (POST) (requires Bearer token)	19
/smartorder/GetExistingOrderId (GET) (requires Bearer token)	22
/smartorder/CancelOrder (GET) (requires Bearer token)	23
/smartorder/CanCancelOrder (GET) (requires Bearer token)	24
/smartorder/MyProductCatalog (GET) (requires Bearer token)	24
/smartorder/MyProductCatalog/flat (GET) (requires Bearer token)	33
/smartorder/CheckUsage (GET) (requires Bearer token)	33
Appendix	35
Miscellaneous Terms	35
GalaxyOrderId and ExternalOrderId	35
Customer ID	35
Sales Program	35
Delivery Method	35
PLU	35
Print at Home Detailed Information	35
Instant Fulfillment (Print on Web) Detailed Information	35
Standard Admission Media Product Template Requirements	36
Authorized Logo	36
Accepted Barcode Types	37
Error Codes	38
SmartOrder Response Definitions	38
Galaxy Error Response Definitions	38
GetExistingOrderId	38
PlaceOrder	39
CanCancelOrder	41
CancelOrder	42
FindEvents	42
MyProductCatalog	43
CheckUsage	43

----- END OF TABLE OF CONTENTS -----



Introduction to SmartOrder

About SmartOrder

SmartOrder is a custom web API hosted by Universal Studios Hollywood ("USH" or "Universal"). It provides Client connectivity to USH's ticketing and access control platform, Galaxy, and allows Clients to sell USH products to their customers.

Clients can access the service via a secure connection to:

- **Stage:** <https://ushstgcorpapi.ucdp.net/>
- **Production:** <https://ushcorpapi.ucdp.net/>

The API allows Clients to:

- Place orders or lookup existing orders.
- Cancel orders or confirm whether or not an order may be cancelled.
- Request live availability, pricing, and descriptions for inventory-managed products.
- Activate tickets and Update ticket expiration in USH's ticketing and access control platform



Guest Fulfillment Options / Delivery Methods

The method in which a Client's Guest(s) retrieve their Admission Media ("ticket") may vary based upon the Delivery Method assigned to the product (PLU). The following Delivery Methods are utilized by SmartOrder, and are described for your reference.

Delivery Method ID	Delivery Method Type	Pickup Location & Information
12	<u>Instant Fulfillment</u> ("Print on Web")	Client will receive visual ID (unique ticket identifier) upon transmission of the order. "Print on Web" is the internal nomenclature for this.
30	<u>Print@Home</u>	Upon Client transmitting the order, which must include a valid Guest e-mail address, the Guest will receive an e-mail from Universal Studios Hollywood with mobile-friendly Print@Home PDF tickets attached.



Onboarding Process

Clients who plan to integrate with Universal Studios Hollywood's SmartOrder need to complete the following prerequisite items. Clients will be assigned to a Trade Distribution Representative, an IT Representative, and a USH Fulfillment Representative for the duration of development. In turn, the Client shall provide USH with a written list of Client Representatives.

Non-Disclosure Agreement

This SmartOrder Onboarding Guide will only be supplied to a client with an executed Admission Media Agreement and/or an executed Non-Disclosure Agreement. Universal will provide Client with the SmartOrder Onboarding Guide as it may be amended from time to time. Client agrees that the SmartOrder Onboarding Guide and Configuration Grid shall be considered Universal's Confidential Information to be governed in accordance with the confidentiality provisions of the Admission Media Agreement and/or the Non-Disclosure Agreement. In the event Client plans to use a third-party provider, such third party provider will be required to execute a Nondisclosure Agreement before the SmartOrder Onboarding Guide is supplied to such third party provider.

Financial Requirements

A Client must be first approved for credit by USH's Accounts Receivable team. This team requires the following items:

- Current W-9 (Domestic) or W8BEN (International)
- Completed & signed Credit Application (see Appendix)
- Projected Annual Sales
- Two (2) Finance contacts to include phone number & e-mail addresses
- Business Model

USH's Accounts Receivable team will evaluate and validate the documentation. Once completed, a payment and collateral term will be offered to the Client which may include a Letter of Credit (LOC), or cash deposit. Once agreed upon, a new Universal client number(s) will be generated.

Client will be invoiced when the visual ID/ticket is generated. Timing of this occurrence may vary based on the delivery method utilized. For example, if the client utilizes USH's straight-to-gate functionality through Print@Home or Instant Fulfillment (Print on Web), the invoice occurs when the visual ID/ticket is generated. In order to lock in the current prevailing contracted rate, it is imperative to successfully import orders at the time of booking into USH's Galaxy database.

If the Client is an existing SmartOrder user that is currently invoiced using the former "upon import" model, please note this method is no longer supported and will require transitioning to the method outlined above. USH's Account Receivable team will assist with the transition. The conversion will not require new Universal client number(s).



Contractual Requirements

The USH Legal team will generate an Admission Media Agreement (“eCommerce agreement”). The Pricing Tool and QA credentials will be provided by Universal Studios Hollywood once an executed Admissions Media Agreement is completed, which then allows the Client to commence development efforts.

USH may, in its reasonable discretion, terminate Client’s access via SmartOrder due to any uncured material and/or technical breach of this document.

Order Submission, Transmission & Validation Requirements

Client must submit electronic Guest orders in accordance with this document, as such may be amended from time to time. Client agrees to comply with all of USH’s policies regarding order submission, notifications, order adjustments, and cancellations. USH is not responsible for misrouted or failed Guest order submissions and will invoice Client based on Client’s electronic Guest order and in accordance with Client’s contracted invoicing and payment terms.

Client will be required to submit all outstanding orders via Live Feed prior to 12AM EST, each day. All orders consisting of a ticket offering (PLU) being terminated or orders being impacted by a price change must be received by Universal Studios Hollywood no later than 12AM EST on the morning of the effective date otherwise the old pricing will not be honored and the order may fail on transmission.

Error monitoring should be in place for live transmissions sent to Universal Studios Hollywood. Client shall maintain a record of all error messages for a period of ninety (90) days. Error report must be available to Universal Studios Hollywood upon request. In the event an error message is received, Client will review error and if assistance is required, Client will inform USH of the error by visiting <https://salesforms.universalparks.com/eticketingcase> . Upon such alert, Universal Studios Hollywood will initiate corrective action. If transmission fails, we recommend the following course of action:

1. Validate SmartOrder lines (Client Number, PLU, Sales Program, Off Sale dates, etc.)
2. Potential Universal Studios Hollywood Server offline. Please submit issue by visiting <https://salesforms.universalparks.com/eticketingcase>
 - a. Once a server is back online, orders should be re-queued and transmitted.

Client is solely responsible for monitoring all ticket order submissions, successful receipt of USH order numbers and validation of all viable ticket orders. Universal will then be responsible for timely reconciliation and invoicing of all resulting viable orders. Client must immediately notify their USH Sales representative(s) and their USH Accounts Receivable representative to report any unauthorized ticket orders or invoicing disputes. Client will be solely responsible for the integration of software that detects fraudulent credit card usage.

Certification & Quality Assurance

Once the Client has completed all prerequisite items which include ticket template approvals (*Please refer to the USH Onboarding Toolkit provided separately for requirements*), the client will be provided a test plan based on products and delivery method offered by the Fulfillment team. After the Client runs



sample transactions in the Production environment according to the script, the Fulfillment team will validate mapping and associated confirmation pages are accurate. The confirmation pages must include the required information as provided by USH. The Fulfillment team will notify the Client when testing is complete. Final approval and go-live date will be provided to the Client and the Trade Distribution Manager. Client should not submit cancellation of sample orders until instructed to do so by the Fulfillment team.

Pre-Launch

Client shall provide USH with a written list of Client Representatives and shall include the following: (i) Client's business name with Client Representative name(s); (ii) contact phone number(s) and email address(es), via email, ten (10) days prior to go-live and shall provide written notice to USH of any change in the Client Representative(s)' contact information within three (3) business days.

Price Change / New Product Requirements

USH reserves the right to change any elements of the Admission Media, including net pricing, terms, and entitlements, if any, and eliminate or add any type of Admission Media on written notice to Client (including notice by email). All changes shall take place on the effective date communicated; provided however, that USH shall nevertheless continue to service all sales for Admission Media products transmitted before the effective date of such change and during the transition period outlined below with the exception of tickets that are not delivered to the Guest prior to the ticket's expiration date. Upon receipt of such net price and/or product change, Client will initiate programming that will result in the new product and/or Client's new pricing being effective in Client's ticketing system and Client will remove any expiring sales program number(s), PLU(s) or pricing from its ticketing system prior to its expiration. If Client fails to convert to the new products or remove discontinued products in accordance with USH's initial written notice, USH may, at its sole discretion, direct Client to suspend SmartOrder sales immediately. In the event of conflict between Client's Universal Studios Hollywood Admission Media Agreement and the terms contained herein, the terms of Client's Universal Studios Hollywood Admission Media Agreement will prevail.

USH will communicate net price and product changes in writing (including notice by email) prior to the effective date. It is the Clients' responsibility to update Client's point of sale and transmit all sales (orders) for discontinued products prior to the communicated effective date of the change. Failure to program new PLUs or a change in sales program will result in failed transmissions.



Support

Post-Launch

USH reserves the right to notify Client of upcoming maintenance outages.

For 24/7 technical issues please visit: <https://salesforms.universalparks.com/eticketingcase>

Transmission Failures

Client is responsible for notifying Universal Studios Hollywood immediately of any failed transmissions affecting arriving Guests. Client may be required to submit a manifest with order details to the USH Fulfillment team.

Modifications

USH requires Clients to cancel/void an existing order requiring a modification and to submit a new order with a new external ID as a replacement. Client will be required to provide Client's Guest with a revised confirmation page that includes the new external ID. Client will be solely responsible for evaluating any price increase that could result from the modification of an existing order.

Reporting

SmartOrder does not provide reporting functionality within the code. If reporting is needed, then it is the Client's responsibility for developing their own reporting needs with their development team. Client can utilize the attributes provided within the SmartOrder endpoints to create personalized reporting.

Fraud Prevention

SmartOrder is a fulfillment API only. Per the Universal Admission Media Agreement, it is the client's responsibility to ensure adequate credit card fraud prevention measures are in place.



How to Use SmartOrder

The SmartOrder API is used to process orders into USH's ticketing and admission control platform. An order is a collection of line items (order lines), each representing a particular Product and quantity.

As an added convenience for our Clients, Parks Technology has utilized SwaggerUI to create API endpoint descriptions with built-in development assistance and testing tools. The service definitions implement OpenAPI v3 and display a user- friendly UI for testing purposes, but only in QA. The URL to view these pages can be found at:

<https://ushstgcorpAPI.ucdp.net/smartorder/help/index.html>

Client Specific Call Details / Information

Before using the API, several pieces of data will be provided to the Client by the Universal team assigned. These are specific to each Client, in particular Client Identifier aka Client Number or Customer ID, and are accessible to Client through Client's "Pricing Grid" via the Universal Partner Community. This document outlines the general integration with the API and gives them contracted product and pricing. Depending on your ability to integrate into the system, you may also use the Product Catalog endpoint described later in this onboarding guide, which only needs a few pieces of data and can return back your available products and pricing by request.

Example Pricing Grid:

FILE HOME INSERT PAGE LAYOUT FORMULAS DATA REVIEW VIEW ACROBAT														Richardson, haa (NBCUniversal, Ontario)									
Cut Copy Paste Font Painter Alignment Number General Conditional Formatting Table Normal Bad Good Neutral Calculation Check Cell Exploratory Input Insert Delete Format Fill Sort & Find & Select Share This Workbook Clear Filter Filter Editing Webex																							
Clipboard Font Alignment Number Styles																							
D17																							
A B C D E F G H I J K L M																							
Client Number Client Name Sales Program Sales Program Name PLU Product Name Date Suggested Retail Net Rate Net Rate with Tax Delivery Method ID Pricing Type Banner Color																							
1 12345 TEST ACCOUNT 3829 Test 110114601010 CityWalk Party Pass 11.99 4.20 4.48 7.85.92.108 Current Yellow																							
2 12345 TEST ACCOUNT 3728 Test 110114601010 CityWalk Party Pass 11.99 4.20 4.48 7.85.92.108 Current Yellow																							
3 12345 TEST ACCOUNT 3829 Test 110114801027 CityWalk Party Pass & Meal Ticket 24.24 24.24 25.82 7.85.92.108 Current Red																							
4 12345 TEST ACCOUNT 3829 Test 110114801027 CityWalk Party Pass & Meal Ticket 24.24 24.24 25.82 7.85.92.108 Current Red																							
5 12345 TEST ACCOUNT 3829 Test 150114801003 CityWalk Meal & Movie Deal Ticket 28.12 26.12 27.82 7.85.92.108 Current Red																							
6 12345 TEST ACCOUNT 3728 Test 150114801003 CityWalk Meal & Movie Deal Ticket 28.12 26.12 27.82 7.85.92.108 Current Red																							
7 12345 TEST ACCOUNT 3728 Test 150141111007 1-Park 1-Day Base Value Adult 119.00 119.00 126.74 7.85.92.108 Current Green																							
8 12345 TEST ACCOUNT 3728 Test 150141111016 1-Park 1-Day Base Regular Adult 128.00 128.00 136.32 7.85.92.108 Current Green																							
9 12345 TEST ACCOUNT 3728 Test 150141111026 1-Park 1-Day Base Value Child 114.00 114.00 121.41 7.85.92.108 Current Green																							
10 12345 TEST ACCOUNT 3728 Test 150141111035 1-Park 1-Day Base Regular Child 123.00 123.00 131.00 7.85.92.108 Current Green																							
11 12345 TEST ACCOUNT 3829 Test 150141120001 1-Park 1-Day Base Ticket Adult - Anytime Season 139.00 139.00 148.04 7.85.92.108 Current Green																							
12 12345 TEST ACCOUNT 3829 Test 150141120001 1-Park 1-Day Base Ticket Adult - Anytime Season 139.00 139.00 148.04 7.85.92.108 Current Green																							
13 12345 TEST ACCOUNT 3829 Test 150141170001 1-Park 1-Day Base Ticket Child - Anytime Season 134.00 134.00 142.71 7.85.92.108 Current Green																							
14 12345 TEST ACCOUNT 3829 Test 150141170001 1-Park 1-Day Base Ticket Child - Anytime Season 134.00 134.00 142.71 7.85.92.108 Current Green																							
15 12345 TEST ACCOUNT 3728 Test 150145301012 2-Park 5-Day Park-to-Park Ticket Child 334.99 276.37 294.34 7.85.92.108 Current Green																							
16 12345 TEST ACCOUNT 3728 Test 150145301012 3-Park 5-Day Base Ticket Adult 349.99 300.99 320.56 7.85.92.108 Current Green																							
17 12345 TEST ACCOUNT 3728 Test 150145301012 3-Park 5-Day Base Ticket Adult 349.99 300.99 320.56 7.85.92.108 Current Green																							
18 12345 TEST ACCOUNT 3829 Test 150145351012 3-Park 5-Day Base Ticket Child 339.99 292.39 311.40 7.85.92.108 Current Green																							
19 12345 TEST ACCOUNT 3728 Test 150145351012 3-Park 5-Day Base Ticket Child 339.99 292.39 311.40 7.85.92.108 Current Green																							
20 12345 TEST ACCOUNT 3829 Test 150145401012 3-Park 5-Day Park-to-Park Ticket Adult 419.99 346.49 369.02 7.85.92.108 Current Green																							
21 12345 TEST ACCOUNT 3728 Test 150145401012 3-Park 5-Day Park-to-Park Ticket Adult 419.99 346.49 369.02 7.85.92.108 Current Green																							
22 12345 TEST ACCOUNT 3829 Test 150145451012 3-Park 5-Day Park-to-Park Ticket Child 409.99 338.24 360.24 7.85.92.108 Current Green																							
23 12345 TEST ACCOUNT 3728 Test 150145451012 3-Park 5-Day Park-to-Park Ticket Child 409.99 338.24 360.24 7.85.92.108 Current Green																							
24 12345 TEST ACCOUNT 3829 Test 170110111007 1-Park 1-Day Base Ticket Dated Adult 2020-08-04T00:00:00-04:00 150.00 150.00 159.75 7.85.92.108 Current Green																							
25 12345 TEST ACCOUNT 3829 Test 170110111007 1-Park 1-Day Base Ticket Dated Adult 2020-08-05T00:00:00-04:00 150.00 150.00 159.75 7.85.92.108 Current Green																							
26 12345 TEST ACCOUNT 3829 Test 170110111007 1-Park 1-Day Base Ticket Dated Adult 2020-08-06T00:00:00-04:00 150.00 150.00 159.75 7.85.92.108 Current Green																							
27 12345 TEST ACCOUNT 3829 Test 170110111007 1-Park 1-Day Base Ticket Dated Adult 2020-08-07T00:00:00-04:00 150.00 150.00 159.75 7.85.92.108 Current Green																							
28 12345 TEST ACCOUNT 3829 Test 170110111007 1-Park 1-Day Base Ticket Dated Adult 2020-08-08T00:00:00-04:00 150.00 150.00 159.75 7.85.92.108 Current Green																							
29 12345 TEST ACCOUNT 3829 Test 170110111007 1-Park 1-Day Base Ticket Dated Adult 2020-08-09T00:00:00-04:00 145.00 145.00 154.43 7.85.92.108 Current Green																							
30 12345 TEST ACCOUNT 3829 Test 170110111007 1-Park 1-Day Base Ticket Dated Adult 2020-08-10T00:00:00-04:00 145.00 145.00 154.43 7.85.92.108 Current Green																							
31 12345 TEST ACCOUNT 3829 Test 170110111007 1-Park 1-Day Base Ticket Dated Adult 2020-08-11T00:00:00-04:00 145.00 145.00 154.43 7.85.92.108 Current Green																							
32 12345 TEST ACCOUNT 3829 Test 170110111007 1-Park 1-Day Base Ticket Dated Adult 2020-08-12T00:00:00-04:00 145.00 145.00 154.43 7.85.92.108 Current Green																							
33 12345 TEST ACCOUNT 3829 Test 170110111007 1-Park 1-Day Base Ticket Dated Adult 2020-08-13T00:00:00-04:00 145.00 145.00 154.43 7.85.92.108 Current Green																							
34 12345 TEST ACCOUNT 3829 Test 170110111007 1-Park 1-Day Base Ticket Dated Adult 2020-08-14T00:00:00-04:00 145.00 145.00 154.43 7.85.92.108 Current Green																							
35 12345 TEST ACCOUNT 3829 Test 170110111007 1-Park 1-Day Base Ticket Dated Adult 2020-08-15T00:00:00-04:00 145.00 145.00 154.43 7.85.92.108 Current Green																							
36 12345 TEST ACCOUNT 3829 Test 170110111007 1-Park 1-Day Base Ticket Dated Adult 2020-08-16T00:00:00-04:00 145.00 145.00 154.43 7.85.92.108 Current Green																							
37 12345 TEST ACCOUNT 3829 Test 170110111007 1-Park 1-Day Base Ticket Dated Adult 2020-08-17T00:00:00-04:00 145.00 145.00 154.43 7.85.92.108 Current Green																							
38 12345 TEST ACCOUNT 3829 Test 170110111007 1-Park 1-Day Base Ticket Dated Adult 2020-08-18T00:00:00-04:00 145.00 145.00 154.43 7.85.92.108 Current Green																							
39 12345 TEST ACCOUNT 3829 Test 170110111007 1-Park 1-Day Base Ticket Dated Adult 2020-08-19T00:00:00-04:00 145.00 145.00 154.43 7.85.92.108 Current Green																							
40 12345 TEST ACCOUNT 3829 Test 170110111007 1-Park 1-Day Base Ticket Dated Adult 2020-08-20T00:00:00-04:00 145.00 145.00 154.43 7.85.92.108 Current Green																							
41 12345 TEST ACCOUNT 3829 Test 170110111007 1-Park 1-Day Base Ticket Dated Adult 2020-08-21T00:00:00-04:00 145.00 145.00 154.43 7.85.92.108 Current Green																							
42 12345 TEST ACCOUNT 3829 Test 170110111007 1-Park 1-Day Base Ticket Dated Adult 2020-08-22T00:00:00-04:00 145.00 145.00 154.43 7.85.92.108 Current Green																							



Object Descriptions

Field Name	Description	Required	Type and restrictions
Client Username & Client Secret	Log in credentials for API access	Yes	String, provided by Universal
CustomerId	This is the unique ID that represents the Client in the Ticket platform.	Yes	Integer, provided by Universal
ExternalOrderId	This is the Client order/booking number with USH approved suffix. This is also referred to as the confirmation number.	Yes	String (max 30)
DeliveryMethodId	The chosen DeliveryMethod ID If one PLU within the order is assigned to the delivery method of 85 – Ticket Window Pick-up, the entire order must also use that delivery method. <i>See Delivery Method Options starting on Page 6</i>	Yes	Integer
OrderSalesProgramId	A value indicating the Client program for which the product is sold.	**	Integer, configured per-product
FirstName	Guest's first name	Yes	String (max 30)
LastName	Guest's last name	Yes	String (max 30)
Street1	Guest's billing address	No	String (max 255)
Street2	Guest's billing address line 2	No	String (max 255)
City	Guest's city on billing address	No	String (max 40)
State	Guest's state on billing address	No	String (max 20)
ZipCode	Guest's billing zip or postal code	No	String (max 16)
Phone	Guest's phone number entered at the time of order/booking or a substitution	Yes	String (max 30)
Email	Guest's email address entered at the time of ordering	*	String (max 128)
ClientLocation	A user-field that can be used to identify where the order/booking was initiated, such as a branch or location within the Client's enterprise; beneficial for Clients whom operate out of multiple offices/desks.	*****	String (max 255)
SmartOrderLines	This is the collection of Order Lines which contain the quantity and type of products being ordered.	Yes	Collection/Array
Line → PLU	This identifies the product. Please refer to the Client Configuration Guide.	Yes	String (max 20), configured per product
Line → Quantity	This number represents the quantity of tickets purchased of the relative PLU. <i>This value must be 1 when providing GuestName value.</i>	Yes	Integer > 0
Line → PluSalesProgramId	A value indicating the Client program for which the product is sold.	**	Integer, configured per product
Line → EventDateTime	The selected date/time for the product (if required).	***	Date/Time
Line → EventId	The specific EventId to be used for the product being sold.	***	Integer, from FindEvents
Line → ByTicketGuestNames	The guest names for Passes and Print @ Home tickets	****	Collection/Array
Line → GuestName → First	The first name of the Guest	****	String (max 30)
Line → GuestName → Last	The last name of the Guest	****	String (max 30)
Line → ForeignMediaVisualId	If allowed for your client, the VisualId to set on the ticket (not commonly used)	No	String (max 40)



* E-mail address is required for the Print@Home delivery method. E-mail address is not used for any re-targeting communication from Universal Studios Hollywood. For tests run in our non-production environment, please utilize a test or Client e-mail address when submitting bookings and refrain from using a consumer/Guest's email address.

** If all of the Products (PLU codes) in the order/booking use the same Sales Program ID, the value at the Order Line level can be set to NULL or excluded from the message entirely. The Order Line Sales Program ID value only needs to be set if multiple Sales Program ID values exist in the same booking. In this case, the OrderSalesProgramIdvalue should be set to NULL. If values exist for both the OrderSalesProgramIdand the PluSalesProgramID, the Product (PLU) level has priority and will be used during Order Validation.

*** EventDateTime is required for capacity managed, event-based, or per-day price scheduled products. With all other products, these parameters should be left out or set null. To populate these values, use the results of a previous FindEvents call (which provided you with a list of "Events" to choose). From the results of that call, the chosen Event's EventDateTime and EventID values are set in their corresponding fields here in the Order message. The EventId, while not required, can assist in choosing a specific event if there is more than one on a particular date (and is required in such a case). Developers may opt to always populate both fields when Event data is required, to avoid the potential future implementation issues.

**** Required for passes, tickets (where the delivery method on the Order is set as Print@Home or Instant Fulfillment), and for other products where a Guest Name has been configured as required, you must give one GuestName (First and Last) entry per Order Line. Note: The quantity on these Order Lines must be set to 1 or a rejection will occur.

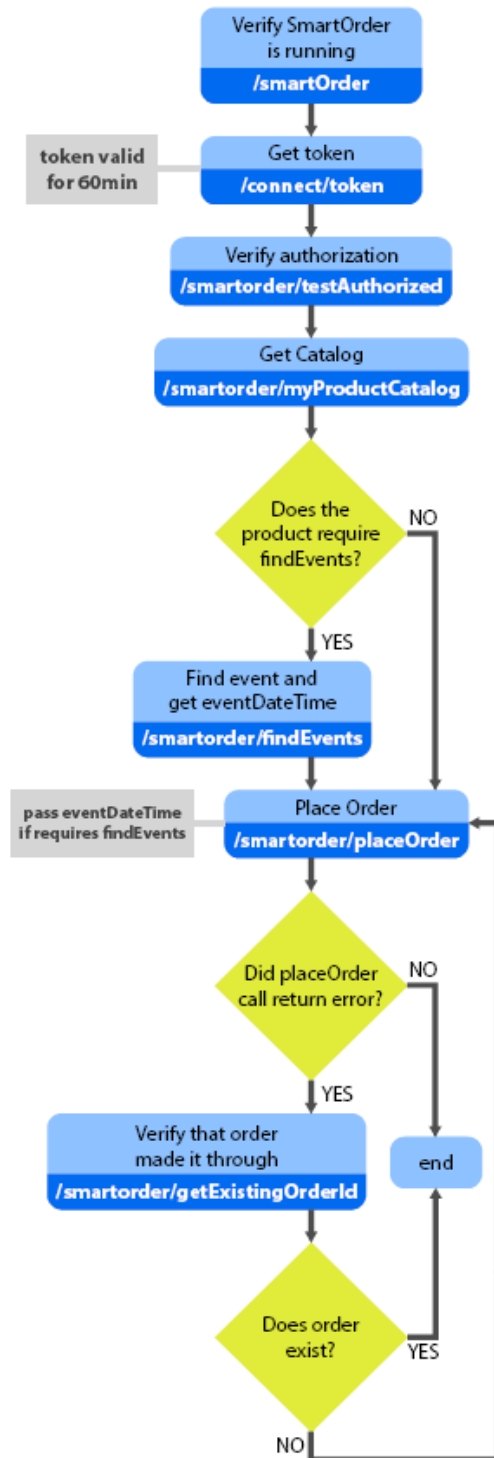
***** Required for clients who sell via multiple in-person locations to help with tracking where the sale originates

NOTE: The rules outlined in this document are the general use of each item. There may be exceptions or additions for particular products. Each Product (PLU) indicates the appropriate data points required on the Pricing Grid supplied by Universal through the Universal Partner Community.



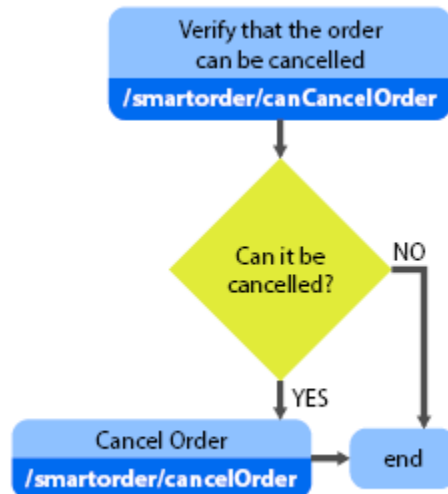
SmartOrder Booking Flow

Place Order Flow





Cancel Order Flow



API Endpoints and Examples

All URLs in this section start with the base URL as follows:

- **Stage:** <https://UshStgCorpApi.ucdp.net/>
- **Production:** <https://UshCorpApi.ucdp.net/>

[/\(root\) or /smartorder \(GET\)](#)

These endpoints are solely for connectivity checks. Client systems should be able to reach these endpoints and get a 200/"OK" response without authenticating. The first is for general network connectivity to the services collection endpoint, the second is for connectivity to SmartOrder specifically.

[/authorization \(General\) and /.well-known/openid-configuration](#)

These endpoints are solely for connectivity checks. Client systems should be able to reach these endpoints and get a 200/"OK" response without authenticating. The first is for general network connectivity to the services collection endpoint, the second is for connectivity to SmartOrder specifically.

[/connect/token \(POST\)](#)

This endpoint is a standard form-post authentication method using your client credentials in the body of the message. The result is a token which shall be placed in the HTTP Authorization header as "Bearer [token]" for all further requests in the same session. The token's default timeout is 1 hour, which can be read from the response or JWT token itself. Many Clients will opt to request a new token for every order to simplify the coding process, using the same token for all requests within a transaction/order and then discarding it afterwards.



In order to use this endpoint, either use an existing OpenID Connect or OAuth2 adapter (in “Client Credentials, in-body” configuration), or use an HTTP form post with data similar to the example below. Note that some language implementations may require you to encode the client_secret.

Example Request:

(Posted to <https://UshStgCorpAPI.ucdp.net/auth>)

```
grant_type=client_credentials
client_id=MyUsername
scope=SmartOrder
client_secret=bc3456d1-95bf-3210-e8e8-22c748db9fe0
```

Example curl Request:

```
curl --location --request POST 'https://ushstgcorpapi.ucdp.net/connect/token' \
--header 'Content-Type: application/x-www-form-urlencoded' \
--data 'client_id=MyUsername' \
--data client_secret=bc3456d1-95bf-3210-e8e8-22c748db9fe0 \
--data scope=SmartOrder \
--data grant_type=client_credentials
```




[/smartorder/FindEvents \(POST\) \(requires Bearer token\)](#)

This endpoint allows Clients to retrieve the current Capacity for a date-based or capacity-managed product, as well as Event ID's & Times available for the Product (PLU) depending on the date range provided.

Note: The MyProductCatalog Endpoint now has a RequiresFindEvents flag that is either true or false to indicate if FindEvents is required.

This call should only be used if the product(s) require this functionality.

Additionally, the pricing fields ("totalPriceWithTax", "price", "tax1", "tax2") returns the price for that product, which may vary if the product's price varies per-day ("price scheduled"). If the correct salesProgramID is included in the request, the pricing fields returned reflect contracted net pricing. If salesProgramID is not included in the request, the pricing fields returned reflect USH's online retail (base) price.

The parameters startDateInclusive and endDateInclusive, as they indicate, are inclusive. HOWEVER, time in the request matters. If time is not provided, the default time will be set to midnight of that day, and default time zone set to the local value of the park in the order. Similarly, the eventDateTime returned will have the time zone of the park in the response.

It is Not Recommended to Cache this call, as event capacity and availability can change on a daily bases. If you require caching, it is recommended to only cache data past a 30-day period.

Example Request:

```
{
  "plu": "123456789",
  "startDateInclusive": "2018-08-01",
  "endDateInclusive": "2018-08-31",
  "salesProgramID": "1234"
}
```

Example Response:



```
{
  "success": true,
  "eventResults": [
    {
      "eventId": 123,
      "eventDateTime": "2018-08-01T13:49:49.556-5:00",
      "capacityAvailable": 10,
      "totalPriceWithTax": 123.45,
      "price": 120.00,
      "tax1": 3.00,
      "tax2": 0.45,
    },
    {
      "eventId": 124,
      "eventDateTime": "2018-08-02T13:49:49.556-5:00",
      "capacityAvailable": 12,
      "totalPriceWithTax": 124.56,
      "price": 120.00,
      "tax1": 4.00,
      "tax2": 0.56,
    }
  ]
}
```

[/smartorder/PlaceOrder \(POST\) \(requires Bearer token\)](#)

This is the main Order endpoint for the SmartOrder Service to create an Order in Galaxy.

NOTE: This call has several optional request and response fields that depend on the configuration of the Client. See the MyProductCatalog Endpoint to determine which fields and values are appropriate. Also, note that Adult and Child tickets are required to be in the same order and cannot be transmitted separately.

Example Request:



```
{
  "externalOrderId": "EOT123456",
  "customerId": 123,
  "orderSalesProgramId": 234,
  "deliveryMethodId": 67,
  "firstName": "John",
  "lastName": "Doe",
  "street1": "123 main st.",
  "street2": null,
  "city": "Orlando",
  "state": "Florida",
  "zipCode": "32819",
  "phone": "123-123-1234",
  "email": "myemail@fakeemail.com",
  "clientLocation": "Loc1234",
  "smartOrderLines": [
    {
      "pluSalesProgramId": null,
      "plu": "123456789",
      "quantity": 2,
      "byTicketGuestNames": [
        {
          "first": "John",
          "last": "Doe"
        },
        {
          "first": "Mom",
          "last": "Doe"
        }
      ]
    },
    {
      "eventDateTime": "2020-01-01T04:00"
    }
  ],
  {
    "pluSalesProgramId": null,
    "plu": "012345678",
    "quantity": 1,
    "byTicketGuestNames": [
      {
        "first": "Kid",
        "last": "Doe"
      }
    ]
  },
  {
    "eventDateTime": "2020-01-01T04:00"
  }
]
}
```

Example Response: (if non-Instant Fulfillment)

```
{
  "success": true,
  "galaxyOrderId": 987654321
}
```



Example Response: (if Instant Fulfillment)

```
{
  "success": true,
  "galaxyOrderId": 987654321,
  "createdTicketResponses": [
    {
      "visualID": "73000000000000000001",
      "plu": "123456789",
      "firstName": "John",
      "lastName": "Doe",
      "eventDateTime": "2020-01-01T04:00:00-04:00",
      "validityRules": [
        {
          "operations": [
            1,2,4,5,6,7,8,9,10,90
          ],
          "calculatedStartDateTime": "2020-01-01T04:00:00-04:00",
          "calculatedExpirationDateTime": "2020-01-05T23:59:00-04:00"
        }
      ],
      "unprocessedRelativeRules": []
    },
    {
      "visualID": "73000000000000000002",
      "plu": "123456789",
      "firstName": "Mom",
      "lastName": "Doe",
      "eventDateTime": "2020-01-01T04:00:00-04:00",
      "validityRules": [
        {
          "operations": [
            1,2,4,5,6,7,8,9,10,90
          ],
          "calculatedStartDateTime": "2020-01-01T04:00:00-04:00",
          "calculatedExpirationDateTime": "2020-01-05T23:59:00-04:00"
        }
      ],
      "unprocessedRelativeRules": []
    },
    {
      "visualID": "73000000000000000003",
      "plu": "012345678",
      "firstName": "Kid",
      "lastName": "Doe",
      "eventDateTime": "2020-01-01T04:00:00-04:00",
      "validityRules": [
        {
          "operations": [
            1,2,4,5,6,7,8,9,10,90
          ],
          "calculatedStartDateTime": "2020-01-01T04:00:00-04:00",
          "calculatedExpirationDateTime": "2020-01-05T23:59:00-04:00"
        }
      ],
      "unprocessedRelativeRules": []
    }
  ]
}
```



/smartorder/GetExistingOrderId (GET) (requires Bearer token)

This endpoint retrieves order details for an order that already exists. (More detail is provided if order used Instant Fulfillment method.)

Example Request:

<https://ushstgcorpapi.ucdp.net/smartorder/GetExistingOrderId?ExternalOrderId=EOT12345>

Example Response: *(if non-Instant Fulfillment)*

```
{  
  "success": true,  
  "galaxyOrderId": 987654321  
}
```

Example Response: *(if Instant Fulfillment)*



```
{
  "success": true,
  "galaxyOrderId": 987654321,
  "createdTicketResponses": [
    {
      "visualID": "73000000000000000001",
      "plu": "123456789",
      "firstName": "John",
      "lastName": "Doe",
      "eventDateTime": "2020-01-01T04:00:00-04:00",
      "validityRules": [
        {
          "operations": [
            1,2,4,5,6,7,8,9,10,90
          ],
          "calculatedStartDateTime": "2020-01-01T04:00:00-04:00",
          "calculatedExpirationDateTime": "2020-01-05T23:59:00-04:00"
        }
      ],
      "unprocessedRelativeRules": []
    },
    {
      "visualID": "73000000000000000002",
      "plu": "123456789",
      "firstName": "Mom",
      "lastName": "Doe",
      "eventDateTime": "2020-01-01T04:00:00-04:00",
      "validityRules": [
        {
          "operations": [
            1,2,4,5,6,7,8,9,10,90
          ],
          "calculatedStartDateTime": "2020-01-01T04:00:00-04:00",
          "calculatedExpirationDateTime": "2020-01-05T23:59:00-04:00"
        }
      ],
      "unprocessedRelativeRules": []
    },
    {
      "visualID": "73000000000000000003",
      "plu": "012345678",
      "firstName": "Kid",
      "lastName": "Doe",
      "eventDateTime": "2020-01-01T04:00:00-04:00",
      "validityRules": [
        {
          "operations": [
            1,2,4,5,6,7,8,9,10,90
          ],
          "calculatedStartDateTime": "2020-01-01T04:00:00-04:00",
          "calculatedExpirationDateTime": "2020-01-05T23:59:00-04:00"
        }
      ],
      "unprocessedRelativeRules": []
    }
  ]
}
```

[/smartorder/CancelOrder \(GET\) \(requires Bearer token\)](#)

This endpoint cancels the created order.



NOTE: Any modifications to the order by Universal may prevent it from being cancelled from this API, including usage of the tickets within. Please consider using [/smartorder/CanCancelOrder](#) before calling this endpoint.

Example Request:

<https://ushstgcorpapi.ucdp.net/smartorder/CancelOrder?ExternalOrderId=EOT12345>

Example Response:

```
{
  "success": true
}
```

[/smartorder/CanCancelOrder \(GET\) \(requires Bearer token\)](#)

This endpoint checks to see if an order can be cancelled.

Note: Any modifications to the order by Universal may prevent it from being cancelled from this API, including usage of the tickets within.

While this endpoint catches the vast majority of common cases, it is not a definitive indicator of CancelOrder being possible. The Galaxy system is complex and changes configuration often and per-client, so your success may vary.

Example Request:

<https://ushstgcorpapi.ucdp.net/smartorder/CanCancelOrder?ExternalOrderId=EOT12345>

Example Response:

```
{
  "success": true
}
```

[/smartorder/MyProductCatalog \(GET\) \(requires Bearer token\)](#)

This endpoint allows clients to retrieve all relevant data for their particular product catalog.

It is recommended to Cache this call, as this it returns a large amount of data.

IMPORTANT NOTE:

It is required that your IT Developer set up an automatic update for pricing ingestion of your cached data at least once per day; however, it is recommended to schedule multiple intra-day updates if possible. This process will ensure you seamlessly pick up changes to pricing and products. Most changes occur at 8AM PST.

To ensure front-end pricing reflects the latest updates, you will need to implement a real-time synchronization process between your back-end and front-end pricing or schedule front-end pricing ingestion to occur at the same time as your automatic update for pricing ingestion of your cached data.



1. This endpoint comes with several caveats/warnings: **WARNING - this can take a large amount of time for large date ranges or clients with a large number of products. While it is a fairly reliable endpoint itself, it must gather data from a large number of disparate sources that are, on occasion, not as reliable. Implement your own retry logic, and know that this is NOT meant to be used for live Order processing,** and is only provided for clients to use in printed material or non-live price and option displays. The “FindEvents” call is to be used when orders are actually being placed, and to receive the most up-to-date pricing.
2. This call has parameters `startDateInclusive` and `endDateInclusive`. These are required if at least one of your products has a price that varies by day. As they indicate, the dates are inclusive, HOWEVER, time in the request matters. If time is not provided, the default time is midnight of that day, and default time zone would be set to the local value of the park in the order.
 - a. Similarly, the `effectiveDateTime` returned will have the time zone of the park in the response.
 - b. Also note that if you provide dates for a product that is date-specific, but there are no dates for that product in the range you specify, the endpoint will not return the product at all. This is an intentional choice – the endpoint only returns data for products you can sell for those dates.

For example: Halloween Horror Nights (HHN) will only show up in your catalog when you search for months around Halloween.
3. This call has many optional parameters to help filter or otherwise reduce the amount of data returned. Please see the Swagger documentation for more specific information – some clients may wish to turn on these filters to simplify processing time, or make several separate calls for different data sets at different times. Some of those filters include as found below, with a brief explanation of important fields they affect in the call.
 - **retrieveOnly** = current/future
 - **CurrentPricing** represents currently available pricing for Universal.
 - **FuturePricing** indicates a change in a price coming from Universal in the future.

Note: Both Current and Future pricing may be identical, which indicated no current price changes available in the near future.
(Universal’s marketing team will also inform clients of any price changes directly.)
 - **pricing** = base/discounted
 - *Found under both Current and Future pricing calls.*
 - **BasePriceData** is the everyday pre-arrival pricing that can be found on Universal’s home websites.

DiscountedPriceData is a client’s specific contracted pricing for the ticket.
Note: There may be times when these two attributes contain the same data which indicates that, although contracted, there is no discount available for the client.
 - **PricesByDay** (*Optional*) if dated product, pricing is broken down for each date.
 - **changesSince** = ‘datetime’
 - Set to a date/time to retrieve data with changes since your specified date/time. This is, any changes Universal has made to a price or description info to a ticket PLU.
 - **collapseDates** = true/false



- Default False. The system will collapse all consecutive dates with similar data into a date range, using PricingDateTime as the start, and PricingRangeEndDateTime as the end.
- **calculateDailyPrice** = true/false
 - Default True. Enable if you the system will not calculate the price for any products that have a price that varied by day. It will instead only return the discount fields to indicate how much of a discount will be provided. **Cannot be set false if collapseDates=true.**
- Response Attributes

RESPONSE ATTRIBUTE	DESCRIPTION	TYPE OF VALUE
"plu"	The unique Product Look Up code assigned to the product	string
"productName"	The name/description of the product	string
"allowedDeliveryMethods"	List of all delivery methods that can be used for the specific product. The Onboarding Guide has a list of all delivery methods with descriptions	list of integers
"bannerColor"	The color the ticket media banner should be for the specific product	string
"numberOfDays"	The number of days the ticket can be used to enter the theme parks after it is first used	integer
"isParkToPark"	Checks if product can be used to enter multiple parks the same day using the same ticket	boolean
"ageValue"	"A" when the ticket can only be used by an adult, "C" for children, and N/A if age doesn't matter	string
"isLimitedExpress"	Checks if product can be used for limited express access to attractions (1-time use for each attraction)	boolean
"isUnlimitedExpress"	Checks if product can be used for unlimited express access to attractions (unlimited express access to attractions)	boolean
"isFloridaResidentRequired"	Checks if product can only be used by a Florida resident.	boolean
"residencyRequirement"	Checks if product can only be used by a resident of the state the theme park is located in	string
"isThemeParkAccess"	Checks if product can be used to enter a theme park	boolean
"themeParkAccessNames"	List of all the theme parks the product can be used to enter ("USF" = "Universal Studios Florida", "UIA" = "Universal Islands of Adventure", "VBP" = "Volcano Bay Park", "USH" = "Universal Studios Hollywood")	list of strings



"isHHNAccess"	Checks if product gives access to "Halloween Horror Nights" ("HHN") event	boolean
"isRTUAccess"	Checks if product gives access to "Rock The Universe" ("RTU") event	boolean
"isGradEventAccess"	Checks if product gives access to "GradBash" or "GradAdventure" events (school graduation events)	boolean
"requiresFindEvents"	Checks if product is a dated product (returns true) or flex product (returns false). The use of the FindEvents endpoint is required for all dated products in order to find out capacityAvailable and eventDateTime	boolean
"productKind"	The type of product	string
"currentPricing"	Lists current basePriceData and dicountedPriceData for product for the given date range ("base" = "suggested retail pricing", "discounted" = "net rates")	list of price data
"basePriceData"	Lists totalPriceWithTax, price, and different types of taxes (tax1 = State Tax, tax2 = County Tax) associated with the specific product for a specific date	list of price data
"discountedPriceData"	Lists totalPriceWithTax, price, different types of taxes (tax1 = State Tax, tax2 = County Tax), discountType, and discountAmount associated with the specific product for a specific date. <u>The totalPriceWithTax in discounterPriceData is the amount the client will pay Universal</u>	list of price data
"futurePricing"	Lists future basePriceData and dicountedPriceData for product for the given date range ("base" = "suggested retail pricing", "discounted" = "net rates")	list of price data

Example Request:

<https://ushstgcorpapi.ucdp.net/smartorder/MyProductCatalog?customerId=12345&salesProgramId=2345&startDateInclusive=2020-01-01&endDateInclusive=2020-02-01>

---- See Next Page for Response ----



Example Response:

(Fabricated Data – please use your designated configuration IDs and code against the swagger and sample data provided from a test call)

```
{
  "catalogBySalesProgram": [
    {
      "salesProgramId": 2345,
      "salesProgramName": "Test VP SC",
      "productCatalogEntries": [
        {
          "plu": "112233445566",
          "productName": "2-Park Power Annual Pass - Florida Resident Adult/Child",
          "allowedDeliveryMethods": [
            85
          ],
          "bannerColor": "Blue",
          "numberOfDays": 365,
          "isParkToPark": true,
          "ageValue": "A",
          "isLimitedExpress": false,
          "isUnlimitedExpress": false,
          "isFloridaResidentRequired": true,
          "residencyRequirement": "NA",
          "isThemeParkAccess": true,
          "themeParkAccessNames": [
            "USF",
            "UIA"
          ],
          "isHHNAccess": false,
          "isRTUAccess": false,
          "isGradEventAccess": false,
          "requiresFindEvents": false,
          "productKind": "Pass",
          "currentPricing": {
            "basePriceData": {
              "singlePriceWithTaxResponse": {
                "totalPriceWithTax": 340.79,
                "price": 319.99,
                "tax1": 19.2,
                "tax2": 1.6
              }
            }
          },
          "futurePricing": {
            "basePriceData": {
              "singlePriceWithTaxResponse": {
                "totalPriceWithTax": 340.79,
                "price": 319.99,
                "tax1": 19.2,
                "tax2": 1.6
              }
            }
          }
        }
      ]
    }
  ],
}
```



```
{
  "plu": "113355779900",
  "productName": "CityWalk Party Pass",
  "allowedDeliveryMethods": [
    7,
    85
  ],
  "bannerColor": "Red",
  "numberOfDays": 0,
  "isParkToPark": true,
  "ageValue": "NA",
  "isLimitedExpress": true,
  "isUnlimitedExpress": true,
  "isFloridaResidentRequired": true,
  "residencyRequirement": "NA",
  "isThemeParkAccess": false,
  "themeParkAccessNames": [],
  "isHHNAccess": true,
  "isRTUAccess": true,
  "isGradEventAccess": true,
  "requiresFindEvents": false,
  "productKind": "Pass",
  "currentPricing": {
    "basePriceData": {
      "singlePriceWithTaxResponse": {
        "totalPriceWithTax": 12.77,
        "price": 11.99,
        "tax1": 0.72,
        "tax2": 0.06
      }
    },
    "discountedPriceData": {
      "singlePriceWithTaxResponse": {
        "totalPriceWithTax": 4.48,
        "price": 4.2,
        "tax1": 0.26,
        "tax2": 0.02,
        "discountType": 2,
        "discountAmount": 7.79
      }
    }
  },
  "futurePricing": {
    "basePriceData": {
      "singlePriceWithTaxResponse": {
        "totalPriceWithTax": 12.77,
        "price": 11.99,
        "tax1": 0.72,
        "tax2": 0.06
      }
    },
    "discountedPriceData": {
      "singlePriceWithTaxResponse": {
        "totalPriceWithTax": 4.48,
        "price": 4.2,
        "tax1": 0.26,
        "tax2": 0.02,
        "discountType": 2,
        "discountAmount": 7.79
      }
    }
  }
},
},
```



```
{
  "plu": "110117001386",
  "productName": "Universal Studios Florida Universal Express Pass",
  "allowedDeliveryMethods": [
    7,
    85
  ],
  "bannerColor": "Yellow",
  "numberOfDays": 0,
  "isParkToPark": false,
  "ageValue": "string",
  "isLimitedExpress": true,
  "isUnlimitedExpress": false,
  "isFloridaResidentRequired": true,
  "residencyRequirement": "NA",
  "isThemeParkAccess": false,
  "themeParkAccessNames": [],
  "isHHNAccess": false,
  "isRTUAccess": false,
  "isGradEventAccess": false,
  "requiresFindEvents": true,
  "productKind": "Ticket",
  "currentPricing": {
    "basePriceData": {
      "pricesByDay": [
        {
          "priceWithTax": {
            "totalPriceWithTax": 85.19,
            "price": 79.99,
            "tax1": 4.8,
            "tax2": 0.4
          },
          "pricingDateTime": "2020-03-02T00:00:00-05:00"
        },
        {
          "priceWithTax": {
            "totalPriceWithTax": 85.19,
            "price": 79.99,
            "tax1": 4.8,
            "tax2": 0.4
          },
          "pricingDateTime": "2020-03-03T00:00:00-05:00"
        },
        {
          "priceWithTax": {
            "totalPriceWithTax": 85.19,
            "price": 79.99,
            "tax1": 4.8,
            "tax2": 0.4
          },
          "pricingDateTime": "2020-03-04T00:00:00-05:00"
        }
      ]
    }
  }
},
```



```
"discountedPriceData": {
  "pricesByDay": [
    {
      "priceWithTax": {
        "totalPriceWithTax": 76.67,
        "price": 71.99,
        "tax1": 4.32,
        "tax2": 0.36,
        "discountType": 1,
        "discountAmount": 10
      },
      "pricingDateTime": "2020-03-02T00:00:00-05:00"
    },
    {
      "priceWithTax": {
        "totalPriceWithTax": 76.67,
        "price": 71.99,
        "tax1": 4.32,
        "tax2": 0.36,
        "discountType": 1,
        "discountAmount": 10
      },
      "pricingDateTime": "2020-03-03T00:00:00-05:00"
    },
    {
      "priceWithTax": {
        "totalPriceWithTax": 76.67,
        "price": 71.99,
        "tax1": 4.32,
        "tax2": 0.36,
        "discountType": 1,
        "discountAmount": 10
      },
      "pricingDateTime": "2020-03-04T00:00:00-05:00"
    }
  ]
},
"futurePricing": {
  "basePriceData": {
    "pricesByDay": [
      {
        "priceWithTax": {
          "totalPriceWithTax": 74.54,
          "price": 69.99,
          "tax1": 4.2,
          "tax2": 0.35
        },
        "pricingDateTime": "2020-03-02T00:00:00-05:00"
      },
      {
        "priceWithTax": {
          "totalPriceWithTax": 74.54,
          "price": 69.99,
          "tax1": 4.2,
          "tax2": 0.35
        },
        "pricingDateTime": "2020-03-03T00:00:00-05:00"
      },
      {
        "priceWithTax": {
          "totalPriceWithTax": 85.19,
          "price": 79.99,
          "tax1": 4.8,
          "tax2": 0.4
        },
        "pricingDateTime": "2020-03-04T00:00:00-05:00"
      }
    ]
  }
},
}}
```



```
"discountedPriceData": {
  "pricesByDay": [
    {
      "priceWithTax": {
        "totalPriceWithTax": 67.09,
        "price": 62.99,
        "tax1": 3.78,
        "tax2": 0.32,
        "discountType": 1,
        "discountAmount": 10
      },
      "pricingDateTime": "2020-03-02T00:00:00-05:00"
    },
    {
      "priceWithTax": {
        "totalPriceWithTax": 67.09,
        "price": 62.99,
        "tax1": 3.78,
        "tax2": 0.32,
        "discountType": 1,
        "discountAmount": 10
      },
      "pricingDateTime": "2020-03-03T00:00:00-05:00"
    },
    {
      "priceWithTax": {
        "totalPriceWithTax": 76.67,
        "price": 71.99,
        "tax1": 4.32,
        "tax2": 0.36,
        "discountType": 1,
        "discountAmount": 10
      },
      "pricingDateTime": "2020-03-04T00:00:00-05:00"
    }
  ]
}
}
```




[/smartorder/MyProductCatalog/flat \(GET\) \(requires Bearer token\)](#)

See [/smartorder/MyProductCatalog](#) – nearly identical request parameters and function, but returns in a (more verbose) flat CSV format. If your integration methodology can support it, Universal suggests using the JSON version of the service if possible, as the payload returned is significantly smaller.

As of this time, the only optional parameter the [/flat](#) endpoint does not implement is `collapseDates`, as that function only significantly reduces the size of the message response in the JSON format.

[/smartorder/CheckUsage \(GET\) \(requires Bearer token\)](#)

This endpoint returns ticket, usage, and modification details for a given list of PLUs, ExternalOrderID, or BookingReferenceID.

Only one parameter can be submitted within each request (for example, an ExternalOrderID and a list of VisualIDs cannot be referenced in a single request). The max number of VisualIDs submitted in the list of VisualIDs is 20, any more will return an error.

tickets.status values

Value	Definition
Valid	Ticket has not been used or modified and is valid for entry. Can be cancelled if purchased within current cancel window.
Expired	Ticket usage window has passed and is not valid for entry. Can be cancelled if purchased within current cancel window.
Modified	Ticket has been upgraded or modified by guest in some way. Cannot be cancelled through SmartOrder.
Used	Ticket has been used by guest. Cannot be cancelled through SmartOrder
Error	Error gathering ticket details. Contact Universal.

Example Request:

<https://qacorpapi.ucdp.net/smartorder/CheckUsage?ExternalOrderId=EOT12345>

Example Response:



```
{
  "success": true,
  "error": {
    "errorCode": 0,
    "errorText": "string"
  },
  "tickets": [
    {
      "visualID": "string",
      "plu": "string",
      "dateSold": "2023-03-09T20:14:40.363Z",
      "expirationDate": "2023-03-09T20:14:40.363Z",
      "status": "Valid",
      "cancellable": true,
      "lastUsage": "2023-03-09T20:14:40.363Z",
      "lastModification": "2023-03-09T20:14:40.363Z"
    }
  ]
}
```



Appendix

Miscellaneous Terms

GalaxyOrderId and ExternalOrderId

ExternalOrderId in the request is the Client's order/booking number with an approved suffix added. This number, when referenced from USH, is the ExternalID. A GalaxyOrderID is an internal order identifier generated at the time of creation. This number shall not be referenced on any collateral the Guest will receive. The only order number presented to consumers should be the ExternalOrderId (or Client order/booking number and approved suffix).

Customer ID

A unique identifier assigned to a Client. References could also include USH Client Number.

Sales Program

Sales Programs are defined by a group of PLUs and associated pricing. Sales Programs are used at a Client level to define allowable products and pricing for those products. Orders may contain PLUs from more than one Sales Program – in these cases, the Sales Program must be specified at the PLU (Order Line) level.

Delivery Method

Delivery method is a way for USH to control where an order can be fulfilled for ticket issuance. For example, enforcing fulfillment at a kiosk or at a Front Gate Ticket Window.

PLU

The PLU is a unique identifier for an Admission Media product. There is a separate PLU for each product-type and its variants i.e. adult and child.

Print at Home Detailed Information

Purchase Path for PRINT AT HOME - [Click Here](#)

Primary Version:

Your tickets will be included in a subsequent e-mail directly from Universal Studios Hollywood to the e-mail provided at the time of purchase. To ensure delivery to your inbox (not bulk or junk folders), please add OnlineStoreConfirmation@nbcuni.com to your address book. Your tickets will be attached to the e-mail as an Adobe PDF file. The attachment may be printed and presented at the front of Universal Studios Hollywood theme park(s) or displayed on your mobile device. Tickets do not include parking or admission to separately ticketed events or venues. Entry subject to biometric scan. Operating hours are subject to change. (633 Characters)

Instant Fulfillment (Print on Web) Detailed Information

Purchase Path for INSTANT FULFILLMENT - [Click Here](#)



In this section are pre-approved templates, Universal Studios Hollywood Logo, and Legal line copy that must be utilized if approved to utilize the Instant Fulfillment (Print on Web) delivery method.

Standard Admission Media Product Template Requirements

The below are the requirements that will be needed on the ticket template for standard Admission Media products that must be customized by Admission Media product type while adhering to all requirements outlined under the “Instant Fulfillment (Print on Web) Requirements” section. Template layouts can be found by [Clicking Here](#).

Authorized Logo

The below logo must appear on all Admission Media products generated.





Accepted Barcode Types

Universal Studios Hollywood's turnstile entrances are currently configured to support the following barcode types:

- 1D Barcodes
- 2D Barcodes
 - PDF-417
 - Data
 - Matric
 - QR
 - Aztec





Error Codes

Note to Developers:

While this section of the document attempts to list out all error codes for all scenarios involved, there are many different downstream systems that each have their own error code schemes. Many of those codes have been consolidated, changed, or combined in order to create the lists below. However, we do surface unexpected codes to the Client, as triaging an issue is always better with more information.

For this reason, expect that there can (and will be) other error codes not listed here. For the most part, we attempt to add text messages to the codes that describe the issue in enough detail to let the Client know the best course of action towards resolution. That's not always possible for codes we did not expect, so do not solely rely on the message only.

Lastly, these codes and messages are typically not meant for end-users – they are meant for Developers. Only surface the details of the error to end-user UIs through a filter/translation of some kind developed on the Client's end (Clients may group many of them into general errors or input/data errors, for example). Log the full error returned from our APIs for development, testing, and triage purposes only.

SmartOrder Response Definitions

Field Name	Description
Success	If true, the order submission was successful and GalaxyOrderId contains the Galaxy Order Number. If false, see either the Error or OrderStatusErrors objects to retrieve the errors.
GalaxyOrderId	Contains USH's Galaxy Order Number if the PlaceOrder request was successful.
Error	This is a GalaxyError object which contains errors that are generally caused by service or connectivity issues.
OrderStatusErrors	This is a collection of GalaxyError object which contain errors that are generally caused by incorrect data in the order objects, such as an incorrect CustomerId or an unavailable PLU or Sales Program.

Galaxy Error Response Definitions

Field Name	Description
ErrorCode	Contains Galaxy's unique code for the error.
ErrorText	Contains Galaxy's error message, giving details about the error that occurred.

GetExistingOrderId

Code	Text	Additional Notes
9900	Unexpected error from downstream service call. Please try again.	Try again or check the data. If it persists, contact an admin.
11603	Order not found or no data available from downstream system	The order does not exist or an error prevented the order data from being retrieved. Try again or check the data.



PlaceOrder

The following codes almost always represent configuration issues – either on the part of the Client or within Universal’s systems. Please review the documentation and correct the values used to match the ones designated by Universal. If the problem persists, see your assigned Universal contact to ensure both parties have configured the agreed-upon values correctly.

Code	Text	Additional Notes
104	PLU not available in customer's sales program	The PLU you entered was not valid
107	Order already exists in Galaxy	
114	VisitDate is either out of range, or has an invalid value	One of the dates you entered was not valid
124	Sales program 862 not found on customer XXXX or its category	One or more SalesProgramIds you entered (either on Order or OrderLine) was not valid
145	Event is not on sale	
146	Event does not have enough quantity available	
147	Event EventID is over.	
150	Item amount is too large for tax table	The amount or quantity of the ticket is too high to properly calculate taxes – use a smaller quantity or choose a different ticket
151	Event not active currently	
1408	PLU not available in sales program	The combination of PLU and SalesProgramId you entered was incorrect
9984	One or more ForeignMediaVisualIds have been specified, but your order configuration does not meet the prerequisites to specify your own VisualID. Data seen: Quantity Per Line = X (should be 1); Delivery Method is Instant Fulfillment = xxxx (should be true)	Confirm with Universal that you are able to supply external VisualIDs. If so, check to make sure that you only have 1 quantity per line, and your Delivery Method is Instant Fulfillment (the error message should tell you which is in error).
9985	PLU could not be matched with a valid price or event with a price	Double check the eventDateTime and eventId in the request. Refer to /findEvents endpoint response for each PLU.
9987	PLU not found	
9991	PLU could not be matched to an appropriate event based on the provided EventDateTime and/or EventId. Please call FindEvents in order to retrieve valid data for use in your order.	
9992	PLU found multiple matching events, but EventId was not provided. Please call	Include eventId in request



	FindEvents in order to retrieve valid data for use in your order and choose exactly one EventId	
9993	PLU found no matching events available. Please call FindEvents in order to retrieve valid data for use in your order	
9994	PLU is event-based and requires an EventDateTime	
9995	PLU could not retrieve event info for creating an order. Please call FindEvents in order to retrieve valid data for use in your order	
9996	Delivery method ID XX is not valid for one or more of the given PLUs	Double check the Delivery method in the request. Refer to /myProductCatalog endpoint response for each PLU.
9997	Print At Home delivery methods require an email address on the order	
9998	If guest names are supplied for an order, the number of guest names must match the quantity	
9999	Can not place SmartOrder without a SPID on either the Order or each of the OrderLines	
30000	Unhandled error or no response from downstream system. This is most typically due to an internal timeout and the order may still be processing. Please use the GetExistingOrderID call to check if the order was placed successfully	



The following codes are system issues that, if repeated, require contacting Universal. In most cases they are not something that can be corrected by the Client, but may be a temporary issue that is resolved if the request is attempted again a few minutes later.

Code	Text	Additional Notes
0	Error in web service attempting to place an order, try again or contact IT.	
123	Source is inactive. Cannot process orders from an inactive source.	Contact USH & report the error
143	PLU (PLU) with a Kind of KindIdentifier (KindName) is not supported on an order line with a DetailType of DetailType	
504	Timeout error	Wait a few minutes and try again
1302	The customer of this ticket is restricted by the source	
9986	No event requests found for order that requires events. Please contact an administrator.	
9989	Order requires taxes but no tax information found due to the following error:	Contact USH & report the error
9990	Customer lookup for ID failed due to the following error:	Contact USH & report the error

CanCancelOrder

Code	Text	Additional Notes
6303	Order cannot be voided, it contains a Pass, for which voiding is not supported	
9000	An error occurred while attempting to read order data from a downstream system. Please try again. The error was:	Read the downstream error and correct if possible. Otherwise, contact USH & report the error.
9000	An error occurred while attempting to read order line data from a downstream system. Please try again.	
11603	Order not found.	
11610	The system could not retrieve transactions for the existing order due to the following error from a downstream service:	Read the downstream error and correct if possible. Otherwise, contact USH & report the error.
11611	The order has one or more transactions created by actions against it, and cannot be voided.	
11620	Error during query for ticket usage.	
11621	Cannot void - one or more tickets have been used.	



CancelOrder

These codes are in addition to the reasons shown in CanCancelOrder – it is suggested you always call CanCancelOrder first.

Code	Text	Additional Notes
116	Cannot cancel order, order is closed	
123	Source is inactive. Cannot process orders from an inactive source.	
504	Timeout error	Try again in a few minutes

FindEvents

Code	Text	Additional Notes
4601	EventID is invalid	Configuration issue – Contact USH & report the error
4602	PLU is invalid	Check the PLU and try again
4603	CustomerID is invalid	Check the data and try again
4604	Error querying events	Try again in a few minutes
20000	No request body provided	
20001	No PLU provided	
20002	No Start Date provided	
20003	No End Date provided	
20004	Product not found or error retrieving product data	
20005	Product is not event-based	
20007	No event data from downstream system	
20008	Unable to retrieve price data for products.	



MyProductCatalog

Note: These endpoints have no dedicated error codes per-se. As they only gather data from various endpoints in the downstream systems, the only error code would be one surfaced from one of those downstream services. As such, there is no numeric error code, and the endpoint only returns back a string error message indicating which piece of the call errored. Here are some typical examples:

Text	Additional Notes
Customer Not Found	The customer you used or were given is in valid. Contact Universal or check your input.
Customer is not configured with any SalesPrograms, or specified SalesProgram is not valid.	The SalesPrograms were misconfigured by Universal, or check your SalesProgramId input
Error calculating taxes for taxTableId X	Contact Universal – tax calculation is not working correctly for the product
One or more errors from a downstream service call:	Generic error – contact Universal with the information
No products found for SalesProgram X	Contact Universal – products are not configured correctly
Error during GetProductCatalog logic:	Generic error – contact Universal with the information

CheckUsage

Code	Text	Additional Notes
5001	Order not found or no data available from downstream system.	The order does not exist, or an error prevented the order data from being retrieved. Try again or check the data.
5003	Please pass in visualIDs or remove from query parameters	Cannot pass in an empty list of visualIDs.
5004	Maximum number of VisualIDs per request is 20. Please split the request or use External Order ID or Booking Reference ID in request.	
5005	Tickets not found or no data available from downstream system.	The tickets do not exist, or an error prevented the ticket data from being retrieved. Try again or check the data.
9900	Unexpected error from downstream service call. Please try again.	Try again or check the data. If it persists, contact an admin.

----- END OF DOCUMENT -----