Xpresslitecoin.com

Xpresslitecoin Integration Guide

This guide describes how to integrate Xpresslitecoin on a website or application.

Version 1.0

Table of Contents

1. ABOUT THIS GUIDE

1.1. Objectives and target audience

This guide provides details on how to integrate xpresslitecoin.com to your website using the xpresslitecoin.js or API. It is intended for users who have a good knowledge HTML, TCP.

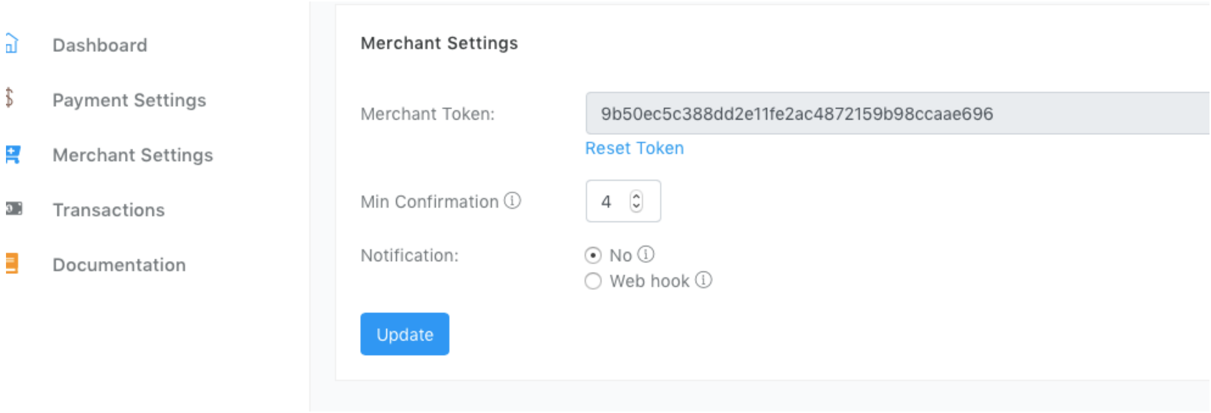
2. Introduction

Xpresslitecoin.com provides with two methods two to integrate litecoin payments to your website. You can either integrate it with our xpresslitecoin javascript library or using our API to fully customize the payment experience.

With our xpresslitecoin javascript library, we provide you with a predefined template for accepting payments on your website, however if you wish to add your own custom buttons to your website you can make user of our API.

2.1. Prerequisite

First of all, you will need to register on our website https://xpresslitecoin.com and get your merchant_token from the merchant settings page.



Merchant Settings

Merchant Token: 9b50ec5c388dd2e11fe2ac4872159b98ccaae696

Min Confirmation ⓘ 4

Notification: ● No ⓘ ○ Web hook ⓘ

Figure 1

According to Figure 1 our merchant_token is:

9b50ec5c388dd2e11fe2ac4872159b98ccaae696

Please keep this token safe, since anyone having this token can make call to our api.

Second, you must set the desired "Min Confirmation", this value represents the min Litecoin confirmations are required for you to accept a particular transaction as confirmed, however, for us for a transaction to be valid we need at least 7 confirmations before we can pay you out. It is recommended to never let this value below 4.

If you wish, you can configure webhook to notify you upon a received payment. Webhooks can be configured. The webhook will post the following data to your configured url:

```
{
   'transactionID' => [The transaction ID],
   'transactionStatus' => "confirmed",
   'amount' => [The amount received],
   'amountreceivedto' => [ltc address on which fund was received]
}
```

3. Integration

As mentioned earlier you have two ways of integrating xpresslitecoin payment gateway to your website.

- xpresslitecoin.js + API
- API

The advantage of xpresslitecoin.js over using only the API is that it already has predefined buttons and widgets to help your customers go through a smooth and seamless payment process. This is handy for website owners who do not have an advance knowledge of programming and making use of API calls. On the other hand, making use of the API, allows you to fully customize your payment process and integrate with your business logic.

3.1. xpresslitecoin.js + API

Integrating xpresslitecoin.js with your website is very easy all you need is to include the script to your website as follows:

```
<script type="text/javascript" src="xpresslitecoin.js"></script>
```

By default, the script will render all html elements bearing id "xpl-donate" or "xpl-paynow" to "donate" and "paynow" buttons. It is advised not to have both donate button and paynow button on the same page as the script will return an error.

When clicked on the button, the script will send the request to your application backend which will forward the request to out api. Your application will act as a proxy which bears the "merchant_token", this way the "merchant_token" is not exposed to the client side since it is handled by your application backend.

The xpresslitecoin.js file can be downloaded via the following url:

https://xpresslitecoin.com/download/xpresslitecoinjs/xpresslitecoin.js.gz

3.1.1. Paynow example

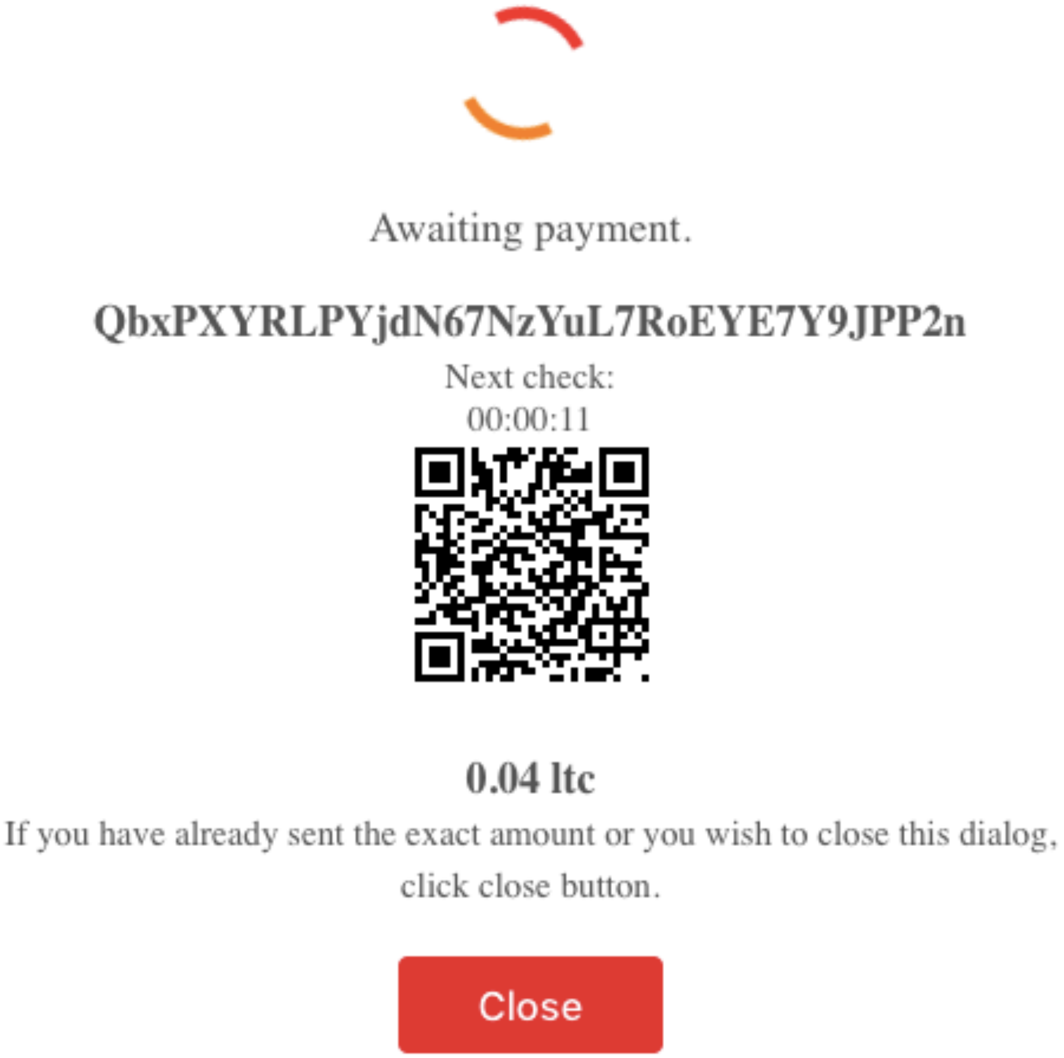
Below is an example code of a paynow button integration:

```
<!DOCTYPE html>
<html>
 <head>
    <meta charset="UTF-8">
      <title>Paynow</title>
 </head>
    <body>
    <button id="xpl-paynow" data-amount="0.04" data-
invoiceid="12345" data-
redirectsuccess="/success/">Paynow</button>

    <script type="text/javascript"
src="xpresslitecoin.js"></script>

    </body>
</html>
```

The paynow button has several mandatory attributes:

Attribute	**Meaning**
data-amount	The amount to charge for.
data-invoice	Invoice ID (generated from your backend) if you wish to process on your backend side.
data-redirectsuccess	The page to redirect to when the payment is received and if the payment page is still open.

The above code will render the following html when Paynow button is clicked:



Awaiting payment.

QbxPXYRLPYjdN67NzYuL7RoEYE7Y9JPP2n

Next check:
00:00:11

0.04 ltc

If you have already sent the exact amount or you wish to close this dialog, click close button.

Figure 2

When the client clicks on the paynow button, the xpresslitecoin.js library will post to your backend on the url /xpresslitecoin/ the attributes values "data-amount" and "data-invoice". If you need to validate the amount and the invoice data you can do it on the backend side and return a 5XX error code if the values are not correct. If the values are good, you will need to create a transaction with our API.

Below is an example of a nodejs backend code that accepts post to "/xpresslitecoin/ receive values when client clicks paynow and once validated send the data to our API to initiate a transaction.

```
var express = require('express');
var app = express();
var bodyParser = require('body-parser');
var request = require('request');

app.use(express.static('public'));
app.use(bodyParser.json());
app.use(bodyParser.urlencoded({ extended: true}));

var api_url = "https://xpresslitecoin.com/api1.0/";

const api_token = "9b50ec5c388dd2e11fe2ac4872159b98ccaae696"; //
Merchant_token

app.post("/xpresslitecoin/",function(req,res){
   // If create a transaction
    if (req.body.type == "createTransaction"){
       // We can check if the values are ok before we initiate the
transaction
        request.post(api_url +"transaction/create/",
        {"headers":{'Authorization':"Token "+api_token},
        form:{amount:req.body.amount}},
        function(err,httpResponse,body){
            body = JSON.parse(body);
            if (httpResponse.statusCode == 200){
       //We send to the payment page the ltcaddress generated and
the trxid
                res.send({"addr":body.ltcaddress,
"trxid":body.transaction_id});
            }else{
                //We send error if the transaction request failed
                res.status(500);
                res.send("Error");
            }
        });

    }else if (req.body.type == "checkIfReceived") {
       //Checking if payment is received
        console.log(req.body.ltcaddress);
        request.get(api_url +"transaction/isconfirmed/" +
req.body.ltcaddress + "/",
        {"headers":{'Authorization':"Token "+api_token}},
        function(err,httpResponse,body){
            body = JSON.parse(body);
            if (httpResponse.statusCode == 200){
                //Return Result
                console.log(body.Result)
                res.send(body.Result);
            }else{
                res.status(500);
                res.send("Error");
            }
        });
    }
})

var server = app.listen(80,function(){
    var host = server.address().address
    var port = server.address().port

    console.log("Listening at http://%s:%s",host,port)
})
```

Note that every 15 sec xpresslitecoin.js will query your backend on /xpresslitecoin/ by posting the post "type=checkIfReceived" and the relevant "ltcaddress" to check if the payment has been received or not.

To differentiate between a paynow button and a donate button, the data "donation=false" or "donation=true" is posted.

Data posted to /xpresslitecoin/ when paynow is clicked:

Data	**Value**
amount	value of "data-amount"
donation	false
invoice_id	value of "data-invoiceid"
type	"createTransaction"

Data posted to /xpresslitecoin/ every 15 sec when awaiting payment:

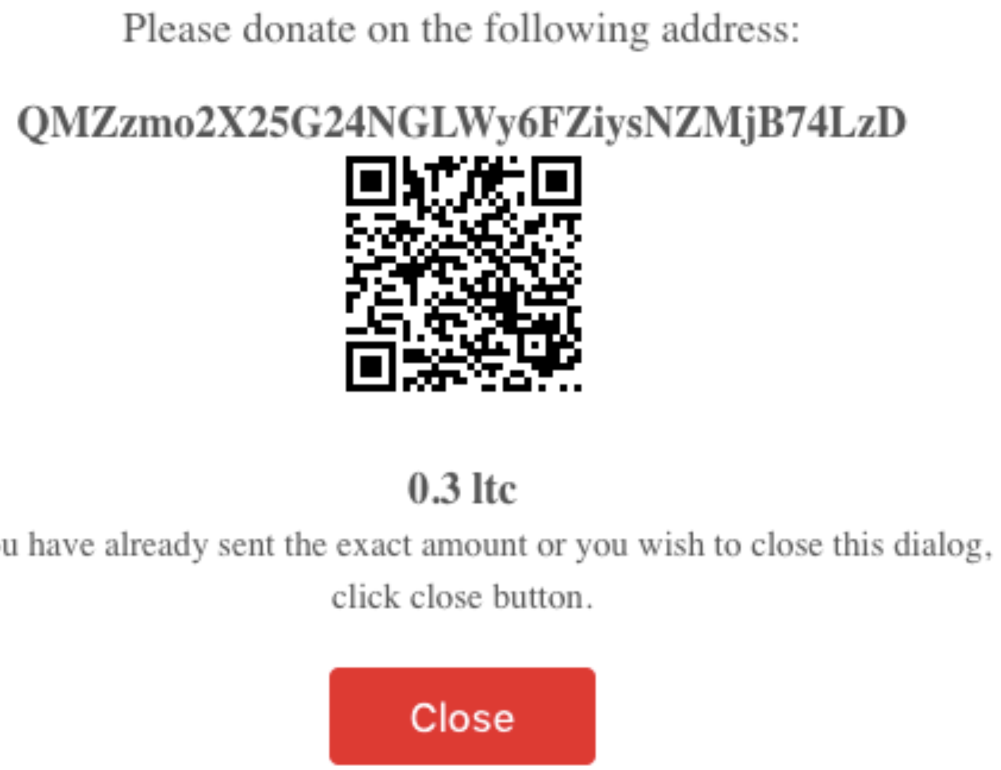
Data	**Value**
ltcaddress	ltcaddress received by API when creating transaction
type	"checkIfReceived"

3.1.2. Donate button example

Below is an example of the donate button

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Donate</title>
  </head>
  <body>
      <button id="xpl-donate">Donate</button>
  <script type="text/javascript"
src="xpresslitecoin.js"></script>
</body>
</html>
```

The above code will render the button as follows:



Please donate on the following address:

QMZzmo2X25G24NGLWy6FZiysNZMjB74LzD

0.3 ltc

If you have already sent the exact amount or you wish to close this dialog, click close button.

Figure 3

Data posted to /xpresslitecoin/ when donate button is clicked:

Data	**Value**
amount	value of "data-amount"
donation	true
invoice_id	null
type	"createTransaction"

3.2. API

You can also make use of Xpresslitecoin APIs without using the xpresslitecoinJS script, this gives you more advance control on the flow of the payment process you want to present your customer. You can get more information on the API's on https://xpresslitecoin.com dashboard's documentation section.