



UPT TRADE SERVICES

SMARTORDER X-TRAINING



# SMARTORDER OVERVIEW

## **Description**

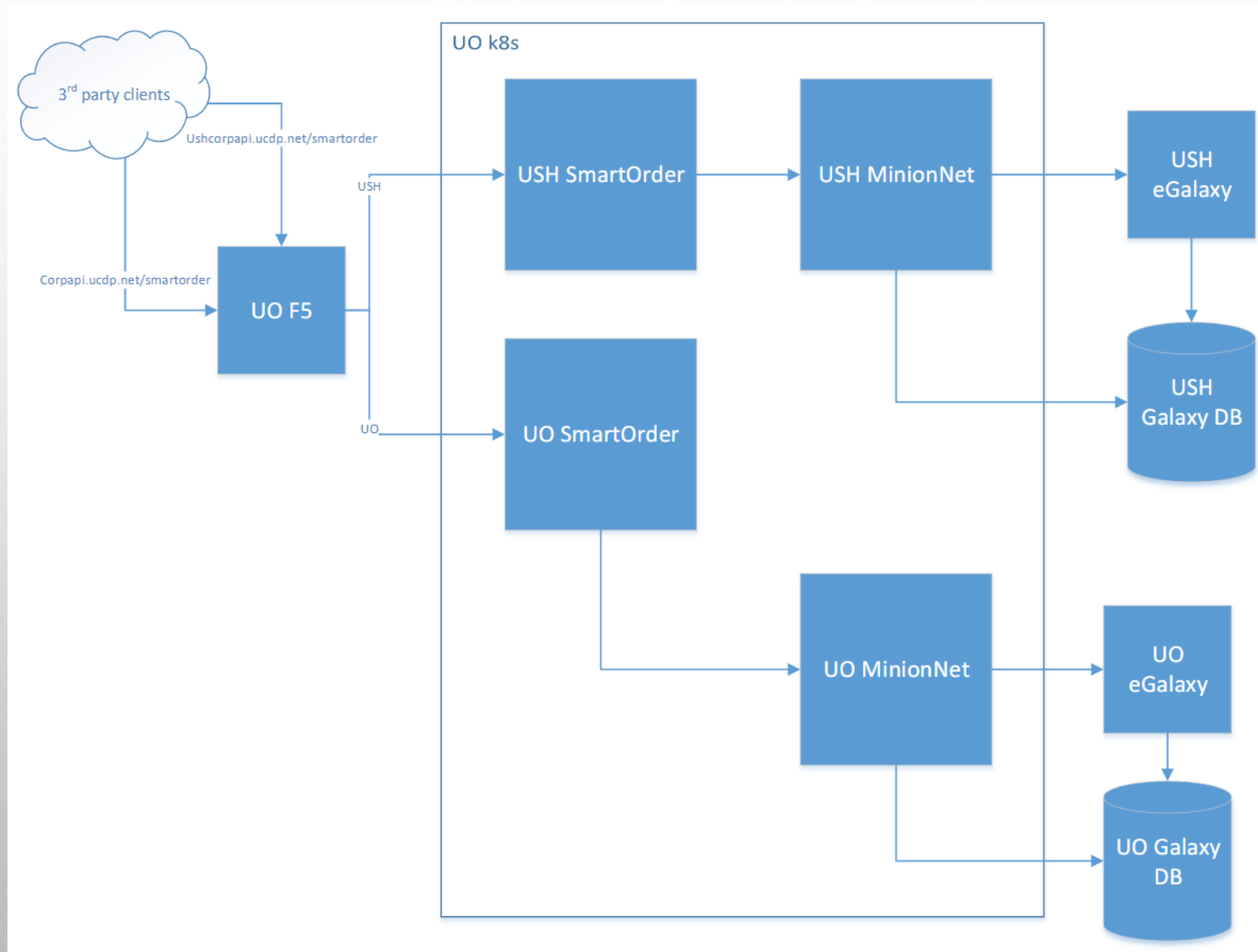
The SmartOrder platform is an API system that uses a custom web service, called Minion, which connects clients to the eGalaxy server. It allows 3rd party partners to sell Universal Orlando and Universal Hollywood tickets to their customers directly from their websites or POS systems.

## **Technology**

SmartOrder is a REST API built in .NET Core and is hosted on a Kubernetes cluster.



# SYSTEM ARCHITECTURE





# SOFTWARE AND APPLICATIONS FOR SUPPORT

## Development

Gitlab - Source code repo - <https://gitlab.Use.Ucdp.Net/corpapi/smartorder>

## Documentation

Swagger - <https://gitlab.use.ucdp.net/CorpAPI/SmartOrder>

WIKI - <HTTPS://WIKI.INBCU.COM/PAGES/VIEWPAGE.ACTION?PAGEID=281808876>

## Management

Rancher – Kubernetes management - <https://usorancher01.use.ucdp.net/login>

CorpAPI Manager – To manage users

1. <HTTP://10.18.74.53:30082/HOME/LOGIN?RETURNURL=%2F> - UO QA
2. <HTTP://10.18.74.53:30182/HOME/LOGIN?RETURNURL=%2F> - USH STAGE
3. <HTTP://10.18.78.72:30082/HOME/LOGIN?RETURNURL=%2F> - UO PROD
4. <HTTP://10.18.78.72:30182/HOME/LOGIN?RETURNURL=%2F> - USH PROD

## Testing and Demo

Insomnia – API platform with friendly UI - <https://insomnia.rest/download>

Postman – API platform with advanced testing functionality - <https://www.postman.com/downloads/>



# REPORTING AND TROUBLESHOOTING

Ptktreporting – Custom reports -

<https://ptkreporting/Reports/Pages/Folder.aspx?ItemPath=%2fETDS+and+Genesis%2fUO+SmartOrder&ViewMode=List>

- We currently have only 1 report (as of 08/10/2022) for looking up identical orders from clients.

Elastic (formerly called Kibana) – For troubleshooting - <https://elastic.use.ucdp.net/app/discover>

- The “Discover” and “APM” views are the 2 most used views for troubleshooting issues with SmartOrder.





# COMMON REQUESTS – NEW CLIENT CREDENTIALS



## Request tracking

- Jira  
<https://jira.inbcu.com/secure/RapidBoard.jspa?rapidView=19628>
- Jira is used to keep track of SmartOrder development projects and new credential requests. Whoever works on a Jira ticket, must log their time spent on it in Jira plus move the ticket(s) from “TO DO” to “IN PROGRESS”, and to “DONE” as needed.

## Procedure:


- Wiki  
<https://wiki.inbcu.com/pages/viewpage.action?pageId=289394379>
- The wiki shows the following
  1. Creating a new eGalaxy source for new client
  2. Assigning customer ID(s) to new eGalaxy source
  3. Configuring new eGalaxy source
  4. Creating new user in CorpAPI Manager for new client
  5. Sending and saving new credentials


Atlassian Jira (UPTC-4611) Universal Orlando - SmartOrder Credentials - World of DVC


 Richardson, Naz (Jira) <jira.issues@nbcuni.com>  
To  Rangel, Christopher (NBCUniversal, Parks and Resorts) Fri 8/5/2022 12:35 PM



Retention Policy MBX Delete after 5 years (5 years) Expires 8/4/2027

**NEW TICKET**

 Follow up. Completed on Friday, August 5, 2022.  
You forwarded this message on 8/5/2022 3:12 PM.  
If there are problems with how this message is displayed, click here to view it in a web browser.

 Richardson, Naz created an issue

UPT Core /  UPTC-4611  
**Universal Orlando - SmartOrder Credentials - World of DVC**

**Issue Type:**  Story  
**Assignee:** Rangel Almazan, Christopher  
**Components:** SmartOrder  
**Created:** 05/Aug/22 12:34 PM  
**Priority:**  Low  
**Reporter:** Richardson, Naz

Please create credentials for Universal Orlando in QA and Production for the following Account:

Client Name: World of DVC  
Client #: 133989

**Request Example**



# INSOMNIA - SETTING UP SMARTORDER TESTING ENVIRONMENTS

1. **Install insomnia (default install):**  
insomnia.rest/download
2. **Import Trade Services's SmartOrder Insomnia file:** \OneDrive - uniparks\Trade Services\SmartOrder
3. **Install KeePass:**  
<https://keepass.info/download.html>
4. **Open Trade Services's KeePass file:**  
\OneDrive - uniparks\Trade Services
5. **Get Token (must renew once every hour as needed):** Use the usernames and passwords for “\_SmartOrder Test” found in KeePass to use the Token endpoint in Insomnia for the different environments
6. **Set the token variable:** The token can be set as a variable within each environment, so that you don't need to get a new token for each endpoint separately

The screenshot shows the Insomnia application interface with four numbered callouts indicating the steps for setting up the PROD environment:

- 1:** A red box highlights the 'PROD' environment selector in the top left corner.
- 2:** A red box highlights the 'Manage Environments' button in the bottom left corner of the 'ACTIVATE ENVIRONMENT' panel.
- 3:** A red box highlights the 'PROD' environment in the 'Sub Environments' list.
- 4:** A red box highlights the 'token' variable in the 'PROD' environment's JSON body, which is set to '""'.

The JSON body for the PROD environment is shown as follows:

```
1 {  
2   "base_url": "https://corpapi.ucdp.net",  
3   "token": "",  
4 }
```

Setting token as variable for PROD



# TESTING API

## After deployment

If you are testing the API after a feature or bug fix deployment, you can use **client\_id “EO-SmartOrderTest”** and **salesProgram 3746**.

## For specific client

If testing a specific client, you can find username (client\_id) and password (client\_secret) in KeePass

## Procedure

- To test the API, we use either Insomnia and Postman to make a call to each of the following endpoints (in the order listed):
  1. SmartOrder
  2. Token
  3. MyProductCatalog
  4. FindEvents (test with dated product)
  5. PlaceOrder (test with dated, non-dated, and mixed)
  6. GetExisting OrderId
  7. CanCancel
  8. Cancel
- All endpoints should return **“200 OK”** and expected results. Step by step validation/testing instructions are available in the “Smart Order Technical Continuity Plan”.
- Make sure to cancel all test orders





# COMMON REQUESTS – DEMO FOR NEW CLIENTS

## Software

- Insomnia or Postman

## Procedure

- Clients are given the latest Onboarding Guide by the sales team, and the guide includes detailed information about all the API endpoints. The guide can also be found in the SmartOrder WIKI
- An onboarding meeting is scheduled for clients to ask questions which we usually answer by showing the API's functionality, and examples of API requests and responses

## Common questions

**Q:** How often should the client pull their product catalog?

**A:** At least once a day

**Q:** What delivery method should the client use for their products?

**A:** There are multiple delivery methods that can be used depending on the product. The client can use any of the available delivery methods for a specific product, except “instant fulfillment” (92). In order for the client to use “instant fulfillment” (92), they must be approved to use it and get their admission media template approved as well by the sales team.

**Q:** For how long is the token good for?

**A:** The token's default timeout is 1 hour. The client can either request a new token for each order or renew their token before it times out.



# COMMON REQUESTS – DEMO FOR NEW CLIENTS

## Common questions - Continued

**Q:** When should the FindEvents endpoint be used?

**A:** The FindEvents endpoint must be called for all data-based or capacity-managed products (AKA dated products). The MyProductCatalog's response includes the boolean attribute "**requiresFindEvents**": for every PLU, and if the attribute's value is true for the PLU in question, then FindEvents must be used for that PLU prior to using PlaceOrder.

**Q:** What price should be used?

**A:** MyProductCatalog includes basePriceData and discountedPriceData for both currentPricing and FuturePricing. The basePriceData are the suggested retail rates and the discountedPriceData are the net rates (what the client will be charged). Clients usually either use the basePriceData rates as their rates for their customers OR use the basePriceData + a markup.

**Q:** What if products are missing from MyProductCatalog?

**A:** The sales team would arrange for the products to be added as needed. The UPT Trade Services team does not add or modify products.

**Q:** What is the externalOrderId?

**A:** A unique orderId created by the client that includes the approved client's suffix (such as "-2AFFRD")

**Q:** Should the price from FindEvents be used?

**A:** If salesProgramId IS NOT included in the request, FindEvents returns the base prices (suggested retail rates). If salesProgramId IS included, FindEvents returns the discounted prices (net rates)



# COMMON ERRORS

## MyProductCatalog

1. Client uses the wrong customerId. Make sure that the client is using one of the customerIds assigned to them.
2. Client does not include dates in the request. The parameters startDateInclusive and endDateInclusive **are required** in the MyProductCatalog request for Price Scheduled products to show up correctly.
3. Client does not see discounted pricing in MyProductCatalog because they are pulling only base prices. Clients can use the parameter **pricing=** in their request, but it is not required. If **pricing=base** is used, then only base prices are included in the response, and likewise if **pricing=discounted**, then only discounted prices are included. If the parameter is not included, then all prices are included.

## FindEvents

1. Client does not see discounted pricing in FindEvents because they are pulling only base prices. Clients can use the parameter **"salesProgramId"** in their JSON request, but it is not required. If **"salesProgramId"** is provided, then the response returns discounted rates (net rates), and if not, then it returns base rates (suggested retail prices).

## PlaceOrder

1. Unable to place an order. Make sure that the client is using an EventDateTime exactly as shown in the FindEvents response for a dated PLU (formatting example: "2022-08-09T04:00:00-04:00"). The correct customerId, salesProgramId, and available deliveryMethodId should also be used.
2. Timeout error. It is recommended for clients to **increase their timeout to 180 seconds for all requests**. If they encounter a timeout error when using PlaceOrder, they should use GetExistingOrderId to check if the order made it through or not. If they want to resubmit the order, they should use the same externalOrderId to prevent duplicate orders.



# COMMON ERRORS

## **CanCancelOrder**

1. CanCancelOrder responding false. Many 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.

## **CancelOrder**

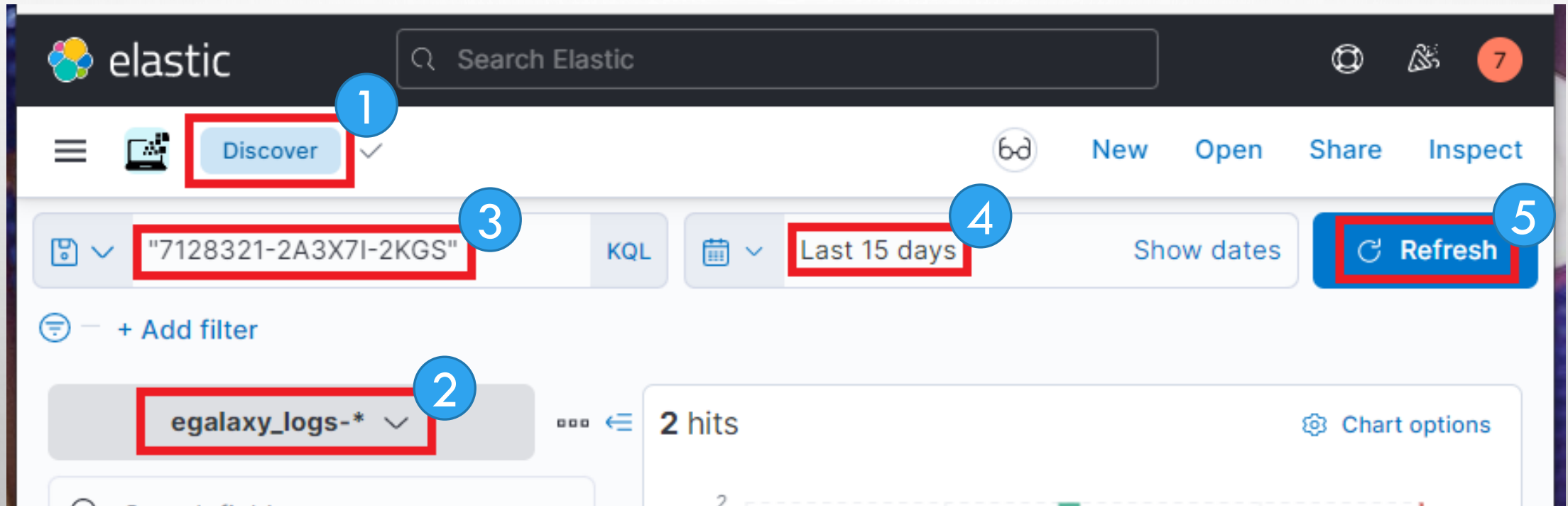
1. CanCancelOrder responding false. Many modifications to the order by Universal may prevent it from being cancelled from this API, including usage of the tickets within. Client can use CanCancelOrder before calling CancelOrder.



## COMMON REQUESTS – CHECK IF ORDER WENT THROUGH

Elastic is used to lookup orders and possible errors.

Select the following settings to lookup an order:







## COMMON REQUESTS – CHECK IF ORDER WENT THROUGH

You should be able to see both the request and the response. If the response includes the GalaxyOrderID, then it means that the order made it to Galaxy.

Time ↓	Document
> Aug 2, 2022 @ 13:12:01.209	message: 309 s 17F8 Server Thread 41 <b>Response:</b> <?xml version="1.0" encoding="UTF-8"?> <Envelope><Header><MessageID>0</MessageID><MessageType>SetOrderStatus</MessageType><SourceID>E0-Kissimmee Guest Services</SourceID><TimeStamp>2022-08-02 13:06:54</TimeStamp><EchoData>0000000000000000a83941d289758ded</EchoData><SystemFields/>
> Aug 2, 2022 @ 13:06:51.408	message: 135 us 17F8 Server Thread 41 <b>Request:</b> <Envelope> <Header><SourceID>E0-Kissimmee Guest Services</SourceID><MessageType>Orders</MessageType><EchoData>0000000000000000a83941d289758ded</EchoData> </Header> <Body><Orders> <Order> <OrderID>7128321-2A3X7I-2KGS</OrderID>



## COMMON REQUESTS – CHECK IF ORDER WAS SUCCESSFULLY CANCELLED

1. Microsoft SQL Server Management Studio (SSMS) can be used to lookup the order in Galaxy (Server: PDBGLXYRPT1, Database: Galaxy)
2. Verify that the OrderAmt is 0 and that the quantity of product-related OrderLines is 0

```
SELECT ol.OrderID, o.ExternalID, o.OrderAmt, sum(ol.Quantity) AS TotalPLUOrderLines FROM
OrderLines ol (NOLOCK)
JOIN Orders o (NOLOCK) on o.OrderID = ol.OrderID
WHERE o.ExternalID = '[ExternalID]' AND ol.PLU <> ''
Group by ol.OrderID, o.ExternalID, o.OrderAmt
```