QUX Pay® API DOCUMENTATION

Introduction

Welcome to the QUX Pay® API documentation. This API is designed to provide developers with access to QUX Pay®'s platform, allowing them to integrate QUX® features into their applications. Whether you're building a website, mobile app, or any other digital platform, the QUX Pay® API can help enhance your user experience.

Overview

The QUX Pay ® API offers a range of features, including account management, content discovery, and user engagement tools. With this API, developers can access user data, manage accounts, and provide personalized content recommendations based on user preferences.

Base URL
- For API: https://api.qux.tv
- Website URL: https://quxpay.com

Authentication

To authenticate requests to the QUX Pay® API, you need an API key from QUX Pay® . Here's how to get started:

1. **Register as a Merchant**: Sign up at https://qux.tv
2. **Add a Store**: After registration, add a store to your account.
3. **Get API Key**: Visit https://quxpay.com/biller-api
4. **Generate an API key for your store.**

Endpoints

1. **Checkout Details API**
   Description: Retrieves details of a purchase order.
   Endpoint URL: /web/wp/po-details

2. **Payment API**

Description: Marks a purchase order as paid.
Endpoint URL: /web/wp/po-paid

HTTP Method: GET, POST, PUT, DELETE, etc.

APIs

Endpoint Title: Order Details

Description: Retrieves the details of a specific order from the QUX Pay® website.
HTTP Method: GET
Endpoint URL: /web/wp/po-details

Request:
Parameters:

- wp (longText): The encoded in base64 data from QUX Pay® Plugin
  - x_amount (number): Total amount in checkout process
  - x_invoice_num (number): Order ID from woocommerce checkout
  - x_fp_timestamp (timestamp): Date and time of the transaction was made
  - x_first_name (string): Biller first name from user input in checkout
  - x_last_name (string): Biller last name from user input in checkout
  - x_company (string): Biller company from user input in checkout
  - x_address (string): Biller address from user input in checkout
  - x_country (string): Biller country from user input in checkout
  - x_state (string): Biller state from user input in checkout
  - x_city (string): Biller city from user input in checkout
  - x_zip (string): Biller zip from user input in checkout
  - x_phone (string): Biller phone number from user input in checkout
  - x_email (string): Biller email from user input in checkout
  - x_ship_to_first_name (string): Shipping first name from user input in checkout
  - x_ship_to_last_name (string): Shipping last name from user input in checkout
  - x_ship_to_company (string): Shipping company from user input in checkout
  - x_ship_to_address (string): Shipping address from user input in checkout
  - x_ship_to_country (string): Shipping country from user input in checkout
  - x_ship_to_state (string): Shipping state from user input in checkout
  - x_ship_to_city (string): Shipping city from user input in checkout
  - x_ship_to_zip (string): Shipping zip from user input in checkout
  - x_cancel_url (string): The cancel url redirect when user cancel the payment while they are in QUX Pay® website

- o secret_key (string): The secret key generated by the merchant of the website in QUX®
- o app_key (string): The api key generated by the merchant of the website in QUX®
- o products (array): The list of single purchase that made of the user in merchant website
  - ▪ id (number): Product ID in merchant website
  - ▪ quantity (number): Total quantity of specific product to be checkout by the user
  - ▪ product_name (string): Product name in merchant website
  - ▪ product_price (number): Product price in merchant website
  - ▪ subscription (boolean): Checker for specific product if subscription or not; default is false
- o sandbox (boolean): The option in merchant website if they want to test the QUX Pay® in their website
- o plugin (string): The plugin version for us to identify the plugin used in merchant website

Headers:
- • Authorization: Bearer Token
- • Content-Type: application/json

Request:

```
{
"wp":"eyJ4X2Ftb3VudCI6IjEyLjAwIiwieF9pbnZvaWNlX251bSI6ODEsInhfcmVsYXlfcm
VzcG9uc2UiOiJUUlVFIiwieF9mcF9zZXF1ZW5jZSI6ODEsInhfZnBfdGltZXN0YW1wIjo
xNzEzNzg1NzMzLCJ4X2ZpcnN0X25hbWUiOiJKb2huIiwieF9sYXN0X25hbWUiOiJEb
2UiLCJ4X2NvbXBhbnkiOiIiLCJ4X2FkZHJlc3MiOiIxMjM0NTY3OCAiLCJ4X2NvdW50c
nkiOiJQSCIsInhfc3RhdGUiOiIwMCIsInhfY2l0eSI6IkNhbG9vY2FuIiwieF96aXAiOiIxMT
IxIiwieF9waG9uZSI6IjEyMzQ1Njc4OSIsInhfZW1haWwiOiJnbGVubmhcmxvMTIzQG
dtYWlsLmNvbSIsInhfc2hpcF90b19maXJzdF9uYW1lIjoiSm9obiIsInhfc2hpcF90b19sY
XN0X25hbWUiOiJEb2UiLCJ4X3NoaXBfdG9fY29tcGFueSI6IiIsInhfc2hpcF90b19hZG
RyZXNzIjoiMTIzNDU2NzggIiwieF9zaGlwX3RvX2NvdW50cnkiOiJQSCIsInhfc2hpcF90
b19zdGF0ZSI6IjAwIiwieF9zaGlwX3RvX2NpdHkiOiJDYWxvb2NhbiIsInhfc2hpcF90b19
6aXAiOiIxMTIxIiwieF9jYW5jZWxfdXJsIjoiaHR0cDpcL1wvbG9jYWxob3N0OjgwMDBc
L2luZGV4LnBocFwvY2hlY2tvdXRcLyIsInhfZnJlaWdodCI6IiLCJ4X2NhbmNlbF91cm
xfdGV4dCI6IkNhbmNlbCBQYXltZW50Iiwic2VjcmV0X2tleSI6IkVmY3hGemZHeXRlbIA
0ZW9SakE3V0Rtc2kwTjVhRREtRMltUnJIb3dPVXQzTFp4WDloIiwiYXBwX2tleSI6Iklz
Q2FhMUdjUnFFRb2w5T3NBSjFnIiwiY2FydCI6eyJhYWIzMjM4OTIyYmNjMjVhNmY2M
DZlYjUyNWZmZGM1NiI6eyJrZXki0iJhYWIzMjM4OTIyYmNjMjVhNmY2MDZlYjUyNW
ZmZGM1NiIsInByb2R1Y3RfaWQiOjE0LCJ2YXJpYXRpb25faWQiOjAsInZhcmlhdGlvb
iI6W10sInF1YW50aXR5IjoxLCJkYXRhX2hhc2giOiJiNWMxZDVjYThiYWU2ZDQ4OTZ
```

jZjE4MDdjZGY3NjNmMCIsImxpbmVfdGF4X2RhdGEiOnsic3VidG90YWwiOltdLCJ0b3RhbCI6W119LCJsaW5lX3N1YnRvdGFsIjoxMiwibGluZV9zdWJ0b3RhbF90YXgiOjAsImxpbmVfdG90YWwiOjEyLCJsaW5lX3RheCI6MCwiZGF0YSI6e319fSwicHJvZHVjdHMiOlt7ImlkIjoxNCwicXVhbnRpdHkiOjEsInByb2R1Y3RfbmFtZSI6InRlc3QiLCJwcm9kdWN0X3ByaWNlIjoiMTIiLCJzdWJzY3JpcHRpb24iOmZhbHNlfV0sInByb2R1Y3Rfc3Vic2NyaXB0aW9uIjpbXX0"
}

Response:
Status Code:
- 200: Success
- 400: Error/Merchant API or Secret key is invalid

Response Body:

```json
{
  "data": {
    "sending_to": "TEST STORE",
    "single_purchase_amount": 0,
    "recurring_payment_amount": 12,
    "token_fee": 0,
    "total_token_amount": "12.00",
    "is_single_purchase": false,
    "recurring_payment": true,
    "single_and_recurring_payment": false,
    "details": {
      "x_amount": "12.00",
      "x_invoice_num": 289,
      "x_relay_response": "TRUE",
      "x_fp_sequence": 289,
      "x_fp_timestamp": 1718273397,
      "x_first_name": "four",
      "x_last_name": "Baquero",
      "x_company": "",
      "x_address": "123 Street ",
      "x_country": "PH",
      "x_state": "00",
      "x_city": "Caloocan",
      "x_zip": "1123",
      "x_phone": "1234567890",
      "x_email": "glenncarlo123@gmail.com",
      "x_ship_to_first_name": "four",
```

```json
        "x_ship_to_last_name": "Baquero",
        "x_ship_to_company": "",
        "x_ship_to_address": "123 Street ",
        "x_ship_to_country": "US",
        "x_ship_to_state": "OHIO",
        "x_ship_to_city": "Test",
        "x_ship_to_zip": "1123",
        "x_cancel_url": "http://localhost:8000/index.php/checkout/",
        "x_freight": "0",
        "x_cancel_url_text": "Cancel Payment",
        "secret_key": "Bd1voyy37ju8HWiGcid4CSgoiBEJtkUIJBOO0EDM0KbIIYinh6",
        "app_key": "eos8WrN4etzZ9RDaIvGI",
        "products": [],
        "product_subscription": [
            {
                "id": 91,
                "quantity": 1,
                "product_name": "TEST SUBSCRIPTION",
                "product_price": "12",
                "subscription": true
            }
        ],
        "sandbox": "yes",
        "plugin": "wordpress_plugin v1.0.2"
    }
},
"status": {
    "responseCode": 200,
    "success": true,
    "message": "PO Details"
}
}
```

Endpoint Title: Order Payment
Description: Processes a payment for an order on the QuxPay website.
HTTP Method: POST
Endpoint URL: /web/wp/po-paid

Request:
Parameters:

- wp (longText): The encoded in base64 data from QUX Pay® Plugin
    - x_amount (number): Total amount in checkout process
    - x_invoice_num (number): Order ID from woocommerce checkout
    - x_fp_timestamp (timestamp): Date and time of the transaction was made
    - x_first_name (string): Biller first name from user input in checkout
    - x_last_name (string): Biller last name from user input in checkout
    - x_company (string): Biller company from user input in checkout
    - x_address (string): Biller address from user input in checkout
    - x_country (string): Biller country from user input in checkout
    - x_state (string): Biller state from user input in checkout
    - x_city (string): Biller city from user input in checkout
    - x_zip (string): Biller zip from user input in checkout
    - x_phone (string): Biller phone number from user input in checkout
    - x_email (string): Biller email from user input in checkout
    - x_ship_to_first_name (string): Shipping first name from user input in checkout
    - x_ship_to_last_name (string): Shipping last name from user input in checkout
    - x_ship_to_company (string): Shipping company from user input in checkout
    - x_ship_to_address (string): Shipping address from user input in checkout
    - x_ship_to_country (string): Shipping country from user input in checkout
    - x_ship_to_state (string): Shipping state from user input in checkout
    - x_ship_to_city (string): Shipping city from user input in checkout
    - x_ship_to_zip (string): Shipping zip from user input in checkout
    - x_cancel_url (string): The cancel url redirect when user cancel the payment while they are in QuxPay website
    - secret_key (string): The secret key generated by the merchant of the website in Qux
    - app_key (string): The api key generated by the merchant of the website in Qux
    - products (array): The list of single purchase that made of the user in merchant website
        - id (number): Product ID in merchant website
        - quantity (number): Total quantity of specific product to be checkout by the user
        - product_name (string): Product name in merchant website
        - product_price (number): Product price in merchant website
        - subscription (boolean): Checker for specific product if subscription or not; default is false

- o sandbox (boolean): The option in merchant website if they want to test the QuxPay in their website
- o plugin (string): The plugin version for us to identify the plugin used in merchant website

Headers:
- Authorization: Bearer Token
- Content-Type: application/json

Request:

```
{
"wp":"eyJtb2RlbCI6IkFkdmVydGlzZW1lbnQiLCJ4X2Ftb3VudCI6MTksInhfaW52b2ljZV
9udW0iOjEsInhfcmVsYXlfcmVzcG9uc2UiOiJUUIVFIiwieF9mcF9zZXF1ZW5jZSI6MS
wieF9mcF90aW1lc3RhbXAiOiIyMDI0LTA2LTEyVDE5OjExOjE1LjE1MTYyNloiLCJ4X
2NhbmNlbF91cmwiOiJodHRwczpcL1wvcXV4dGVjaC50dlwvY2hlY2tvdXQiLCJ4X2Zy
ZWlnaHQiOjAsInhfY2FuY2VsX3VybF90ZXh0IjoiQ2FuY2VsIFBheW1lbnQiLCJzZWNy
ZXRfa2V5IjoiaW50ZXJuYWwiLCJhcHBfa2V5IjoiaW50ZXJuYWwiLCJwcm9kdWN0cyI
6W3siaWQiOjEsInF1YW50aXR5IjoxLCJwcm9kdWN0X25hbWUiOiJBZHMgMSIsInBy
b2R1Y3RfcHJpY2UiOjE5LCJzdWJzY3JpcHRpb24iOmZhbHNlfV0sInByb2R1Y3Rfc3V
ic2NyaXB0aW9uIjpbXSwic2FuZGJveCI6dHJ1ZSwicGx1Z2luIjoiaW50ZXJuYWwifQ==
"
}
```

Response
Status Code:
- 200: Success
- 400: Error/Merchant API or Secret key is invalid

Response Body:

```
{
  "data": {
    "total_payment": "12.00",
    "sent_to": "Ray Store",
    "redirect": "http://localhost/wordpress 2/index.php//order-successful/?success=eyJyZWZlcmVuY2VfbnVtIjoxNzAsInN1Y2Nlc3MiOnRydWUsInN
0YXR1cyI6ImNvbXBsZXRlZCIsImludm9pY2VfbnVtIjoyODl9",
    "success": true
  },
  "status": {
    "responseCode": 200,
    "success": true,
```

```
    "message": "Checkout success"
  }
}
```

SDKs and Libraries

https://github.com/QUX-Official

Code Samples

```php
$args = array(
        'x_amount'              => $order_total,
        'x_invoice_num'          => $order_id,
        'x_relay_response'        => "TRUE",
        'x_fp_sequence'           => $order_id,
        'x_fp_timestamp'          => $timeStamp,
        'x_first_name'           => $order->get_billing_first_name() ,
        'x_last_name'            => $order->get_billing_last_name() ,
        'x_company'              => $order->get_billing_company() ,
        'x_address'              => $order->get_billing_address_1() .' '. $order->get_billing_address_2(),
        'x_country'              => $order->get_billing_country(),
        'x_state'                => $order->get_billing_state(),
        'x_city'                 => $order->get_billing_city(),
        'x_zip'                  => $order->get_billing_postcode(),
        'x_phone'                => $order->get_billing_phone(),
        'x_email'                => $order->get_billing_email(),
        'x_ship_to_first_name'    => $order->get_shipping_first_name() ,
        'x_ship_to_last_name'     => $order->get_shipping_last_name() ,
        'x_ship_to_company'       => $order->get_shipping_company() ,
        'x_ship_to_address'       => $order->get_shipping_address_1() .' '. $order->get_shipping_address_2(),
        'x_ship_to_country'       => $order->get_shipping_country(),
        'x_ship_to_state'         => $order->get_shipping_state(),
        'x_ship_to_city'          => $order->get_shipping_city(),
        'x_ship_to_zip'           => $order->get_shipping_postcode(),
```

```
        'x_cancel_url'          => wc_get_checkout_url(),
        'x_freight'           => $order->get_total_shipping(),
        'x_cancel_url_text'      => 'Cancel Payment',
        'secret_key'           => $this->secret_key,
        'app_key'             => $this->app_key,
        // 'cart'              => WC()->cart->get_cart(),
        'products'            => $products,
        'product_subscription'    => $productHasSubscription,
        'sandbox'             => $this->enabled,
        'plugin'             => 'wordpress_plugin v1.0.2'
    );

    $data = base64_encode(json_encode($args));

    $redirect = 'https://quxpay.com/login-or-register?t='.$data;

}
```

FAQs

Troubleshooting
Here are some solutions to common problems developers might encounter:

How to Use:
1. Download the plugin from our website's download page.
2. In the WordPress admin, go to Plugins > Add Plugin > Upload the downloaded plugin from our website.
3. Activate the plugin.
4. Add the API Key and Secret Key in WooCommerce > Settings > Payments > QUX® Payment > Save.

Where to Get API Key and Secret Key:
You can obtain them from here: https://quxpay.com/biller-api

I Got an Error:
- Make sure you are registered as a Merchant on https://qux.tv
- Add a store on QUX® (https://qux.tv)
- If the error persists, please contact QUX® Support.

<u>Glossary</u>

API Key: A unique identifier generated by the merchant of the website in QUX®, used for authentication when making API requests.

Secret Key: A confidential key generated by the merchant of the website in QUX®, , used alongside the API Key for secure authentication and access control.

Bearer Token: A token obtained when a user logs in to QUXPay QUX®, used to authenticate API requests on behalf of the user.