ISV - Delivery API's Information (Hypur Pay)

Saturday, February 22, 2020
1:43 PM

DELIVERY / ORDERS API

The Delivery / Orders API enables merchants to process orders received through Integrated software vendors (ISV) and their delivery website, eCommerce websites, or mobile apps.

Through this API, participating merchants can:

- Receive and acknowledge new orders using order API
- Reflect their progress of the delivery using update API
- Complete order for delivery using update API.**
- Complete order for instore using update API.**

**The step is not required if you use the Hypur's Mobile responsive dashboard.

Provide refunds in response to returns or other events
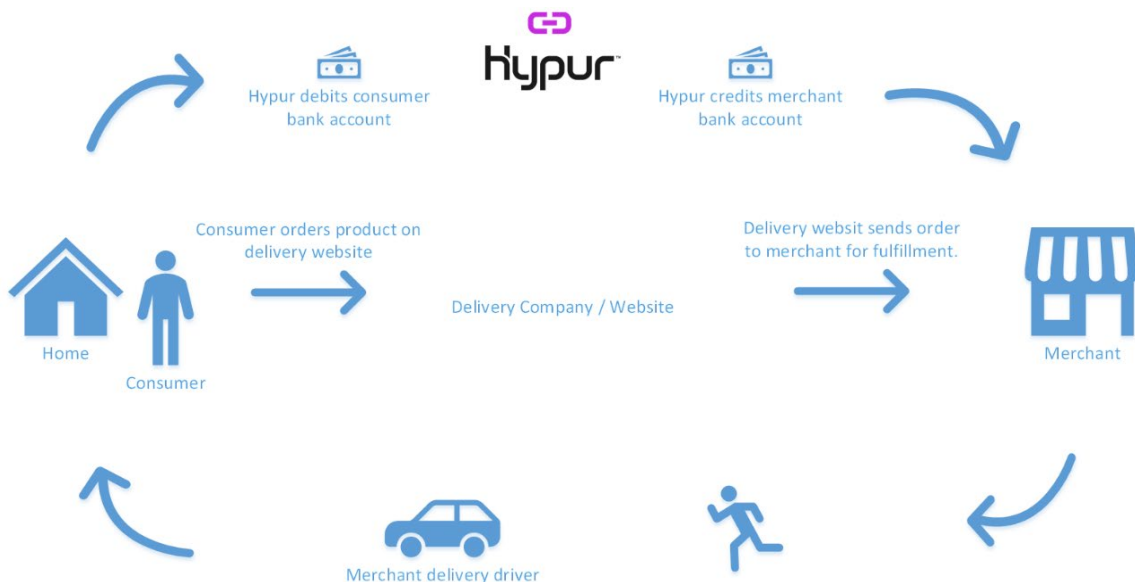
Below are the links to the Test APIs you will be using for Orders and Delivery completions.
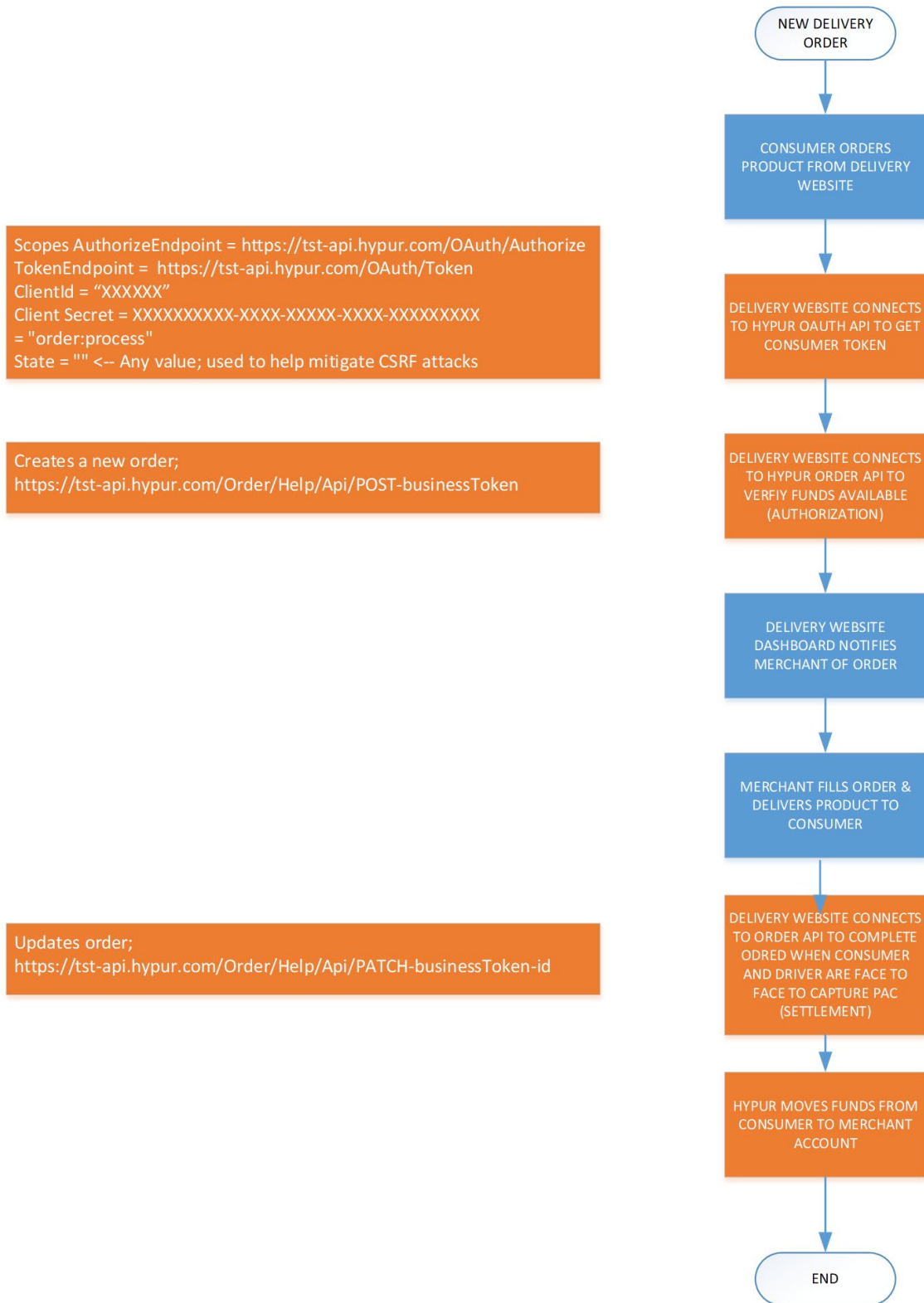
HELP:
https://tst-api.hypur.com/Order/Help

CREATES A NEW ORDER:
https://tst-api.hypur.com/Order/Help/Api/POST-businessToken

UPDATES ORDER:
https://tst-api.hypur.com/Order/Help/Api/PATCH-businessToken-id

Video of User Experience
https://www.screencast.com/t/CncamD23fFQ

HYPUR DELIVERY / INSTORE FLOW CHART:

NEW DELIVERY ORDER

↓

CONSUMER ORDERS PRODUCT FROM DELIVERY WEBSITE

↓

Scopes AuthorizeEndpoint = https://tst-api.hypur.com/OAuth/Authorize
TokenEndpoint = https://tst-api.hypur.com/OAuth/Token
ClientId = "XXXXXX"
Client Secret = XXXXXXXXXX-XXXX-XXXXX-XXXX-XXXXXXXXX
= "order:process"
State = "" <-- Any value; used to help mitigate CSRF attacks

DELIVERY WEBSITE CONNECTS TO HYPUR OAUTH API TO GET CONSUMER TOKEN

↓

Creates a new order;
https://tst-api.hypur.com/Order/Help/Api/POST-businessToken

DELIVERY WEBSITE CONNECTS TO HYPUR ORDER API TO VERFIY FUNDS AVAILABLE (AUTHORIZATION)

↓

DELIVERY WEBSITE DASHBOARD NOTIFIES MERCHANT OF ORDER

↓

MERCHANT FILLS ORDER & DELIVERS PRODUCT TO CONSUMER

↓

Updates order;
https://tst-api.hypur.com/Order/Help/Api/PATCH-businessToken-id

DELIVERY WEBSITE CONNECTS TO ORDER API TO COMPLETE ODRED WHEN CONSUMER AND DRIVER ARE FACE TO FACE TO CAPTURE PAC (SETTLEMENT)

↓

HYPUR MOVES FUNDS FROM CONSUMER TO MERCHANT ACCOUNT

↓

END

OAUTH2 API

To get the necessary token to post an order, you'll need to utilize our OAuth2 API. At checkout, the consumer can grant the merchant/business permission to process a payment on their behalf.

AuthorizeEndpoint = https://tst-api.hypur.com/OAuth/Authorize
TokenEndpoint = https://tst-api.hypur.com/OAuth/Token
ClientId = "XXXXXX"
Client Secret = XXXXXXXXX-XXXX-XXXXX-XXXX-XXXXXXXXX
Scopes = "order:process"
State = "" <-- Any value; used to help mitigate CSRF attacks

```
var url = $"{AuthorizeEndpoint}?client_id={WebUtility.UrlEncode(ClientId)}&response_type=code&scope={WebUtility.UrlEncode(Scopes)}&redirect_uri={WebUtility.UrlEncode(RedirectUri)}&state={WebUtility.UrlEncode(State)}";
```

Instructions:
The Redirect URL provided above will need to handle converting the authorization code into an access token by following the steps below.

1. Validate the State value returned in the query string equals the one provided in the initial call.

2. Set the Authorization header

    a. The authorization scheme: Basic

    b. The value is the Base64 encoded client credentials

        I. Format before Base64 encoding: Clientid: ClientSecret

3. Set the following headers

    a. "grant_type" = "authorization_code"

    b. "code" = {code}

        I. The code value from the querystring

    c. "redirect_uri" = {RedirectUri}

4. Post this request to the Token Endpoint provided

5. The response, if successful, will be JSON and contain the following data.

    a. "access_token"

    b. "token_type"

    c. "expires_in"

    d. "refresh_token"

6. The access token and refresh token will need to be stored along with your user identifier, to prevent the user from having to sign in to their Hypur profile again.

a. The access token is valid for 30 minutes or 1 hour. The refresh token is valid for 30 days, and can be used to get a new access token without any user interaction.

To get a new access token using the refresh token.

1. Set the Authorization header

   a. The authorization scheme: Basic

   b. The value is the Base64 encoded client credentials

      I. Format before Base64 encoding: Clientid:ClientSecret

2. Set the following headers

   a. "grant_type" = "refresh_token"

   b. "scope" = {scope}

      I. The scope from above

   c. "refresh_token" = { refresh_token }

3. Post this request to the Token Endpoint provided

4. The response, if successful, will be JSON and contain the following data.

   a. "access_token"

   b. "token_type"

   c. "expires_in"

   d. "refresh_token"

5. The access token and refresh token will need to be stored along with your user identifier,

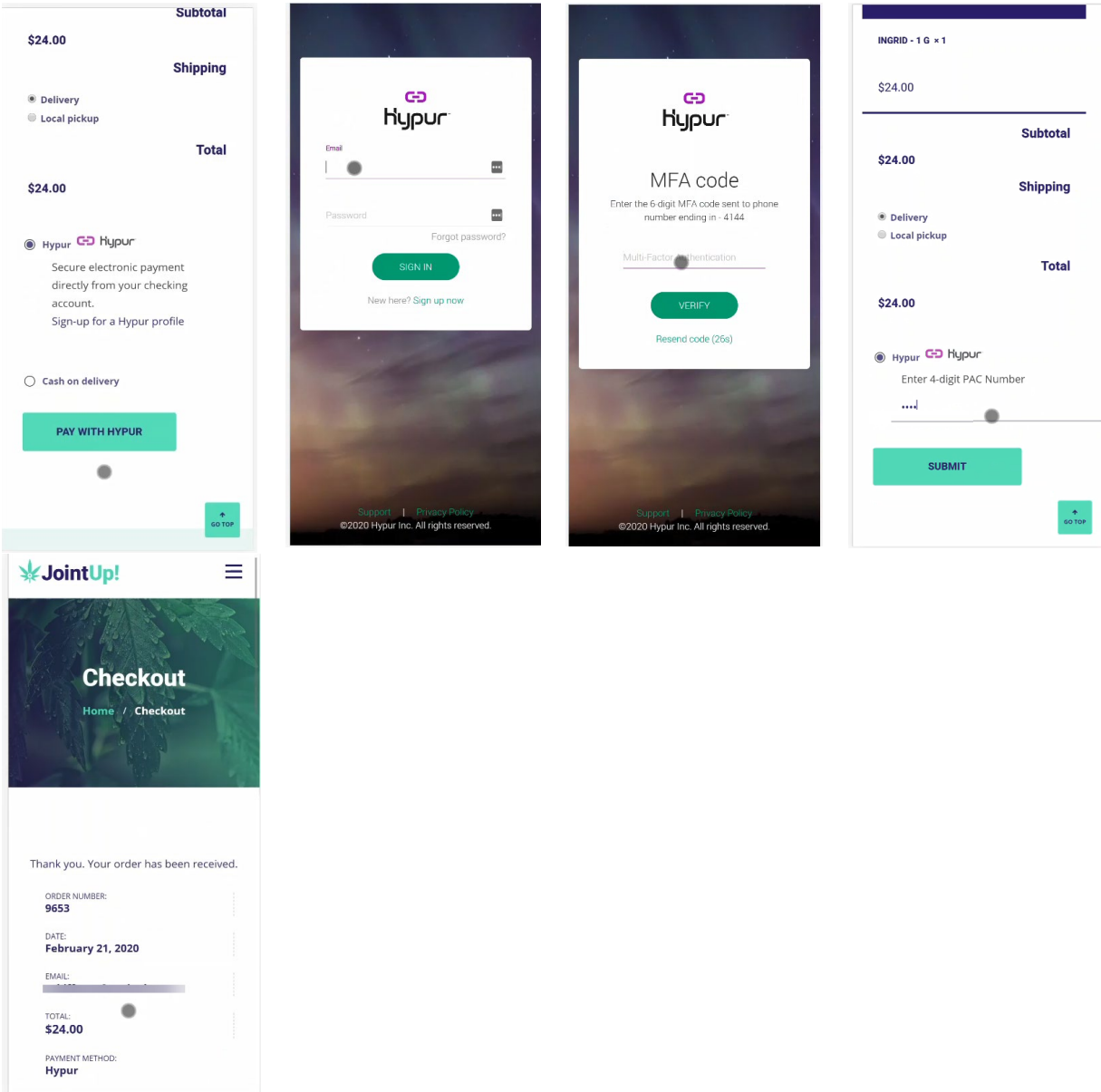   to prevent the user from having to sign in to their Hypur profile again.

Once the access token has been retrieved, a call can be made against the protected API by sending the access token in the Authorization header.

- The authorization scheme: Bearer
- The value is the user access token

OAUTH REDIRECT FOR LOGIN / ACCESS TOKEN FOR ODER API:

From the ISV's Checkout process, you will call the Order API for Hypur Pay.

Once the access token has been retrieved, a call can be made against the protected API by sending the access token in the Authorization header.

SAMPLE (MOBILE RESPONSIVE) FOR HYPUR PAY ON PORCH DELIVERY:

Once the order posts, it will appear on the Hypur Orders dashboard for instore or delivery fulfillment.
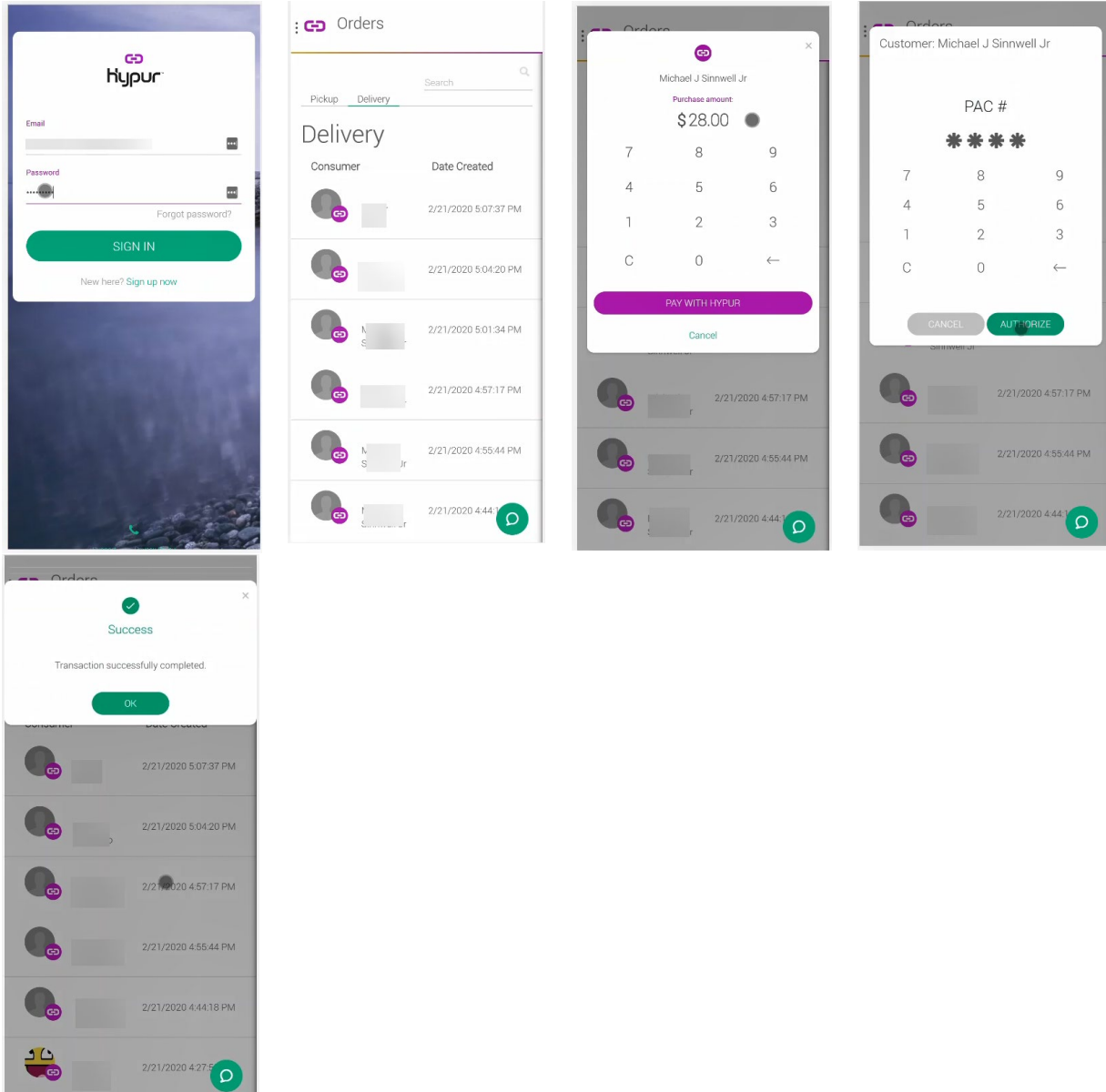
For your information, the dashboard is mobile responsive and has delivery/pickup employee Permissions.