NICEPAY ENTERPRIZE

Integration Quick Start Guide

Credit Card

VERSION 1.15.0

26 May 2016

CONFIDENTIAL

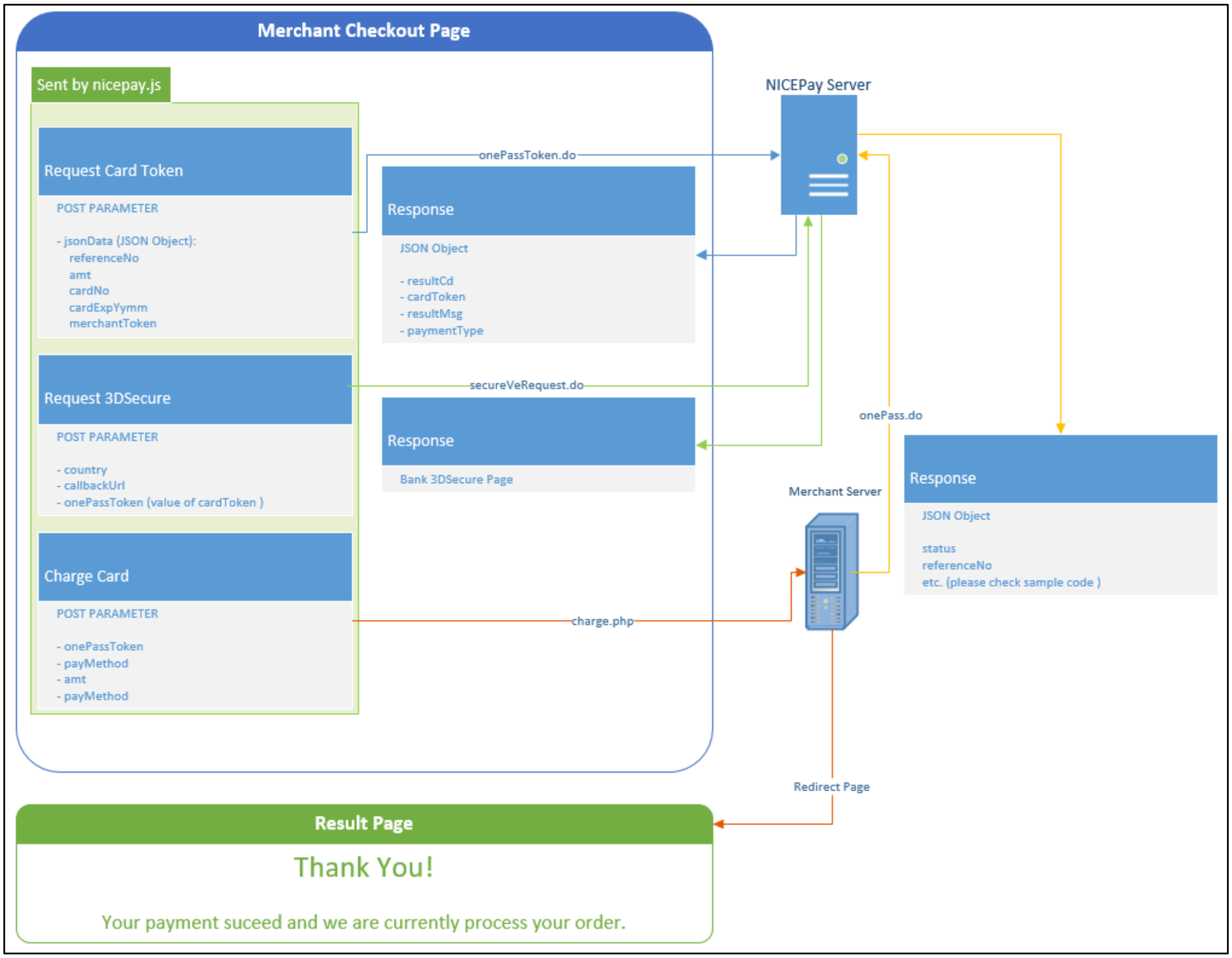
Confidentiality

The information in this document is confidential to the person or company to whom it is addressed and should not be disclosed to any other person. It may not be reproduced in whole, or in part, nor may any of the information contained therein be disclosed without the prior consent of the directors of PT IONPAY NETWORKS. A recipient may not solicit, directly or indirectly (whether through an agent or otherwise) the participation of another institution or person without the prior approval of the directors of the PT IONPAY NETWORKS. Any form of reproduction, dissemination, copying, disclosure, modification, distribution and or publication of this material is strictly prohibited.

Table of Contents

Overview

Payment Flow



Hands On

1. Configure Payment Page (checkout.php)

Generate merchantToken, amt, and referenceNo

First, we need to include library to generate merchantToken, amt & referenceNo in The Server Side.

```
<?php
include_once "lib/NicepayLib.php";

$amount = 100;
$referenceNo = 'Invoice-001';

$nicepay = new NicepayLib();
$nicepay->set('amt', $amount);
$nicepay->set('referenceNo', $referenceNo);
$nicepay->merchantToken();
?>
```

Then, we echo the generated amt, referenceNo and merchantToken

```
<!DOCTYPE html>
<html lang="en">
    <head>
      <title>NICEPAY - Secure Checkout</title>
      <link rel='stylesheet' href='index.css' type='text/css'/>
      <!-- Please include featherlight.js for lightbox demo -->
      <link rel='stylesheet' href='featherlight.min.css' type='text/css'/>
      <meta charset="UTF-8">
      <!-- Please include nicepay.js and jquery in the head block -->
      <script type="text/javascript" src="jquery.1.11.3.min.js"></script>
      <script type="text/javascript" src="nicepay.js"></script>
      <script type="text/javascript" src="featherlight.min.js"></script>
    </head>
    <body>
    <div class="wrapper">
      <div class="btn-group" id="form-selector">
        <button type="button" class="btn btn-selector active" id="va-btn">Bank Transfer</button>
        <button type="button" class="btn btn-selector" id="cc-btn">Credit Card </button>
      </div>
      <div class="form" id="va-form">
        <form name="vaForm" action="charge.php" method="post">
        <h2 class="form-title">Bank Transfer</h2>
        <h4>Total</h4>
        <hr>
        <h3>Rp. 12.000,00</h3>
        <hr>
         <select name="bankCd">
            <option value="CENA">BCA</option>
            <option value="BNIN">BNI</option>
            <option value="BMRI">Mandiri</option>
            <option value="BBBA">Permata</option>
            <option value="IBBK">BII Maybank</option>
            <option value="HNBN">KEB Hana Bank</option>

            <!-- Not available yet, coming soon-->
            <option disabled>────────── option>
            <option value="BNIA" disabled>CIMB Niaga</option>
            <option value="BRIN" disabled>BRI</option>
            <option value="BDMN" disabled>Danamon</option>
          </select>
          <input type="hidden" name="payMethod" value="02">
        <button type="submit" class="btn-submit" id="va">Get Virtual Account!</button>
        </form>
      </div>

      <div class="form" id="cc-form">
        <form id="ccForm" action="charge.php" method="post">
        <h2 class="form-title">Credit Card</h2>
        <h4>Total</h4>
        <hr>
        <h3>Rp. 100,00</h3>
        <hr>
        <input name="billingNm" type="text" class="input-std" id="name" placeholder="Cardholder's Name">
        <input name="cardNo" type="text" class="input-std" id="number" placeholder="Card Number" maxlength="16">
        <input name="cardCvv" type="password" class="input-half" id="cvc" placeholder="CVV" maxlength="3">
        <input name="cardExpYymm" type="text" class="input-quarter" id="exp-month" placeholder="YYMM" maxlength="4">
        <input type="hidden" name="payMethod" value="01" id="payMethod">
        <input type="hidden" name="referenceNo" value="<?php echo $referenceNo; ?>" id="referenceNo">
        <input type="hidden" name="amt" value="<?php echo $amount; ?>" id="amt">
        <input type="hidden" name="resultMsg" value="" id="resultMsg">
        <input type="hidden" name="resultCd" value="" id="resultCd">
        <input type="hidden" name="onePassToken" id="onePassToken" value="">
        <button type="button" class="btn-submit" id="cc" onclick="javascript:nicepay('<?php echo $nicepay->merchantToken(); ?>', '<?php echo $nicepay->get('iMid'); ?>');">Make Payment!</button>
      </form>
      </div>
    </div>

    <script type="text/javascript" src="index.js"></script>
    <body>
</html>
```

Make sure you have all <**input**> tag above and do not change **name, id** and **type** attributes. If you have to change the **name, id** and **type** attributes, you need to change **nicepay.js** as well.

2. Configure Global Parameters (NicepayConfig.php)

Configure MID and Merchant Key

Define the NICEPAY_IMID and NICEPAY_MERCHANT_KEY given by Nicepay Representative. If you don’t have one, please contact **it@nicepay.co.id**.

Setting Debit Process URL

Debit Process Url is URL of Notification Handler, in example file named **notification-handler.php**. Notification Handler should listen to notification, check status to nicepay and update the latest payment status.

Hereby example if your domain is www.example.com and **notification-handler.php** located in web application root directory.

```
define("NICEPAY_DBPROCESS_URL",      "http://example.com/notification-handler.php");
            // Merchant's notification handler URL
```

Notification Handler Url will be visited by NicePay Notification Engine, not human nor customer. And it should be accessible via public internet. If you still develop in your localhost. You can set Notification Handler Url using HTTP Responder free service, for example;

```
define("NICEPAY_DBPROCESS_URL",      "http://httpresponder.com/nicepay");
     // Merchant's notification handler URL
```

You can check the Notification in http://httpresponder.com/nicepay?view

```
<?php
/*
 * ____________________________________________________________
 *
 * Copyright (C) 2016 NICE IT&T
 *
 *
 * This config file may used as it is, there is no warranty.
 *
 * @ description : PHP SSL Client module.
 * @ name        : NicepayConfig.php
 * @ author      : NICEPAY I&T (tech@nicepay.co.kr)
 * @ date        :
 * @ modify      : 26.05.2016
 *
 * 2016.05.26 Update Log
 *
 * ____________________________________________________________
 */

// Please set the following

define("NICEPAY_IMID",               "IONPAYTEST");
     // Merchant ID
define("NICEPAY_MERCHANT_KEY",       "33F49GnCMS1mFYlGXisbUDzVf2ATWCl9k3R++d5hDd3Frmuos/XLx8XhXpe+LDYAbpGKZYSwtlyyLOtS/8aD7A=="); // API Key

define("NICEPAY_CALLBACK_URL",       "http://httpresponder.com/nicepay");
     // Merchant's result page URL
```

```
define("NICEPAY_DBPROCESS_URL",     "http://httpresponder.com/nicepay");     // Merchant's notification handler URL

/* TIMEOUT - Define as needed (in seconds) */
define( "NICEPAY_TIMEOUT_CONNECT", 15 );
define( "NICEPAY_TIMEOUT_READ", 25 );


// Please do not change

define("NICEPAY_PROGRAM",           "NicepayLite");
define("NICEPAY_VERSION",           "1.11");
define("NICEPAY_BUILDDATE",         "20160309");
define("NICEPAY_REQ_VA_URL",        "https://www.nicepay.co.id/nicepay/api/onePass.do");     // Request Virtual Account API URL
define("NICEPAY_3DSECURE_URL",      "https://www.nicepay.co.id/nicepay/api/secureVeRequest.do");     // 3D Secure API URL
define("NICEPAY_CANCEL_VA_URL",     "https://www.nicepay.co.id/nicepay/api/onePassAllCancel.do");     // Cancel Virtual Account API URL
define("NICEPAY_CHARGE_URL",        "https://www.nicepay.co.id/nicepay/api/onePass.do");     // Charge Credit Card API URL
define("NICEPAY_CANCEL_URL",        "https://www.nicepay.co.id/nicepay/api/onePassAllCancel.do");     // Cancellation URL
define("NICEPAY_ORDER_STATUS_URL",  "https://www.nicepay.co.id/nicepay/api/onePassStatus.do");     // Check payment status URL
define("NICEPAY_READ_TIMEOUT_ERR",  "10200");

/* LOG LEVEL */

define("NICEPAY_LOG_CRITICAL", 1);
define("NICEPAY_LOG_ERROR", 2);
define("NICEPAY_LOG_NOTICE", 3);
define("NICEPAY_LOG_INFO", 5);
define("NICEPAY_LOG_DEBUG", 7);
```

3. Configure nicepay.js

Setting 3dsecure.php file location

3dsecure.php responsible to automatically send payment form in parent window after 3D Secure Process close itself. We should set 3dsecure.php file location before processing 3D Secure transaction.

```
function cardPaySubmit(){
    var ccForm = document.getElementById("ccForm");
        ccForm.submit();
}

//3DSecure Result Setting Value in 3dsecure.php
function setAcsInfo(name,values) {
      var names = document.getElementById(name);
      names.value = values;
}

function getOnePassToken(merchantToken,imid)
{
    var ajaxUrl = "https://www.nicepay.co.id/nicepay/api/onePassToken.do";
    var dataObject = new Object();
    var paymentType = "1";
    var iMid = imid;

    dataObject.iMid = iMid;
    dataObject.referenceNo = $("#referenceNo").val();
    dataObject.amt = $("#amt").val();

    var threeDsecurephp = "http://localhost/nicepay-enterprize/3dsecure.php";
    dataObject.cardNo = $("#number").val();
    dataObject.cardExpYymm = $("#exp-month").val();
    dataObject.merchantToken = merchantToken;

    $.ajax({
        url : ajaxUrl,
        type : "post",
        crossDomain: true,
        data : "jsonData=" + JSON.stringify(dataObject),
        dataType: "jsonp",
        timeout : 30000,
        success : function(data) {
            var get_data = eval(data);
            var resultCd = get_data["resultCd"];
            var resultMsg = get_data["resultMsg"];
            if(resultCd == "0000"){
                $("#onePassToken").val(get_data["cardToken"]);
                var amount = $("#amt").val();
                var onePassToken = get_data["cardToken"];
                //3d secure
                if(onePassToken == null || onePassToken == ""){
                    alert("onePassToken failed.")
                    return;
                }
                $("#onePassToken").val(onePassToken);
                    var targetUrl = "https://www.nicepay.co.id/nicepay/api/secureVeRequest.do?country=360" +
                        "&callbackUrl="+threeDsecurephp+
                        "&onePassToken=" + onePassToken;
                    $.featherlight({iframe: targetUrl, iframeMaxWidth: '100%', iframeWidth: 450, iframeHeight: 450});
            }
            else{
                alert(resultMsg);
            }
        },
        error : function(request,status,error){
            alert(status);
        }
    });
}

    /**
        cancel Button
    */
    function cancelOK(sel){
        var cancelForm = document.getElementById("cancelForm");
        if(sel == "1"){
            //void data
            document.getElementsByName("cancelType")[0].value = 1;
        }else{
            //refund Data
            document.getElementsByName("cancelType")[0].value = 2;
        }
        cancelForm.submit();
    }

    /**
        payment button
    */
    function nicepay(merchantToken,imid) {
        var ccForm = document.ccForm;
        var ccPayMethod = "01";
        //card payment
        if( ccPayMethod== "01"){
            //OnePass Token
            getOnePassToken(merchantToken,imid);
        }
        else{
            return;
        }
    }

    /**
        OnePass Inquiry Submit
    */
    function inquiry()
    {
               var onePassInquiryForm = document.getElementById("onePassInquiryForm");
               onePassInquiryForm.submit();
           }

           function errMsg(code,message)
           {
               console.log("3D Secure failed");
               console.log(code+" : "+message);
           }
```

4. Charge Card Using Card Token (charge.php)

After 3D Secure Process, the **3dsecure.php** will automatically send form to **charge.php** to charge customer card balance. Please make sure the **referenceNo** and **amt** sent while request card token is identical to **referenceNo** and **amt** sent while charging the card.

```
<?php
/*
 * ____________________________________________________________
 *
 * Copyright (C) 2016 NICE IT&T
 *
 * Please do not modify this module.
 * This module may used as it is, there is no warranty.
 *
 * @ description : PHP SSL Client module.
 * @ name        : charge.php
 * @ author      : NICEPAY I&T (tech@nicepay.co.kr)
 * @ date        :
 * @ modify      : 16.05.2016
 *
 * 2016.02.22 Update Log
 * Please contact it.support@ionpay.net for inquiry
 *
 * ____________________________________________________________
 */
```

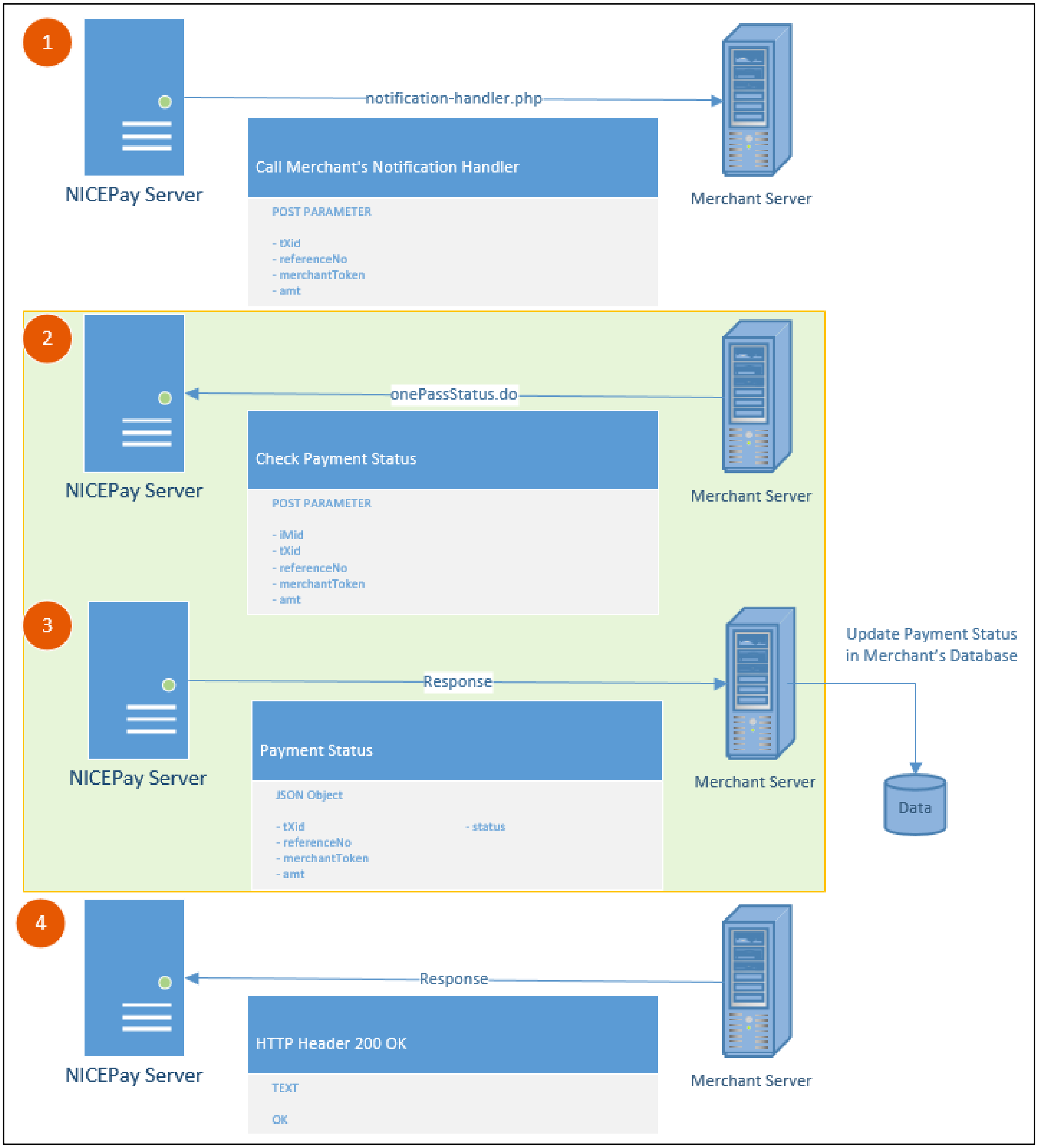
```php

// Include Config File
include_once "lib/NicepayLib.php";
$nicepay = new NicepayLib();

/*
 * ____________________________________________________________
 *
 * Credit Card Payment Method
 * ____________________________________________________________
 */
if(isset($_POST['payMethod'])
        && $_POST['payMethod'] == '01'
        && isset($_POST['billingNm'])
        && $_POST['billingNm']
        && isset($_POST['onePassToken'])
        && $_POST['onePassToken']
        )
    {

        $billingNm      = $_POST['billingNm'];
        $onePassToken   = $_POST['onePassToken'];
        $cardExpYymm    = $_POST['cardExpYymm'];
        $cardCvv        = $_POST['cardCvv'];

        // Populate Mandatory parameters to send
        $nicepay->set('payMethod', '01');
        $nicepay->set('currency', 'IDR');
        $nicepay->set('amt', 100); // Total gross amount
        $nicepay->set('referenceNo', 'Invoice-001'); // Invoice Number or Referenc Number Generated by merchant
        $nicepay->set('description', 'Payment of Invoice No '.$nicepay->get('referenceNo')); // Transaction description

        $nicepay->set('billingNm', 'John Doe'); // Customer name
        $nicepay->set('billingPhone', '02112345678'); // Customer phone number
        $nicepay->set('billingEmail', 'john@example.com'); //
        $nicepay->set('billingAddr', 'Jl. Jend. Sudirman No. 28');
        $nicepay->set('billingCity', 'Jakarta Pusat');
        $nicepay->set('billingState', 'DKI Jakarta');
        $nicepay->set('billingPostCd', '10210');
        $nicepay->set('billingCountry', 'Indonesia');

        $nicepay->set('deliveryNm', 'John Doe'); // Delivery name
        $nicepay->set('deliveryPhone', '02112345678');
        $nicepay->set('deliveryEmail', 'john@example.com');
        $nicepay->set('deliveryAddr', 'Jl. Jend. Sudirman No. 28');
        $nicepay->set('deliveryCity', 'Jakarta Pusat');
        $nicepay->set('deliveryState', 'DKI Jakarta');
        $nicepay->set('deliveryPostCd', '10210');
        $nicepay->set('deliveryCountry', 'Indonesia');
        $nicepay->set('onePassToken', $onePassToken);
        $nicepay->set('cardExpYymm', $cardExpYymm);
        $nicepay->set('cardCvv', $cardCvv);

        // Send Data
        $response = $nicepay->chargeCard();


        // Response from NICEPAY
//        var_dump($response);
//        exit();
        if (isset($response->resultCd) && $response->resultCd == "0000") {
            echo "<pre>";
            echo "tXid              : $response->tXid\n";
            echo "callbackUrl       : $response->callbackUrl\n";
            echo "description       : $response->description\n";
            echo "payment date      : $response->transDt\n"; // YYYYMMDD
            echo "payment time      : $response->transTm\n"; // HH24MISS
            echo "result code       : $response->resultCd\n";
            echo "result message    : $response->resultMsg\n";
            echo "reference no      : $response->referenceNo\n";
            echo "payment method    : $response->payMethod";
            echo "</pre>";
            // var_dump($response);
        } elseif(isset($response->resultCd)) {
            // API data not correct or error happened in bank system, you can redirect back to checkout page or echo error message.
            // In this sample, we echo error message
            echo "<pre>";
            echo "Oops! Something happened, please notice your system administrator.\n\n";
            echo "result code       : $response->resultCd\n";
                   echo "result message    : $response->resultMsg\n";
                   echo "</pre>";
               } else {
                   // Timeout, you can redirect back to checkout page or echo error message.
                   // In this sample, we echo error message
                   echo "<pre>Connection Timeout. Please Try again.</pre>";
               }
           }
           elseif(isset($_POST['payMethod']) && $_POST['payMethod'] == '01') {
               echo "Please fill all card data field";
               // print_r($_POST);
           }
           else {
               echo "Unknown payment method";
           }
```

5. Configure Notification Handler (notification-handler.php)

Overview

Notification Handler should listen to notification and check payment status to Nicepay server in order to update the latest payment status in merchant database.

Notification Flow



Updating Payment Status (notification-handler.php)

Right After check payment status, you should update your payment status in database and return HTTP Header 200 OK to Nicepay notification engine.

```php
<?php
/*
 * ____________________________________________________________
 *
 * Copyright (C) 2015 NICE IT&T
 *
 * Please do not modify this module.
 * This module may used as it is, there is no warranty.
 *
 * @ description : PHP SSL Client module.
 * @ name        : notification-handler.php
 * @ author      : NICEPAY I&T (tech@nicepay.co.kr)
 * @ date        :
 * @ modify      : 26.05.2016
 *
 * 2016.05.26 Update Log
 *
 * ____________________________________________________________
 */
// Include the Transferpay class
include_once('lib/NicepayLib.php');

$nicepay = new NicepayLib();

// Listen for parameters passed
$pushParameters = array('tXid',
                    'referenceNo',
                    'merchantToken',
                    'amt'
                    );

$nicepay->extractNotification($pushParameters);

$iMid               = $nicepay->iMid;
$tXid               = $nicepay->getNotification('tXid');
$referenceNo        = $nicepay->getNotification('referenceNo');
$amt                = $nicepay->getNotification('amt');
$pushedToken        = $nicepay->getNotification('merchantToken');


$nicepay->set('tXid', $tXid);
$nicepay->set('referenceNo', $referenceNo);
$nicepay->set('amt', $amt);
$nicepay->set('iMid',$iMid);

$merchantToken = $nicepay->merchantTokenC();
$nicepay->set('merchantToken', $merchantToken);

// <RESQUEST to NICEPAY>
$paymentStatus = $nicepay->checkPaymentStatus($tXid, $referenceNo, $amt);

// <RESPONSE from NICEPAY>
// Please update the payment status in your database right after you get the latest payment status
if($pushedToken == $merchantToken) {
```

```
            // Print only OK, no HTML, no UI, no beauty poem. Also make sure HTTP code = 200
            echo "OK";
            // Update the payment status in your database based on $paymentStatus->StatusDescription,
            // Send email notification to customer to notify payment succeed

            // <RESPONSE from NICEPAY>
            // Please update the payment status in your database right after you get the latest payment status
            echo "<pre>";
            echo "$paymentStatus->status\n"; // This is Payment Status main reference to update Payment Status in merchant database
            /**
            **=============================================================================
            ** Credit Card
            **=============================================================================
            ** $paymentStatus->status == 0 // Success
            ** $paymentStatus->status == 1 // Failed
            ** $paymentStatus->status == 2 // Void or Refund
            ** $paymentStatus->status == 9 // Initialization or Unpaid
            **=============================================================================
            *
            **=============================================================================
            ** Virtual Account
            **=============================================================================
            ** $paymentStatus->status == 0 // Paid
            ** $paymentStatus->status == 3 // Unpaid
            ** $paymentStatus->status == 4 // Expired
            **=============================================================================
            */
            echo "$paymentStatus->tXid\n";
            echo "$paymentStatus->iMid\n";
            echo "$paymentStatus->referenceNo\n";
            echo "$paymentStatus->amt\n";
            // var_dump for more information
            // var_dump($paymentStatus);
            echo "</pre>";

        }
```

As we see highlighted code above, we should update status by using `$paymentStatus->status` and recommended to validate `$paymentStatus->amt`.

Hereby the example of change to the highighted code above;

```
if($pushedToken == $merchantToken) {
      // Print only OK, no HTML, no UI, no beauty poem. Also make sure HTTP code = 200
      echo "OK";
      switch ($paymentStatus->status) {

            case '0':
            // check if $paymentStatus->amt == amount in merchant database
            // if true, update payment status in merchant database as Paid
            // send email to customer

            break;

            case '1':
```

```
            // update payment status in merchant database as Failed
            // send email to customer
            break;

            case '2':
            // update payment status in merchant database as Refund or Void
            // send email to customer
            break;

            case '9':
            // update payment status in merchant database as Pending
            break;

            default:
            // payment status  status in merchant database as Error
            // send email to system administrator
      }
}
```

Discover More

Discover more advanced options by checking our documentation or contact Integration Support it@nicepay.co.id.