



# CorrigoPro Direct API

*Getting Started Guide*

## Contents

<b>INTRODUCTION .....</b>	<b>3</b>
<b>CONFIGURATION OF CORRIGOPRO DIRECT.....</b>	<b>5</b>
<b>BECOME FAMILIAR WITH CORRIGOPRO .....</b>	<b>8</b>
<b>SCOPING YOUR INTEGRATION .....</b>	<b>8</b>
<b>BECOMING FAMILIAR WITH CORRIGOPRO DIRECT SUITE .....</b>	<b>9</b>
THE CORRIGOPRO DEVELOPER HUB.....	9
CORRIGOPRO DIRECT API SPECIFICATION .....	9
<b>AUTHORIZATION MODEL.....</b>	<b>9</b>
REQUESTING A TOKEN .....	10
<b>MAKING YOUR FIRST API CALLS .....</b>	<b>11</b>
GET /API/CUSTOMERS.....	11
GET /API/CUSTOMERSETTINGS.....	12
<b>RECEIVING NEW WORK ORDERS.....</b>	<b>13</b>
EVENT NOTIFICATION.....	13
POLLING FOR WORK ORDERS.....	13
<i>Searching for work orders.....</i>	<i>13</i>
<i>Getting work order details.....</i>	<i>14</i>
<b>CONFIGURING YOUR ENDPOINT .....</b>	<b>15</b>
REQUEST STRUCTURE.....	16
<i>Message Header .....</i>	<i>16</i>
<i>Request Body .....</i>	<i>16</i>
TOKEN VALIDATION .....	17
<b>PROCEED WITH IMPLEMENTATION .....</b>	<b>17</b>
TOOLS.....	18
<i>Event Notification Emulator.....</i>	<i>18</i>
<i>Web Services Log .....</i>	<i>20</i>
<i>Event Notification Retry Queue.....</i>	<i>21</i>
<i>Test Work Orders .....</i>	<i>23</i>
<b>GOING LIVE.....</b>	<b>23</b>
GOING LIVE FOR WORKTRACK USERS.....	23
<b>APPENDIX: FAQ.....</b>	<b>25</b>
<i>What do I use for a MessageId? .....</i>	<i>25</i>
<i>What if there is an outage and I'm temporarily unable to receive requests?.....</i>	<i>25</i>

<i>How do I view the history of incoming and outgoing messages?</i> .....	25
<i>How do I submit an invoice?</i> .....	25
<i>Which fields are optional and which are required?</i> .....	25
<i>Getting Customer Location Data</i> .....	25
<i>I have another question...</i> .....	25
<b>APPENDIX: COMMON ERRORS</b> .....	<b>27</b>
<i>Status: 400 (Bad Request)</i> .....	27
<i>Status: 401 (Unauthorized)</i> .....	28
<i>Status: 500 (Internal Server Error)</i> .....	29

## Introduction

Welcome to the CorrigoPro Direct API Suite!

This suite is comprised of powerful tools and comprehensive resources that can be used together to create a successful integration to connect your work order management system to the CorrigoPro Network.

The scope of this guide is to give an overview of the resources and tools available and walk you through the basics of you first interactions with the API.

This guide assumes the following prerequisites:

- Subscription to CorrigoPro Direct and Sandbox access**

If you are not subscribed to CorrigoPro Direct, log into your CorrigoPro Desktop and select the *CorrigoPro Direct* tile. A Corrigo representative will contact your company to discuss your integration needs.
- Familiarity with JSON and cURL**

You must be familiar with JavaScript Object Notation (JSON) which is used by CorrigoPro Direct. You must be able to translate JSON to the standard that you are using, if applicable.

You must also be familiar with a tool that can be used to for execution of REST requests. In this document, cURL ([curl.haxx.se](http://curl.haxx.se)) will be used in some examples. Postman ([getpostman.com](http://getpostman.com)) is another popular tool that can be used to assist with integration development and testing.

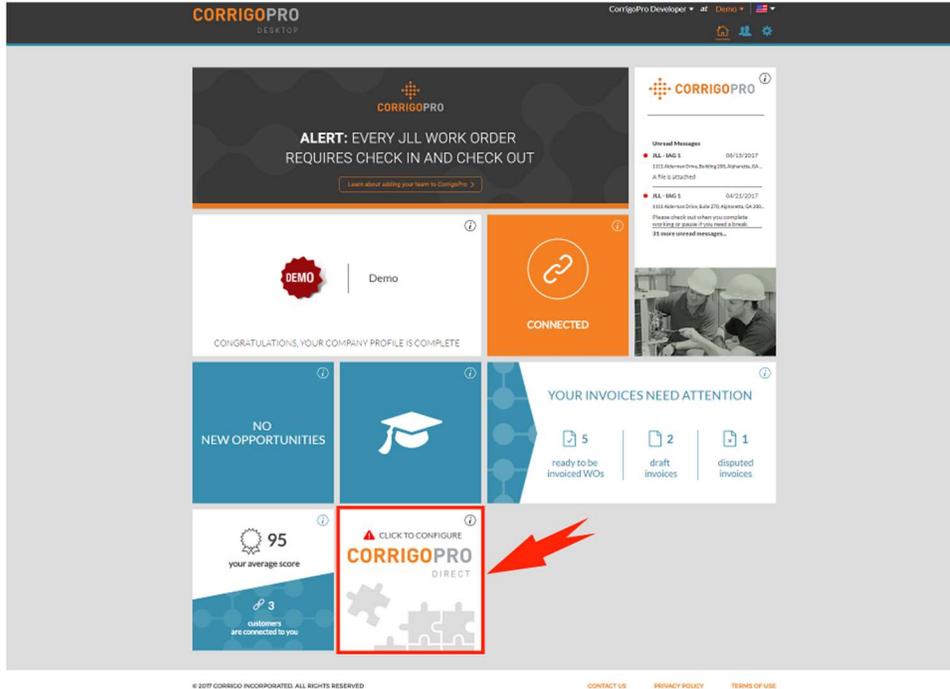
- **Familiarity with OAuth 2.0**

CorrigoPro Direct uses OAuth 2.0 authentication. You should be familiar with the basic concepts, settings, and workflow.

## Configuration of CorrigoPro Direct

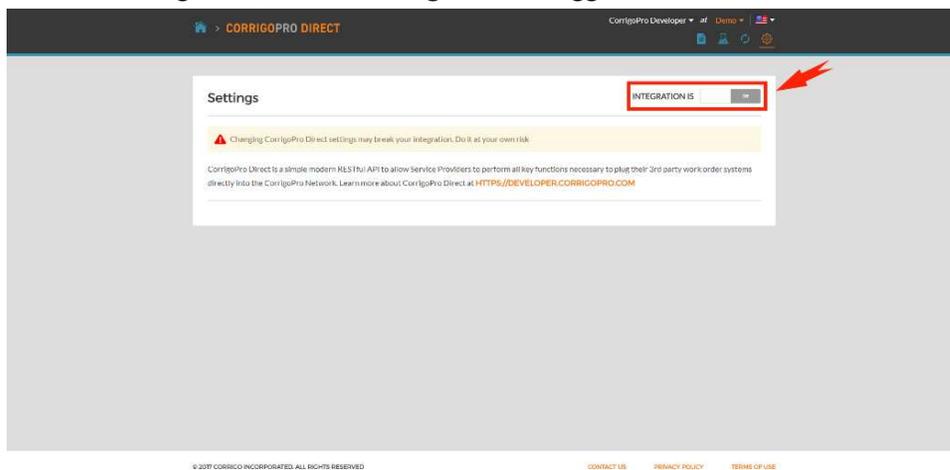
To configure integration with CorrigoPro Direct, perform the following steps:

1. Select the CorrigoPro Direct tile from the CorrigoPro Desktop home page



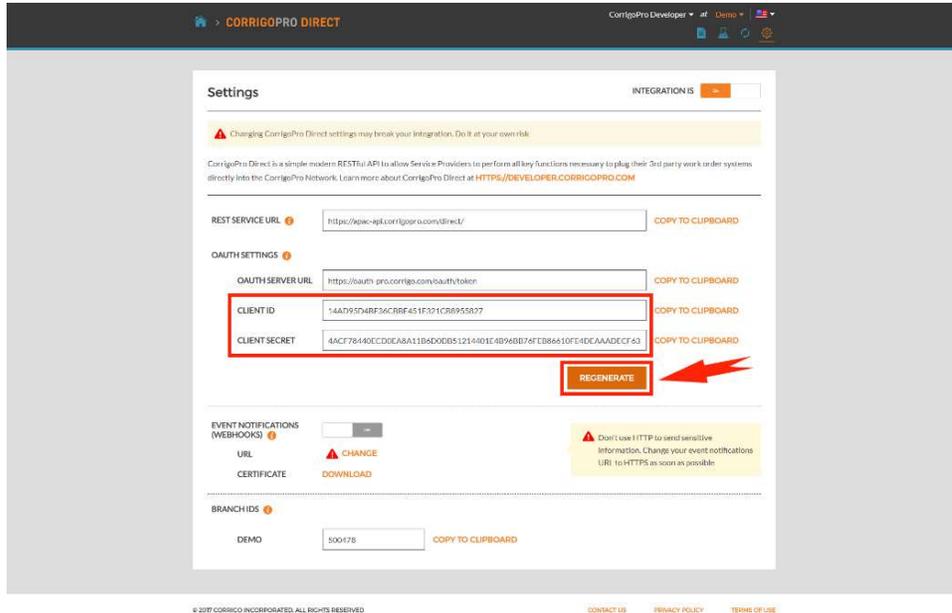
2. Enable Integration

To enable integration, turn the **Integration** is toggle to ON.



### 3. Generate client credentials

Click **Regenerate** to generate the OAuth credentials. You must configure Authorization of your application to be able to send requests to the CorrigoPro Direct API.

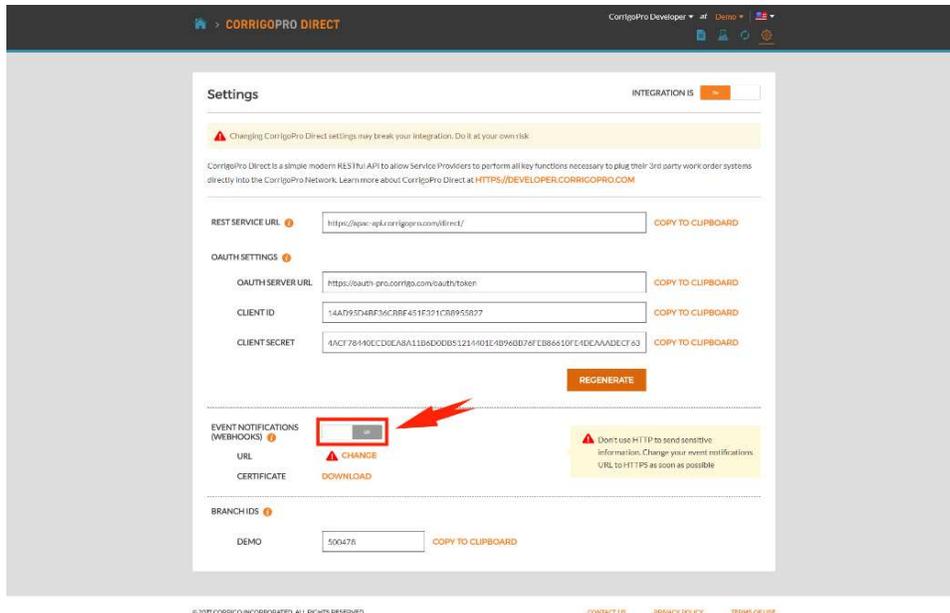


### 4. Configure WebHooks (Optional)

Configure Webhooks for CorrigoPro Direct to receive updates.

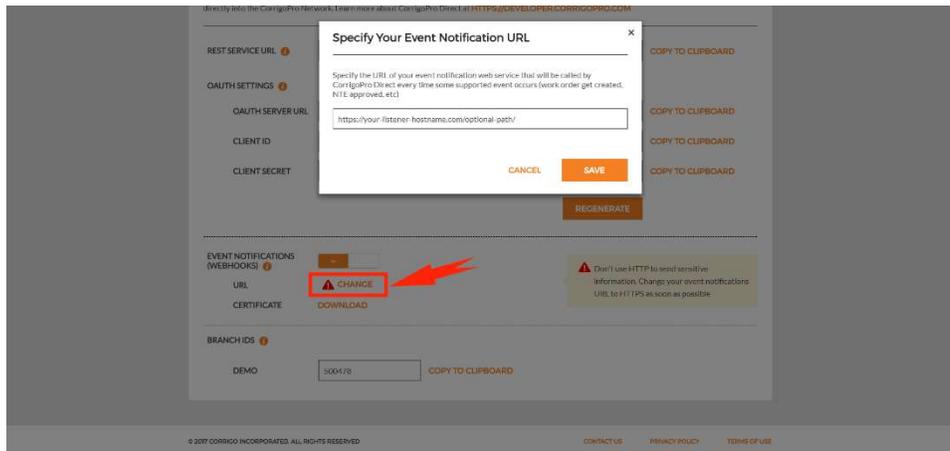
#### 4.1 Enable Webhook Notifications

Turn the Event Notifications (Webhooks) toggle to ON:



## 4.2 Specify your event notification URL

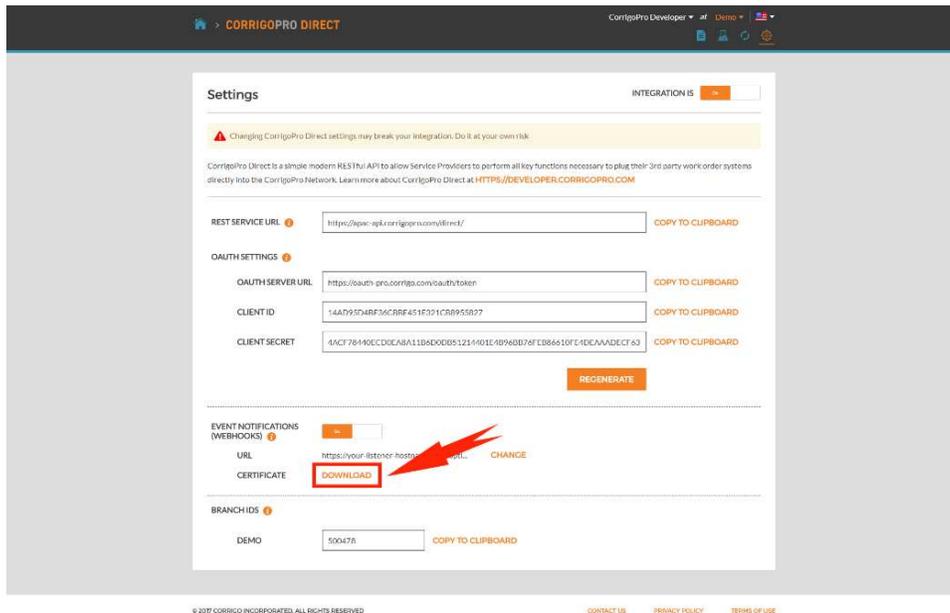
Verify the URL is valid with protocol definition. It is strongly recommended that you use require HTTPS and protect your endpoint with a certificate.



## 4.3 Configure authorization (optional)

Configure authorization for you WebHooks endpoint to be sure that the requests that you will receive, will be sent from the CorrioPro Direct.

You may download the public certificate from the link shown below and see **Token validation** for additional details.

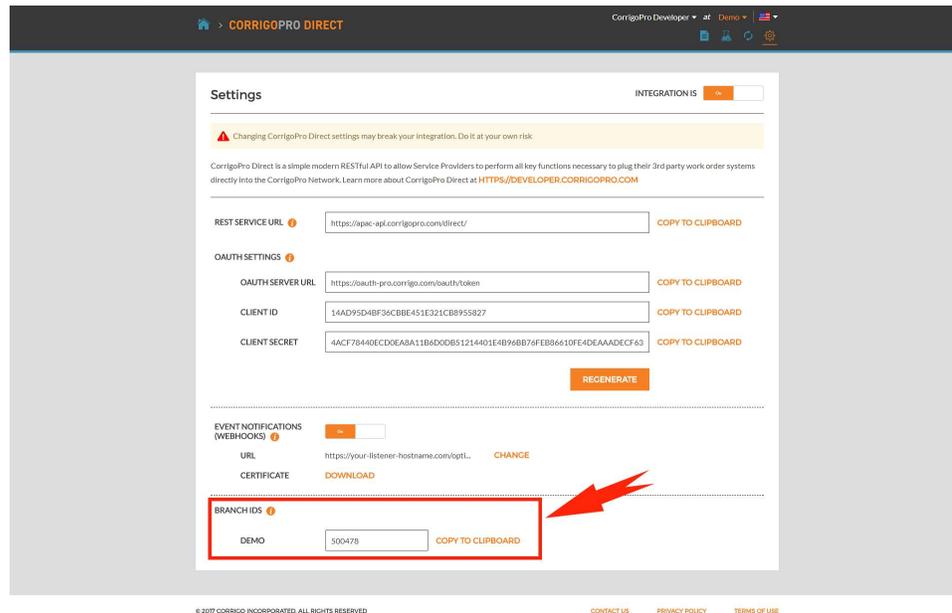


## 4.4 Validate configuration

You can validate your WebHooks configuration by using **Event Notification Emulator**.

## Obtain your Branch Ids

CorrigoPro Desktop application supports multi-branch companies. Some methods of CorrigoPro Direct API may require a **BranchId** for your requests. These Ids may also be used to indicate a branch recipient for work order details or a WebHook notification.



The screenshot shows the 'Settings' page for 'CORRIGOPRO DIRECT'. It includes a warning about integration, a 'REST SERVICE URL' field, 'OAUTH SETTINGS' with fields for 'OAUTH SERVER URL', 'CLIENT ID', and 'CLIENT SECRET', 'EVENT NOTIFICATIONS (WEBHOOKS)' with 'URL' and 'CERTIFICATE' options, and a 'BRANCH IDS' section. The 'BRANCH IDS' section has a table with one entry: 'DEMO' with a value of '500478' and a 'COPY TO CLIPBOARD' button. A red box and arrow highlight this section.

## Become familiar with CorrigoPro

It is recommended that you familiar with workflow and basic functionality of CorrigoPro Desktop before beginning your integration. Contact us if you need to schedule a CorrigoPro training session.

## Scoping your integration

The workflow of a work order within CorrigoPro comprises of six key steps. The first step of scoping your integration should be to confirm which steps your company performs. Then you need to decide which will be executed via the API, which will be performed through CorrigoPro Desktop, and which will be performed using Mobile apps.

These six key steps are:

- Receive work orders
- Accept / Reject work orders
- Check In
- Complete work orders
- Submit quotes
- Invoices

The CorrigoPro Direct Scoping Work Sheet can assist you with these determinations.

## Becoming familiar with CorrigoPro Direct Suite

To create a successful CorrigoPro Direct integration, it is recommended that you become familiar with the CorrigoPro Direct API capabilities and the tools offered by the API suite.

### The CorrigoPro Developer Hub

The CorrigoPro Developer Hub is the most comprehensive source to assist you with your integration.

This resource includes information on the following:

- The work order lifecycle
- Authorization
- Documentation for all API methods and event notifications
- Data Models
- Tools (WebHook Emulator, Communication logs, and Event Notification Retry Queue)

The CorrigoPro Developer Hub is accessible from the following URL:

<https://developer.corrigopro.com/>

### CorrigoPro Direct Swagger API documentation

The CorrigoPro Direct API offers documentation through Swagger (<https://swagger.io>). The Swagger API specification is another key reference source. It summarizes each action offered by the API, documents the model, and provides example requests and responses.

The CorrigoPro Direct API specification is accessible from the URL below:

<http://api.corrigopro.com/documentation/>

## Authorization model

CorrigoPro Direct uses OAuth 2.0 authorization flow to obtain a bearer token to authorize API requests. Every call to this API must specify the *Authorization* HTTP header that includes the bearer token retrieved from the CorrigoPro OAuth server.

#### HTTP Header

```
Authorization: Bearer {token}
```

## Requesting a token

To obtain a Bearer Token, send a POST request to the OAuth Server.

This request should include the following:

- OAuth Server URL: The OAuth Server URL provided in **CorrigoPro Direct Settings** (see **Configuration of CorrigoPro Desktop**, Step 3).
- Header parameter: "Content-Type: application/x-www-form-urlencoded"
- Parameters:
  - **client\_id**: your client id provided in CorrigoPro Direct Settings.
  - **client\_secret**: your client secret provided in CorrigoPro Direct Settings.
  - **grant\_type**: constant value, set as "client\_credentials".

The following examples demonstrate requests as a cURL command and an HTTP request:

### cURL

```
curl -i -X POST \  
  <OAuth_Server_URL> -H "Content-Type: application/x-www-form-urlencoded" \  
  -d "grant_type=client_credentials&client_id=<Your_Client_Id>&client_secret=<Your_Client_Secret>"
```

### HTTP

```
POST /OAuth/Token HTTP/1.1  
Host: <OAuth_Server_URL>  
Cache-Control: no-cache  
Content-Type: application/x-www-form-urlencoded  
  
grant_type=client_credentials&client_id=<Your_Client_Id>&client_secret=<Your_Client_Secret>
```

A valid response to your OAuth request should be like:

### HTTP

```
Cache-Control: no-cache  
Pragma: no-cache  
Content-Type: application/json;charset=UTF-8  
Expires: -1  
Server: Microsoft-IIS/8.5  
X-Powered-By: ASP.NET  
Date: Tue, 10 Apr 2018 19:37:59 GMT  
Content-Length: 1543
```

```
{"access_token":"eyJBdXRoW50aNhdIvbl...<---->OOoH_cGiaJ3jH... -9oO9-vuPlh0Fiu75g_3jPubwDra-qAEcM4HCd8eKh9dg","token_type":"bearer","expires_in":1199}
```

\* *Note: Access Token value abbreviated*

The response includes the following information:

- **access\_token**: A token to be used to authorize you to make API calls.
- **token\_type**: The type of token being returned, which will be “bearer”.
- **expires\_in**: The number of seconds that the token will be valid.

These values will be important once you make calls to the REST Service URL.

## Making your first API calls

Now that you have obtained an `access_token`, you can use that token to perform an API method.

### GET /api/customers

Using the **GET /api/customers** method is a good choice to start with. It is a simple method that is a good choice for validating the proper sending of bearer tokens and passing a proper **messageId**.

To perform this request, you will need the following:

- Your region’s REST Service URL provided in **CorrigoPro Direct Settings** (see **Configuration of CorrigoPro Direct**, Step 3).

These include:

- <https://apac-api.corrigopro.com/Direct/api/> for Asia/Pacific regions
- <https://am-api.corrigopro.com/Direct/api/> for Americas
- <https://emea-api.corrigopro.com/Direct/api/> for Europe, the Middle East, and Africa.
- A valid non-expired **access\_token** (see **Requesting a token**).
- A **messageId** value. It is recommended that this value be unique, and can be generated from GUIDs, timestamps, using prefixes or suffixes, or otherwise as meets your needs.

Below are examples provided in the form of a cURL command and an HTTP request:

#### cURL

```
curl -i -X GET \  
'https://<Your_Region>-api.corrigopro.com/Direct/api/customers/?messageId=<Message Id>' \  
-H 'Authorization: Bearer <Your Bearer Token>'
```

**HTTP**

```
GET /Direct/api/customers/?messageId=<Message Id> HTTP/1.1  
Host: <Your_Region>-api.corrigopro.com  
Authorization: Bearer <Your Bearer Token>
```

This may result in a response like:

**HTTP**

```
Content-Type: application/json; charset=utf-8  
Server: Microsoft-IIS/8.5  
X-Powered-By: ASP.NET  
Date: Tue, 10 Apr 2018 20:40:02 GMT  
Content-Length: 127  
  
{  
  "Customers": [{  
    "BranchIds": [101138, 101470],  
    "Id": 100160,  
    "Name": "APAC Demo"  
  }],  
  "MessageId": "e591f654-f565-4c82-b5e5-bbc9dbbd103a"  
}
```

If you encounter an error, see [Appendix: Common Errors](#) for possible causes.

## GET /api/customerSettings

Once you have successfully retrieved a list of connected customers, you can use a returned customer id value to follow up with a **GET /api/customerSettings request**.

Periodic use of this method will be useful to you in the future for obtaining customer-specific values for custom fields, repair codes, and repair categories which may be required for completing a work order and meeting customer requirements.

Below are examples of cURL and HTTP requests:

**cURL**

```
curl -i -X GET \  
'https://<Your_Region>-api.corrigopro.com/Direct/api/customerSettings/?messageId=<Message Id>  
&customerId=<customer_id>' \  
-H 'Authorization: Bearer <Your Bearer Token>'
```

**HTTP**

```
GET /Direct/api/customerSettings/?messageId=<Message Id>&customerId=100160 HTTP/1.1  
Host: <Your_Region>-api.corrigopro.com  
Authorization: Bearer <Your Bearer Token>
```

## Receiving New Work Orders

There are two different methods for being notified of new work orders. The first is to use the API to periodically search for new work orders. The second method, and the recommended one, is to expose an endpoint for CorrigoPro Direct to send event notifications.

If you decide to use the recommended integration model of using Event Notification, you still may wish to review [Polling for work orders](#) below as an example of using a POST method with JSON data with the API.

### Event Notification

Event Notifications (WebHooks) provide you with a way to be notified of new work orders (and updates to existing work orders) in real-time.

See [Configuring your endpoint](#) for more information.

### Polling for work orders

While Event Notifications are the preferred method for receiving new work orders, you can also choose to poll. You can achieve this by using the work order search method. You should typically search for all work orders with a “New” status created within the last  $n$  minutes.

### Searching for work orders

Submit a **POST /api/workOrder/search** request to receive basic information about work orders that match the criteria included in the JSON body. A common usage would be to search for work orders with a “**New**” status created within the last  $n$  minutes.

Below are examples provided in the form of a cURL command and an HTTP request:

#### cURL

```
curl -i -X GET \  
https://<Your_Region>-api.corrigopro.com/direct/api/workOrder/search/ \  
-H 'Authorization: Bearer <Your Bearer Token>' \  
-H 'Content-Type: application/json' \  
-d '{  
  "Parameters": {  
    "Created": {  
      "From": "2018-03-26T18:01:24.000Z",  
      "To": "2018-03-26T18:01:29.000Z"  
    },  
    "Statuses": [ "New"]  
  },  
  "MessageId": "<Message Id>"  
}'
```

#### HTTP

```
POST /direct/api/workOrder/search/ HTTP/1.1  
Host: <Your_Region>-api.corrigopro.com  
Authorization: Bearer <Your Bearer Token>  
Content-Type: application/json  
Cache-Control: no-cache
```

```
{  
  "Parameters": {  
    "Created": {  
      "From": "2018-03-26T18:01:24.000Z",  
      "To": "2018-03-26T18:01:29.000Z"  
    },  
    "Statuses": [ "New"]  
  },  
  "MessageId": "<Message Id>"  
}
```

For additional information regarding the response model and additional parameter options for **GET /api/workOrder/search**, see:

[http://api.corrigopro.com/documentation/#!/WorkOrder/WorkOrder\\_Search](http://api.corrigopro.com/documentation/#!/WorkOrder/WorkOrder_Search)

### Getting work order details

You can receive detailed information about work orders if you know their IDs by submitting a **GET /api/workorder/** request.

Please notice that a work order Id is distinct from a work order number.

This request should include a **messageId** parameter and one or more work order **ids** parameters as query strings.

### *A note regarding work order IDs vs work order numbers*

It is important to note that a work order Id is distinct from a work order number. **GET /api/workorder/** must use **Ids**, not work order numbers. **Ids** will always be unique within your system. Work order numbers are not guaranteed to be unique, especially between multiple customers.

## Configuring your Endpoint

Enabling Event notifications allows you to receive new work orders and updates to existing work orders (including SLA changes) as quickly as possible.

To use Event Notifications, you must do the following:

- 1) Configure an accessible endpoint to act as a receiver for Event Notifications.
- 2) Provide the URL for the endpoint within CorrigoPro Direct Settings.
- 3) Enable Event Notifications within CorrigoPro Direct Settings.

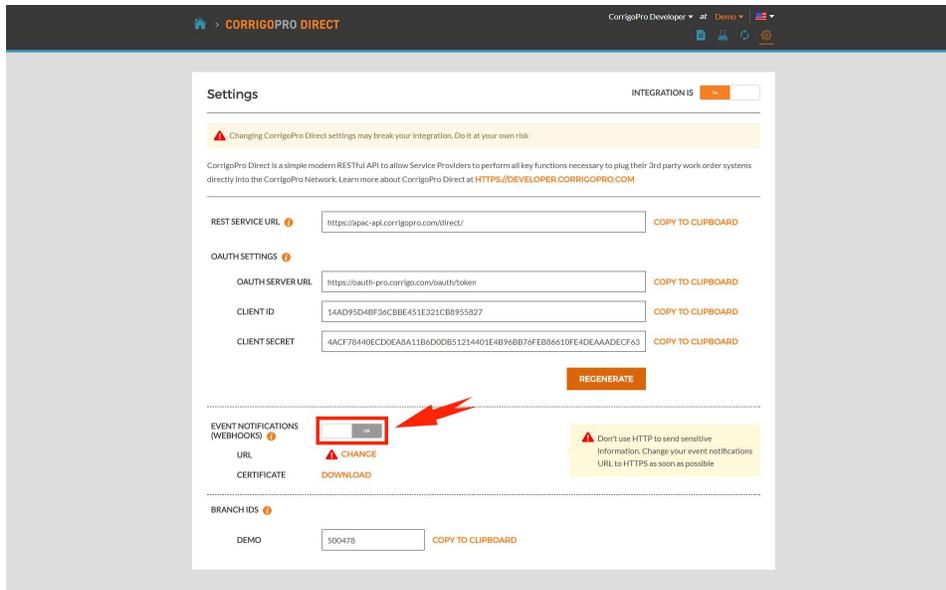


Figure 1: Event Notification configuration within CorrigoPro Direct Settings.

All event notifications will be directed to the URL that you provide within CorrigoPro Direct Settings and will be logged within CorrigoPro Direct Communication Logs.

## Request Structure

Webhook data is sent as JSON in the POST request body. Each request contains an OAuth token that can be used to validate it. See [Token Validation](#) below for more information.

### Message Header

A message header is available in all requests. It includes general information about the notification. The Action parameter shows what was changed. This parameter might be used to determine the data that is received and its structure.

For details on the structure of data, see:

<https://developer.corrigopro.com/v1.0/docs/data-models#section-messageheader>

The following example shows a sample notification header:

#### HTTP

```
Connection: Keep-Alive
Content-Length: 3296
Content-Type: application/json; charset=utf-8
Authorization: Bearer H4sIAAAEAG2MQQr<---->fyyoO7AkAOIJG3 (truncated for clarity)
Expect: 100-continue
Host: your.webhook.com
```

### Request Body

Each request body contains detailed information about event that happened. To identify an event type, review the Header property of the notification JSON included in the request body.

The following JSON represents an example of a notification body:

**JSON**

```
{
  "Header": {
    "WorkOrderId": 42,
    "BranchId": 492042,
    "Sender": {
      "Name": "Jon Snow",
      "Phone": "+12025550141",
      "Type": "CorrigoPro"
    },
    "Action": "{name of an action}",
    "ActionDate": "2017-10-04T10:12:08.1671613Z",
    "ApiVersion": "1.0",
    "MessageId": "c1d3d80f-a76c-452c-9261-682279a659cb"
  },
  {other fields of notification}
}
```

*Note: You can emulate any notification using WebHook Emulator*

## Token Validation

It is strongly recommended to validate each request that your WebHook can receive. That is why CorrigoPro Direct sends each request with the bearer token inside the header. You can validate this token by using the following algorithm:

- 1) Split the token string into data and sign using this separator "<---->".
- 2) Decode sign and data using Base64Url encoder.
- 3) Verify data by using sign and a public certificate.

To obtain the <client\_id>, <client\_secret>, <OAuth Server Url> and the public certificate, open the CorrigoPro Direct tile and turn the integration on.

## Proceed with Implementation

The preceding information in this guide should cover all the core concepts needed to develop your integration. The CorrigoPro Developer Hub and CorrigoPro Direct API Specifications should be able to guide you through most of the remaining details.

## Tools

Tools such as cURL ([curl.haxx.se](http://curl.haxx.se)), Postman ([getpostman.com](http://getpostman.com)), and Fiddler ([telerik.com/fiddler](http://telerik.com/fiddler)) can be useful when prototyping API calls and for testing. In addition, CorrigoPro Direct offers additional tools to assist you with testing, development, and diagnostics.

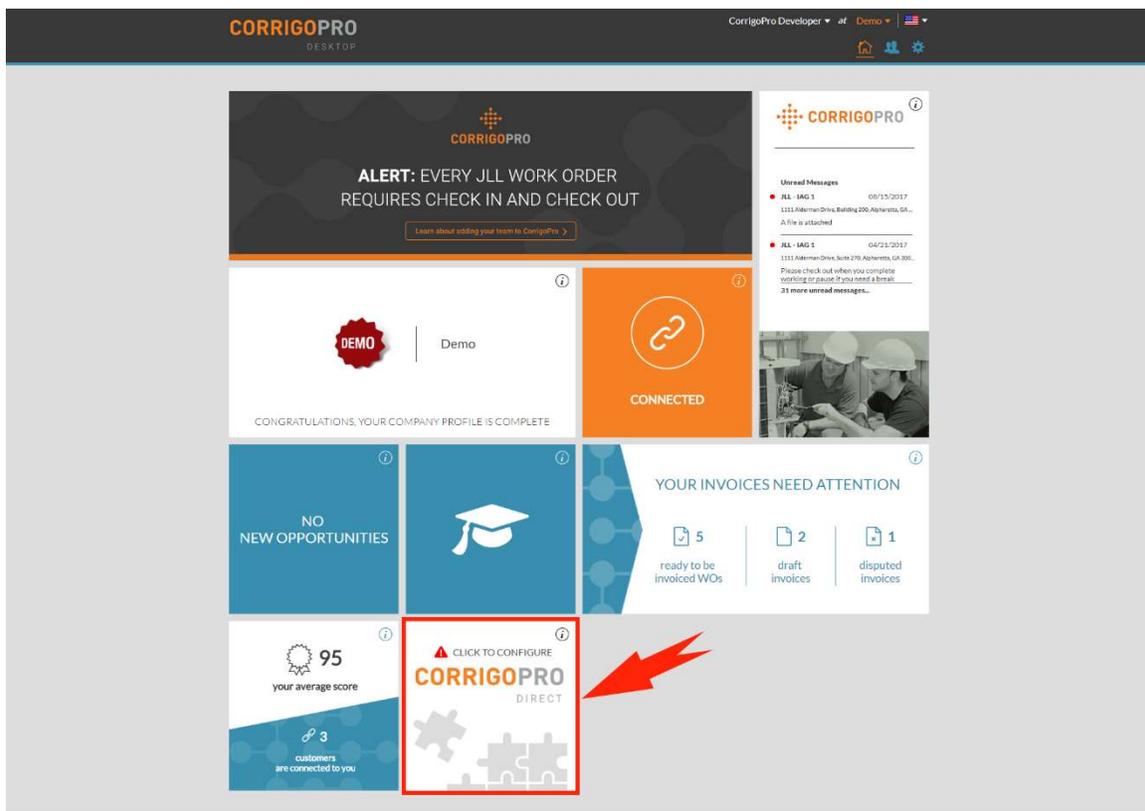
## Event Notification Emulator

CorrigoPro's Event Notification Emulator (also sometimes referred to as the Webhook Emulator) is available to provide emulated messages to assist you in developing and testing your endpoint. Each event notification type is supported by the emulator. See [Configuring your endpoint](#) for information on setup.

To start using the Webhook Emulator, perform the following steps:

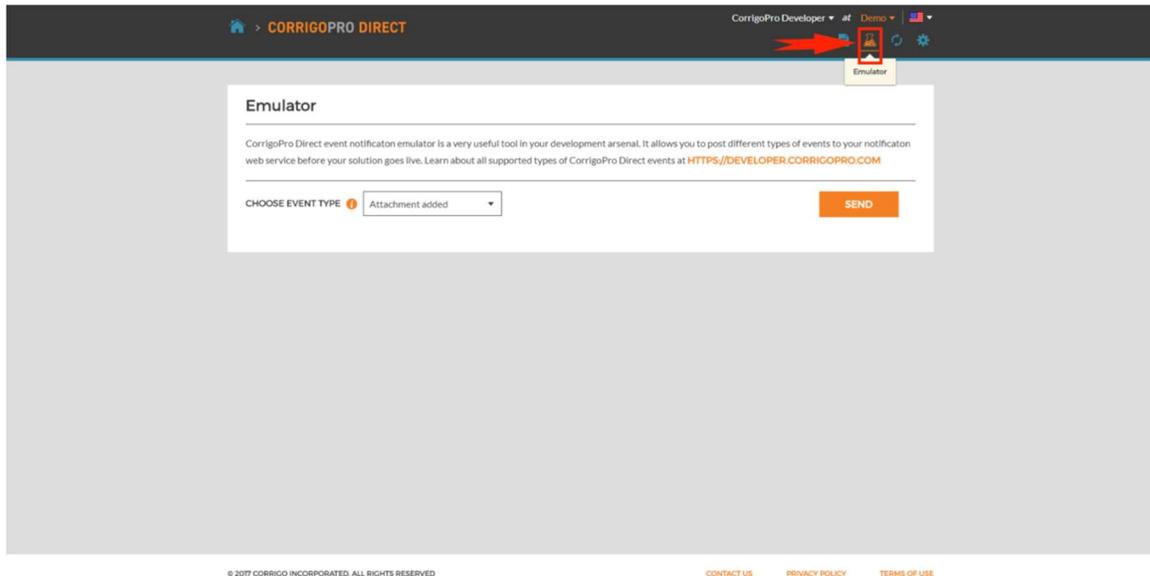
- 1) Log in to Corrigopro Desktop application.

The Corrigopro Direct tile is displayed on the home page:



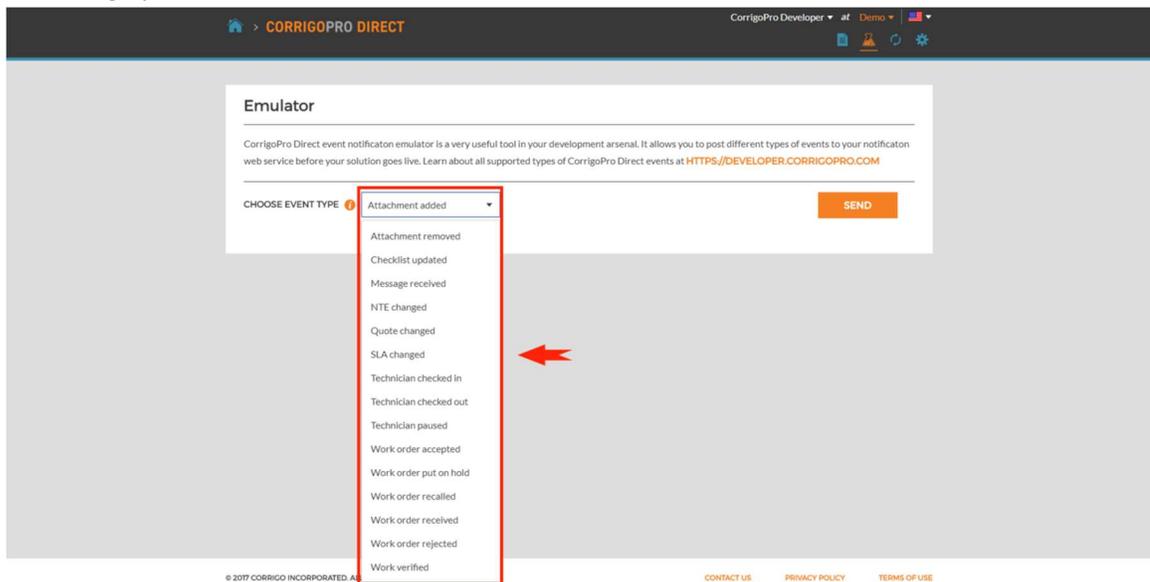
If this tile is not displayed, review the Prerequisites and Configuration.

2) Navigate to the Emulator tab.



3) Select a type of notification to be emulated.

WebHook Emulator provides means to emulate every notification type described in the Receiving updates section.



4) Select a branch (Optional).

CorrigoPro Desktop application supports multi-branch companies. Therefore, an ID of a recipient branch is always specified in any WebHook notification (see **Message Header**).

If your company has more than one branch, a branch selector is displayed in the WebHook Emulator layout. You must select a recipient branch prior to emulating notifications.

- 5) Click Send.

When a notification type (and a branch - if needed) is selected, you can press Send button to generate notification payload, and send it to your WebHook endpoint.

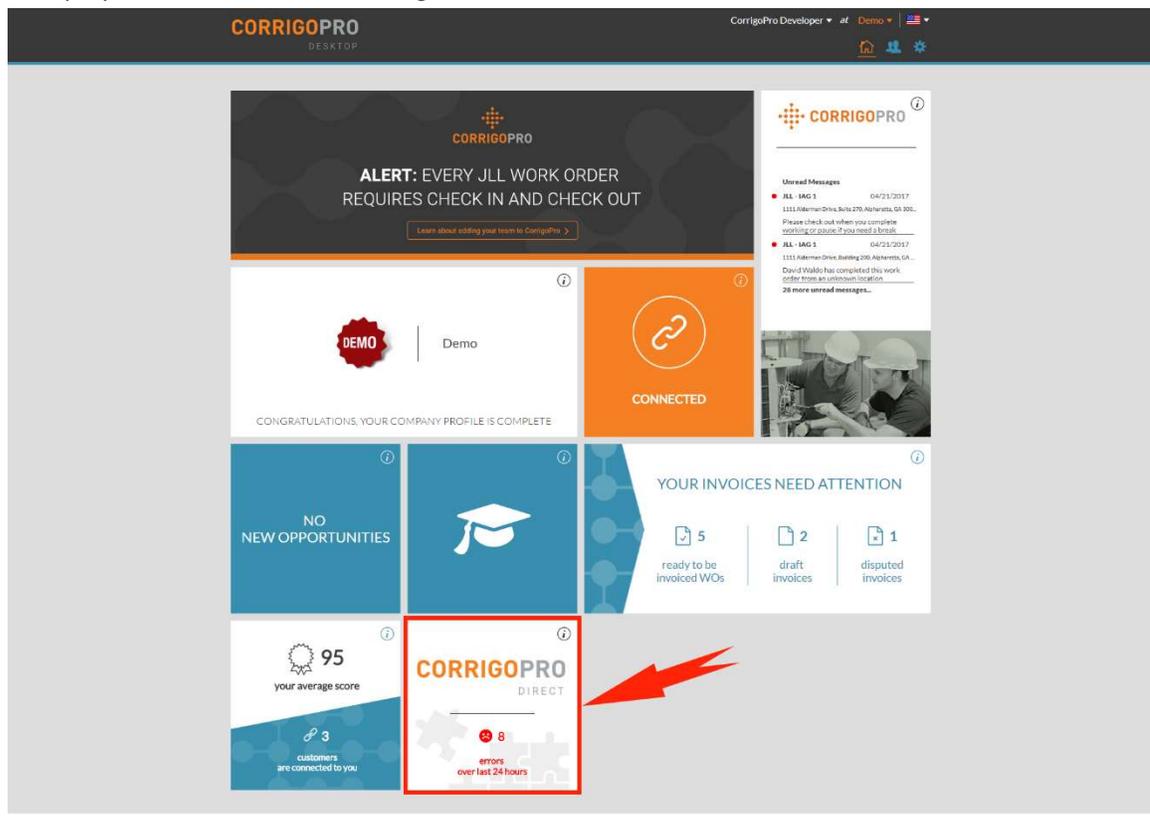
When a request is sent, real and valid HTTP headers are generated by CorrigoPro Direct (including access token), as well as response returned from your WebHook endpoint.

### Web Services Log

Communication logs are particularly useful for Support teams and very helpful in diagnosing and troubleshooting integration issues.

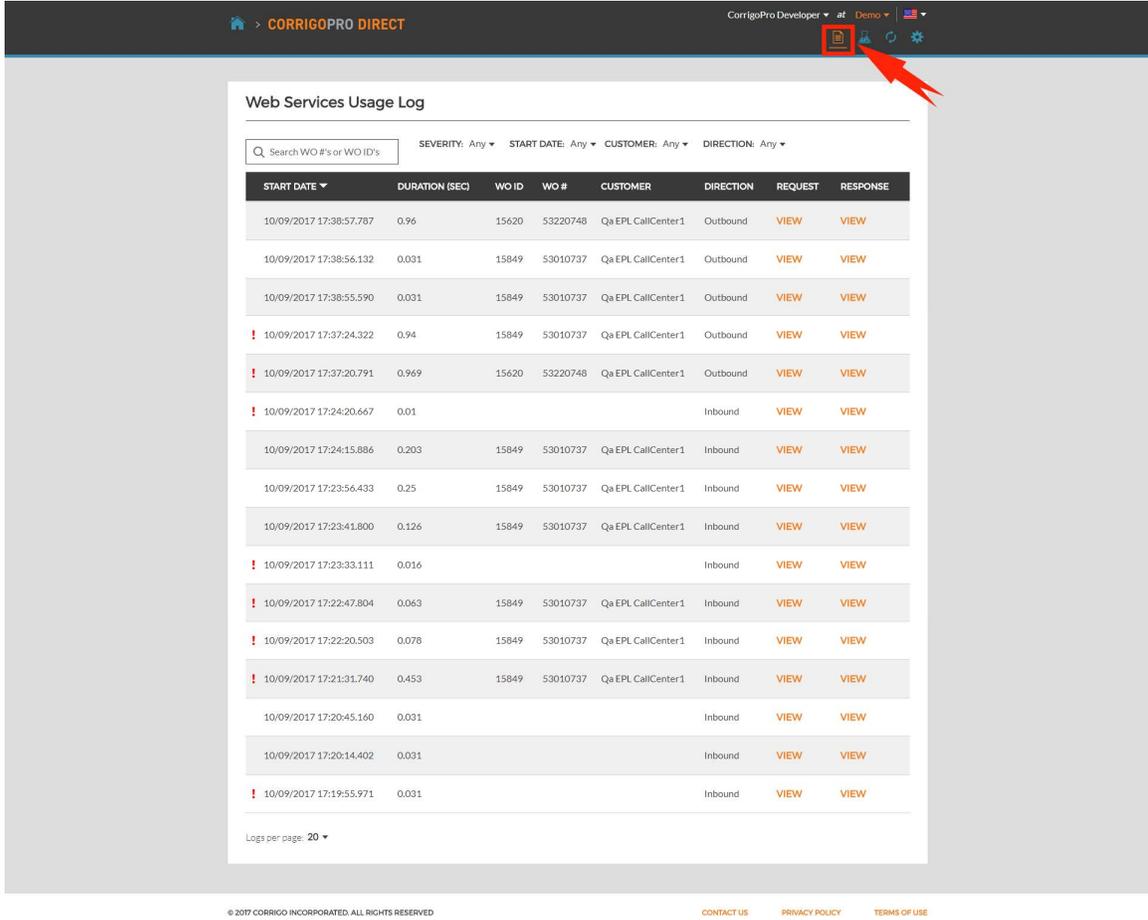
- 1) Login to CorrigoPro Desktop application.

The CorrigoPro Direct tile is displayed on the home page. After you start using CorrigoPro Direct, it displays health status of the integration.



If this tile is not displayed, review the Prerequisites and Configuration.

- 2) Navigate to the Usage Log tab.



The screenshot shows the CorrigoPro Direct interface. At the top, there is a navigation bar with the text "CORRIGOPRO DIRECT" and "CorrigoPro Developer". A red arrow points to a square icon in the top right corner of the navigation bar, which represents the Usage Log. Below the navigation bar, the "Web Services Usage Log" page is displayed. It features a search bar, filter options for SEVERITY, START DATE, CUSTOMER, and DIRECTION, and a table of log entries. The table has columns for START DATE, DURATION (SEC), WO ID, WO #, CUSTOMER, DIRECTION, REQUEST, and RESPONSE. The log entries show various requests and responses, some with error indicators (exclamation marks).

START DATE	DURATION (SEC)	WO ID	WO #	CUSTOMER	DIRECTION	REQUEST	RESPONSE
10/09/2017 17:38:57.787	0.96	15620	53220748	Qa EPL CallCenter1	Outbound	VIEW	VIEW
10/09/2017 17:38:56.132	0.031	15849	53010737	Qa EPL CallCenter1	Outbound	VIEW	VIEW
10/09/2017 17:38:55.590	0.031	15849	53010737	Qa EPL CallCenter1	Outbound	VIEW	VIEW
! 10/09/2017 17:37:24.322	0.94	15849	53010737	Qa EPL CallCenter1	Outbound	VIEW	VIEW
! 10/09/2017 17:37:20.791	0.969	15620	53220748	Qa EPL CallCenter1	Outbound	VIEW	VIEW
! 10/09/2017 17:24:20.667	0.01				Inbound	VIEW	VIEW
10/09/2017 17:24:15.886	0.203	15849	53010737	Qa EPL CallCenter1	Inbound	VIEW	VIEW
10/09/2017 17:23:56.433	0.25	15849	53010737	Qa EPL CallCenter1	Inbound	VIEW	VIEW
10/09/2017 17:23:41.800	0.126	15849	53010737	Qa EPL CallCenter1	Inbound	VIEW	VIEW
! 10/09/2017 17:23:33.111	0.016				Inbound	VIEW	VIEW
! 10/09/2017 17:22:47.804	0.063	15849	53010737	Qa EPL CallCenter1	Inbound	VIEW	VIEW
! 10/09/2017 17:22:20.503	0.078	15849	53010737	Qa EPL CallCenter1	Inbound	VIEW	VIEW
! 10/09/2017 17:21:31.740	0.453	15849	53010737	Qa EPL CallCenter1	Inbound	VIEW	VIEW
10/09/2017 17:20:45.160	0.031				Inbound	VIEW	VIEW
10/09/2017 17:20:14.402	0.031				Inbound	VIEW	VIEW
! 10/09/2017 17:19:55.971	0.031				Inbound	VIEW	VIEW

Logs per page: 20

© 2017 CORRIGO INCORPORATED. ALL RIGHTS RESERVED. CONTACT US PRIVACY POLICY TERMS OF USE

Every request to the API, and every event notification to the Pro's listener is logged. You can find this information on the Web Services Usage Log page of the CorrigoPro Direct tile.

For more detailed information regarding the Web Services Log, see:

<https://developer.corrigopro.com/docs/communication-logs>

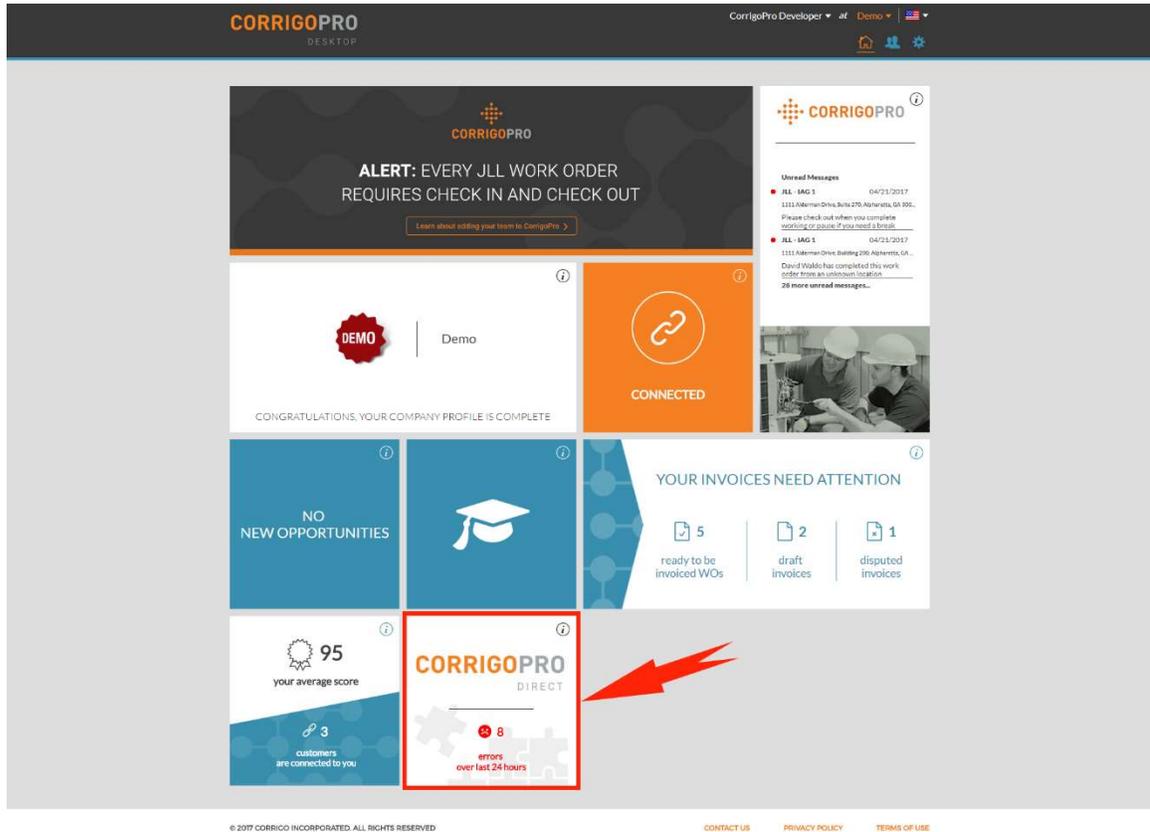
### Event Notification Retry Queue

If communication is disrupted between Corrigo and your system, requests will be added to the **Event Notification Retry Queue**. Corrigo will attempt to periodically resend queued requests after a short delay. This delay will gradually increase with time, up until it reaches 30 minutes for up to 8 days.

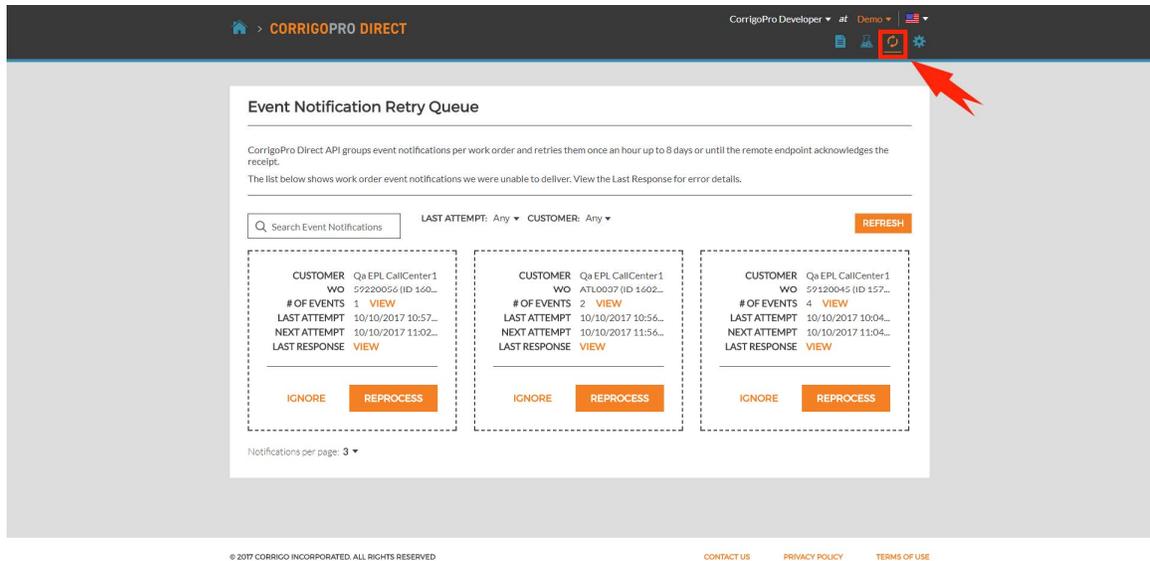
The **Event Notification Retry Queue** allows you to review information, reprocess, or ignore undelivered notifications.

To review all undelivered notifications, perform the following steps:

- 1) Select the **CorrigoPro Direct** tile from within *CorrigoPro Desktop* application.



2) Select the **Event Notification Retry Queue** tab.



## Test Work Orders

As part of the initial setup of your Sandbox, you will also receive a schedule of daily work orders that you can use for testing purposes.

If you need additional work orders, or work orders to meet specific requirements, please contact us with your request.

## Going Live

Once development and testing are complete, you may begin preparations for going live. If the Direct Tile is not already available in your production instance, please contact Direct Support.

The process for configuring your production instance is the same as configuring your Sandbox instance. You will need to regenerate a Client ID and Secret ID, which will be different than that of your Sandbox. You will also need to configure your Event Notification settings if you are using Webhooks.

## Going Live for WorkTrack Users

For the small number of providers that are current users of WorkTrack, there are additional steps.

CorrigoPro Direct is a module of CorrigoPro. To begin using CorrigoPro Direct, you must upgrade and migrate from WorkTrack to CorrigoPro.

Once upgraded to CorrigoPro, all new work orders will flow to CorrigoPro. They will not be available in WorkTrack. You will still need to access WorkTrack to complete and invoice existing work orders, which will be accessible through a link on the CorrigoPro Desktop.

Below are the steps for going live with CorrigoPro and Corrigo Pro Direct:

- Schedule a CorrigoPro training session for any users that have not yet received training or need a refresher.
- **Ensure that everyone is aware that existing work orders will continue to flow through WorkTrack, but all new work orders will only be available through CorrigoPro.**
- Decide upon a day that is easiest to perform the migration. Ideally, this should be a day where work order flow will be at a minimum. A minimum of 1-2 days notice will be required.
- The migration will occur at Midnight, Pacific time, on the scheduled date.
- The new integration needs to be reconfigured to work with your live instance instead of the sandbox.
  - The **Integration** must be turned 'On' from CorrigoPro Direct Settings of your live instance.
  - You must regenerate a new live Client Id and Client Secret and provide them to your integration.
  - If you are using Event Notification, you must enable Event Notifications and provide the URL of your live endpoint.
  - Obtain your Branch Ids and provide them to your integration (if applicable).

## Appendix A

### FAQ

#### What do I use for a MessageId?

This could be a GUID, a timestamp, or other unique identifier that you choose. Additionally, you may choose to prepend or append one of those identifiers with other data, as desired.

#### What if there is an outage and I'm temporarily unable to receive requests?

Messages will be placed in a retry queue. See [Event Notification Retry Queue](#) for more information.

#### How do I view the history of incoming and outgoing messages?

Web services messages are logged and accessible within CorrigoPro Direct. See [Web Services Usage Log](#) for more information.

#### Which fields are optional, and which are required?

The best resource to determining which fields are optional and required are the data model examples within the CorrigoPro Direct API Specification and the Data Models section of the CorrigoPro Developer Hub. Both sites document which fields are optional.

#### Getting Customer Location Data

The API does not grant access to location data for all customer sites.

Location data belongs to each customer, and can be shared with their permission, at their discretion.

However, even if your customer consents to sharing location data, there are still important items to consider:

- 1) Each customer may change site data at any time and there is no mechanism in place to notify you of such a change.
- 2) You must be prepared to handle any instances where incoming work order requests have new locations that are not yet in your system, or locations that have been updated since you obtained the location data.

#### I have another question...

The best resource for information regarding workflow, API use, and the tools available is the CorrigoPro Developer Hub:

<https://developer.corrigopro.com/>

Questions regarding specific API methods may be answered on our API Documentation site:

<http://api.corrigopro.com/documentation/>

You may also submit a question at the Support section of the Developer Hub:]

<https://developer.corrigopro.com/discuss>

## Appendix B

### Common Errors

Below is a list of common errors frequently encountered by developers beginning to use the CorrigoPro Direct API, along with common resolutions.

For a complete list of API Error codes, see: <https://developer.corrigopro.com/docs/error-codes>

#### Status: 400 (Bad Request)

##### Example JSON response:

```
{
  "Code": 2100,
  "Message": "Model data validation failed",
  "ValidationErrors": [
    {
      "FieldName": "",
      "ErrorType": "IncorrectType",
      "Message": "Empty request model"
    }
  ],
  "MessageId": null
}
```

##### Common cause:

- A required parameter is not included in the request or is not the correct type of data. Usually, there will be a message indicating which parameter is missing or incorrect.

Other Status 400 types of responses can include:

##### Example JSON response:

```
{
  "error": "invalid_client"
}
```

##### Common cause:

- Missing or incorrect **client\_id** or **client\_secret** when sending a request to obtain a Bearer Token.

**Example JSON response:**

```
{
  "error": "unsupported_grant_type"
}
```

**Common cause:**

- Missing or incorrect **grant\_type** value (should be "*client\_credentials*") when sending a request to obtain a Bearer Token.

**Example JSON response:**

```
{
  "Code": 2100,
  "Message": "Model data validation failed",
  "ValidationErrors": [
    {
      "FieldName": "",
      "ErrorType": "IncorrectType",
      "Message": "Empty request model"
    }
  ],
  "MessageId": null
}
```

**Common cause:**

- Performing a GET method without sending the required parameters. Ensure that the required parameters are included in a query string (not as a JSON body).

**Status: 401 (Unauthorized)****Example JSON response:**

```
{
  "Code": 2000,
  "Message": "Authorization has been denied for this request.",
  "ValidationErrors": null,
  "MessageId": null
}
```

**Common causes:**

- Authorization header with Bearer token is missing or incorrect.
- Bearer token has expired. Bearer tokens expire after 1199 seconds (about 20 minutes).

## Status: 500 (Internal Server Error)

### Example JSON response:

```
{
  "Code": 1000,
  "Message": "Internal server error occurred. This might be a configuration, integration, or setup
Internal server error.",
  "ValidationErrors": null,
  "MessageId": "f54d6dcb-d166-4794-bc91-c77fde173c31"
}
```

### Common causes:

- Not including the correct **Content-Type** header value with a value of “application/json” in your header.
- Sending a request to the incorrect region. Verify that you are using the correct API.