<?php

/\*\*

\* Do not forget to set these to your Account credentials.

\* It would be better to store these as an admin setting.

\*\*/

define('MERCHANT\_ID', '');

define('MERCHANT\_PASSWORD', '');

define('ENV\_TEST', 0);

define('ENV\_LIVE', 1);

$environment = ENV\_TEST;

?>

<?php

$errors = array();

$is\_link = false;

$parameters = array(

'merchantid' => MERCHANT\_ID,

'txnid' => '000001',

'amount' => 100,

'ccy' => 'PHP',

'description' => 'My order description.',

'email' => 'sample@merchant.ph',

);

$fields = array(

'txnid' => array(

'label' => 'Transaction ID',

'type' => 'text',

'attributes' => array(),

'filter' => FILTER\_SANITIZE\_STRING,

'filter\_flags' => array(FILTER\_FLAG\_STRIP\_LOW),

),

'amount' => array(

'label' => 'Amount',

'type' => 'number',

'attributes' => array('step="0.01"'),

'filter' => FILTER\_SANITIZE\_NUMBER\_FLOAT,

'filter\_flags' => array(FILTER\_FLAG\_ALLOW\_THOUSAND, FILTER\_FLAG\_ALLOW\_FRACTION),

),

'description' => array(

'label' => 'Description',

'type' => 'text',

'attributes' => array(),

'filter' => FILTER\_SANITIZE\_STRING,

'filter\_flags' => array(FILTER\_FLAG\_STRIP\_LOW),

),

'email' => array(

'label' => 'Email',

'type' => 'email',

'attributes' => array(),

'filter' => FILTER\_SANITIZE\_EMAIL,

'filter\_flags' => array(),

),

);

if (isset($\_POST['submit'])) {

// Check for set values.

foreach ($fields as $key => $value) {

// Sanitize user input. However:

// NOTE: this is a sample, user's SHOULD NOT be inputting these values.

if (isset($\_POST[$key])) {

$parameters[$key] = filter\_input(INPUT\_POST, $key, $value['filter'],

array\_reduce($value['filter\_flags'], function ($a, $b) { return $a | $b; }, 0));

}

}

// Validate values.

// Example, amount validation.

// Do not rely on browser validation as the client can manually send

// invalid values, or be using old browsers.

if (!is\_numeric($parameters['amount'])) {

$errors[] = 'Amount should be a number.';

}

else if ($parameters['amount'] <= 0) {

$errors[] = 'Amount should be greater than 0.';

}

if (empty($errors)) {

// Transform amount to correct format. (2 decimal places,

// decimal separated by period, no thousands separator)

$parameters['amount'] = number\_format($parameters['amount'], 2, '.', '');

// Unset later from parameter after digest.

$parameters['key'] = MERCHANT\_PASSWORD;

$digest\_string = implode(':', $parameters);

unset($parameters['key']);

// NOTE: To check for invalid digest errors,

// uncomment this to see the digest string generated for computation.

// var\_dump($digest\_string); $is\_link = true;

$parameters['digest'] = sha1($digest\_string);

$url = 'https://gw.dragonpay.ph/Pay.aspx?';

if ($environment == ENV\_TEST) {

$url = 'http://test.dragonpay.ph/Pay.aspx?';

}

$url .= http\_build\_query($parameters, '', '&');

if ($is\_link) {

echo '<br><a href="' . $url . '">' . $url . '</a>';

}

else {

header("Location: $url");

}

}

}

?>

<!DOCTYPE html>

<html>

<head>

<style>

label {width: 130px; float: left;}

input {width: 250px;}

</style>

</head>

<body>

<?php if (!empty($errors)): ?>

<div class="errors">

<div class="error">

<?php echo implode('</div><div class="error">', $errors); ?>

</div>

</div>

<?php endif; ?>

<div class="form">

<form method="post">

<?php foreach ($fields as $key => $value): ?>

<div class="input">

<span class="label"><label for="<?php echo $key; ?>">

<?php echo $value['label']; ?>:</label></span>

<input type="<?php echo $value['type']; ?>"

<?php echo implode(' ', $value['attributes']); ?>

name="<?php echo $key; ?>" value="<?php echo $parameters[$key]; ?>">

</div>

<?php endforeach; ?>

<input type="submit" name="submit" value="Pay Now">

</form>

</div>

</body>

</html>