Tích hợp thanh toán MOMO vào ASP.NET

**Setup Code**

Trong file**appsetting.jso**n các bạn thêm đoạn code như sau đồng thời sửa lại các thông tin **SecretKey, AccessKey**bằng thông tin của các bạn vừa đăng ký

"MomoAPI": {

"MomoApiUrl": "https://test-payment.momo.vn/gw\_payment/transactionProcessor",

"SecretKey": "K951B6PE1waDMi640xX08PD3vg6EkVlz",

"AccessKey": "F8BBA842ECF85",

"ReturnUrl": "http://localhost:5172/Checkout/PaymentCallBack",

"NotifyUrl": "http://localhost:5172/Checkout/MomoNotify",

"PartnerCode": "MOMO",

"RequestType": "captureMoMoWallet" //payWithATM //captureMoMoWallet

},

Tạo 3 Model

public class MomoCreatePaymentResponseModel

{

public string RequestId { get; set; }

public int ErrorCode { get; set; }

public string OrderId { get; set; }

public string Message { get; set; }

public string LocalMessage { get; set; }

public string RequestType { get; set; }

public string PayUrl { get; set; }

public string Signature { get; set; }

public string QrCodeUrl { get; set; }

public string Deeplink { get; set; }

public string DeeplinkWebInApp { get; set; }

}

public class MomoExecuteResponseModel

{

public string OrderId { get; set; }

public string Amount { get; set; }

public string FullName { get; set; }

public string OrderInfo { get; set; }

}

public class MomoOptionModel

{

public string MomoApiUrl { get; set; }

public string SecretKey { get; set; }

public string AccessKey { get; set; }

public string ReturnUrl { get; set; }

public string NotifyUrl { get; set; }

public string PartnerCode { get; set; }

public string RequestType { get; set; }

}

Tạo dịch vụ Services cho MOMOService và IMOMOService

public class MomoService : IMomoService

{

private readonly IOptions<MomoOptionModel> \_options;

public MomoService(IOptions<MomoOptionModel> options)

{

\_options = options;

}

public async Task<MomoCreatePaymentResponseModel> CreatePaymentAsync(OrderInfoModel model)

{

model.OrderId = DateTime.UtcNow.Ticks.ToString();

model.OrderInfo = "Khách hàng: " + model.FullName + ". Nội dung: " + model.OrderInfo;

var rawData =

$"partnerCode={\_options.Value.PartnerCode}" +

$"&accessKey={\_options.Value.AccessKey}" +

$"&requestId={model.OrderId}" +

$"&amount={model.Amount}" +

$"&orderId={model.OrderId}" +

$"&orderInfo={model.OrderInfo}" +

$"&returnUrl={\_options.Value.ReturnUrl}" +

$"&notifyUrl={\_options.Value.NotifyUrl}" +

$"&extraData=";

var signature = ComputeHmacSha256(rawData, \_options.Value.SecretKey);

var client = new RestClient(\_options.Value.MomoApiUrl);

var request = new RestRequest() { Method = Method.Post };

request.AddHeader("Content-Type", "application/json; charset=UTF-8");

// Create an object representing the request data

var requestData = new

{

accessKey = \_options.Value.AccessKey,

partnerCode = \_options.Value.PartnerCode,

requestType = \_options.Value.RequestType,

notifyUrl = \_options.Value.NotifyUrl,

returnUrl = \_options.Value.ReturnUrl,

orderId = model.OrderId,

amount = model.Amount.ToString(),

orderInfo = model.OrderInfo,

requestId = model.OrderId,

extraData = "",

signature = signature

};

request.AddParameter("application/json", JsonConvert.SerializeObject(requestData), ParameterType.RequestBody);

var response = await client.ExecuteAsync(request);

var momoResponse = JsonConvert.DeserializeObject<MomoCreatePaymentResponseModel>(response.Content);

return momoResponse;

}

public MomoExecuteResponseModel PaymentExecuteAsync(IQueryCollection collection)

{

var amount = collection.First(s => s.Key == "amount").Value;

var orderInfo = collection.First(s => s.Key == "orderInfo").Value;

var orderId = collection.First(s => s.Key == "orderId").Value;

return new MomoExecuteResponseModel()

{

Amount = amount,

OrderId = orderId,

OrderInfo = orderInfo

};

}

private string ComputeHmacSha256(string message, string secretKey)

{

var keyBytes = Encoding.UTF8.GetBytes(secretKey);

var messageBytes = Encoding.UTF8.GetBytes(message);

byte[] hashBytes;

using (var hmac = new HMACSHA256(keyBytes))

{

hashBytes = hmac.ComputeHash(messageBytes);

}

var hashString = BitConverter.ToString(hashBytes).Replace("-", "").ToLower();

return hashString;

}

}

Đăng ký Services vào Program.CS

//Momo API Payment

builder.Services.Configure<MomoOptionModel>(builder.Configuration.GetSection("MomoAPI"));

builder.Services.AddScoped<IMomoService, MomoService>();

Thêm hàm vào Controller Payment

private IMomoService \_momoService;

private readonly IVnPayService \_vnPayService;

public PaymentController(IMomoService momoService)

{

\_momoService = momoService;

}

[HttpPost]

[Route("CreatePaymentUrl")]

public async Task<IActionResult> CreatePaymentUrl(OrderInfoModel model)

{

var response = await \_momoService.CreatePaymentAsync(model);

return Redirect(response.PayUrl);

}

Form Dữ liệu thanh toán MOMO

<**form** method="POST" **asp-action**="CreatePaymentUrl" **asp-controller**="Payment">

<input type="hidden" name="FullName" value="@User.Identity.Name" />

<input type="hidden" name="Amount" value="@Model.GrandTotal" />

<input type="hidden" name="OrderInfo" value="Thanh toán đặt hàng qua Momo tại HieuTanStore" />

<br />

<button class="btn btn-danger" name="PayUrl" type="submit">Pay with MoMo</button>

</**form**>