

Thanh toán tích hợp Paypal

Tham khảo bài viết gốc trên code project :

- <https://www.codeproject.com/Articles/870870/Using-Paypal-Rest-API-with-ASP-NET-MVC>
- <https://code.tutsplus.com/articles/paypal-integration-part-2-paypal-rest-api--cms-22917>

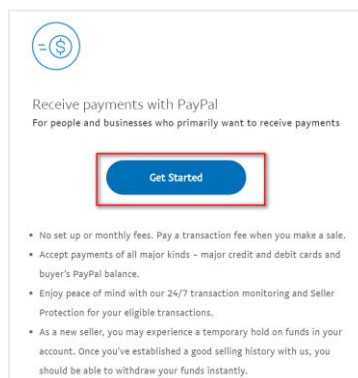
Bài viết có tham khảo và điều chỉnh bỏ bớt một số nội dung không cần thiết, cũng như hướng dẫn và giải thích chi tiết việc tạo tài khoản paypal và môi trường tích hợp để kiểm thử sandbox

Nội dung bổ sung thêm:

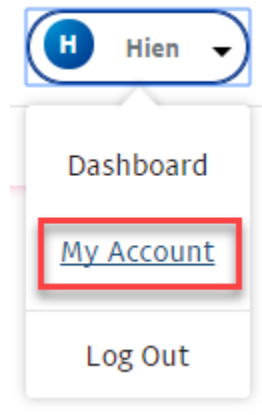
- Hướng dẫn đăng ký tài khoản tích hợp thử trên môi trường sandbox.
- Hướng dẫn chuyển từ môi trường test sandbox sang môi trường thanh toán thật trên paypal.
- Hướng dẫn cách lấy **ClientId** và **ClientSecret** key.

1.1 Đăng ký tài khoản tích hợp cho nhà phát triển

- <https://developer.paypal.com>



- Xác nhận email
- Đăng ký tích hợp ứng dụng vào website



- Tạo thêm account Sandbox dành cho người bán:

A screenshot of the PayPal Developer 'Sandbox Accounts' page. The page has a sidebar on the left with links like 'Dashboard', 'My Apps & Credentials', 'My Account', 'Sandbox', 'Accounts', 'Notifications', 'API Calls', 'IPN Simulator', 'Webhooks Events', 'Mock', and 'Webhooks Simulator'. The main content area is titled 'Sandbox Accounts' and features a 'Create Account' button highlighted with a red arrow. Below the title, there is a section for 'Questions?' with links to 'Testing Guide' and 'FAQ'. A paragraph explains how to link a sandbox account to a developer account. Below this, it says 'Total records: 2' and displays a table of existing sandbox accounts.

Email Address	Type	Country	Date Created
hyhien-facilitator@gmail.com	BUSINESS	US	16 May 2017
hyhien-buyer@gmail.com	PERSONAL	US	16 May 2017

[API Reference](#)[Support](#)[H](#)[Hien](#)

Sandbox Accounts

[Create Account](#)

Questions? Check out the [Testing Guide](#). Non-US developers should read our [FAQ](#).

To link your sandbox account to your developer account, [log in with PayPal](#) and provide your sandbox account credentials.

Total records: 2

<input type="checkbox"/> Email Address	Type	Country	Date Created	
<input type="checkbox"/> ▶ hyhien-facilitator@gmail.com	BUSINESS	US	16 May 2017	
<input type="checkbox"/> ▶ hyhien-buyer@gmail.com	PERSONAL	US	16 May 2017	

Các bạn có quyền tạo tài khoản khác.

[API Reference](#)[Support](#)[H](#)[Hien](#)

Email Address

Available.

Password(8-20 characters)

First Name(optional)

Last Name(optional)

API Reference Support Search H Hien

Payment Methods

PayPal Balance .00 USD

50000

Bank Verified Account

☒ Yes ☐ No

Credit Cards

MasterCard

Note(optional)

Create Account

- Tạo App:

PayPal Developer Docs API Reference Support Search H Hien

Dashboard

My Apps & Credentials

My Account

Sandbox

Accounts

Notifications

API Calls

IPN Simulator

REST API apps

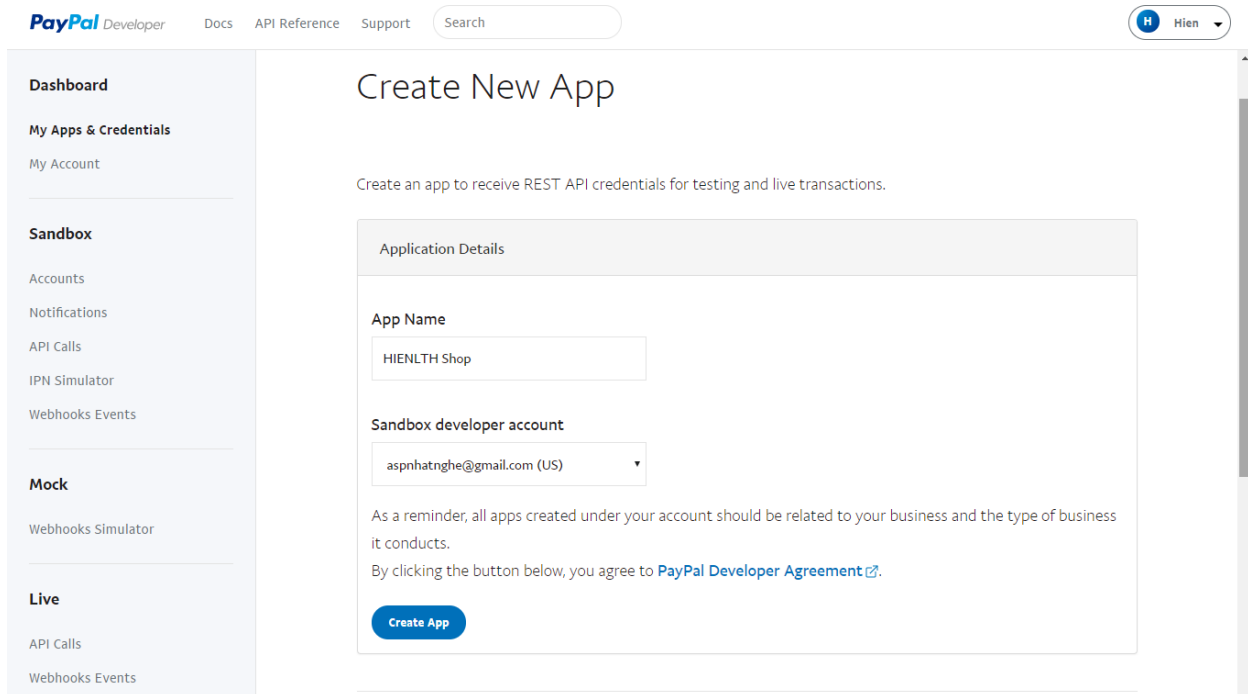
Create an app to receive REST API credentials for testing and live transactions.

Note: Features available for live transactions are listed in your [account eligibility](#).

Create App

Tạo app

App Name	Type	Actions
Create your first app to view it here.		



PayPal Developer Docs API Reference Support Search H Hien

Dashboard

My Apps & Credentials

My Account

Sandbox

Accounts

Notifications

API Calls

IPN Simulator

Webhooks Events

Mock

Webhooks Simulator

Live

API Calls

Webhooks Events

Create New App

Create an app to receive REST API credentials for testing and live transactions.

Application Details

App Name

HIENLTH Shop

Sandbox developer account

aspnhatnghe@gmail.com (US)

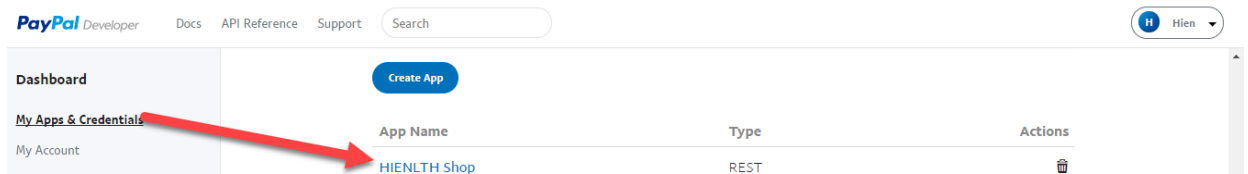
As a reminder, all apps created under your account should be related to your business and the type of business it conducts.

By clicking the button below, you agree to [PayPal Developer Agreement](#).

Create App

- Lấy thông tin Client ID và Client Secret key.

Lưu ý: Tài khoản sandbox: paypal tạo môi trường thanh toán ảo tại **sandbox.paypal.com** cho nhà phát triển gần giống với môi trường thanh toán và quản lý tiền thật **paypal.com**, nhà phát triển cần phải kiểm tra kỹ lưỡng trước khi đưa ra môi trường triển khai thật.



PayPal Developer Docs API Reference Support Search H Hien

Dashboard

My Apps & Credentials

My Account

Create App

App Name	Type	Actions
HIENLTH Shop	REST	

- Vào App

API Reference Support Search H Hien

HIENLTH Shop

[1] Sandbox Live [2]

App display name: HIENLTH Shop ✎

SANDBOX API CREDENTIALS

Sandbox account
aspnhatnghe@gmail.com

Client ID
AU2TuOSoleGYbmOBCukSgcKa5W9TN1QtdiaaGcII3yWjWpFv_oai0mFyqRRd3fyJAQnPEocfHn5hbJAI

Secret
[Show](#)

- [1] Môi trường kiểm thử tích hợp ảo cho nhà phát triển
- [2] Môi trường tích hợp thật

Support

H **Hien**

HIENLTH Shop

Sandbox Live

App display name: HIENLTH Shop

SANDBOX API CREDENTIALS

Sandbox account
aspnhatnghe@gmail.com

Client ID
AU2TuOSoleCYbmOBCukSgcKa5W9TN1QtdiaaGclI3yWjWpFv_oai0mFyqRRd3fyJAQnPEocfHn5hbJAl

Secret
Hide

Note: When you generate a new secret, you still maintain the original secret. The maximum number of client secrets is two. A client secret is either in enabled or disabled state.

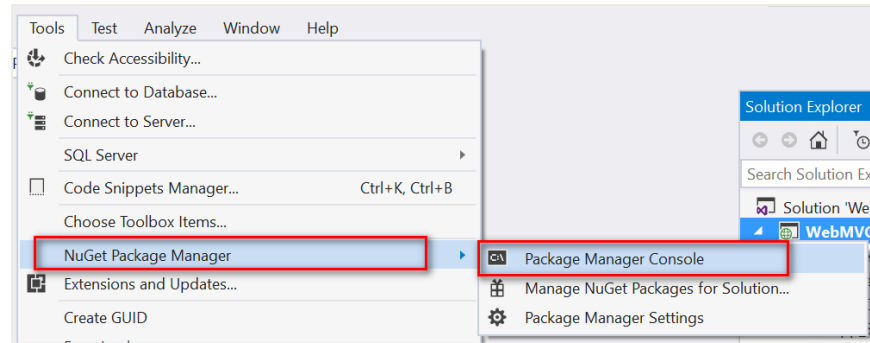
Created	Secret	Status	Action
May 17, 2017	EOEKQr7p8ueMiaZ1QgEswljj_LwCHwwCMrGxc6y29ZWLF13GD9yYAsQv77smmp2t8dDkv9SOAoniVYFa	Enabled	Disable

Generate New Secret

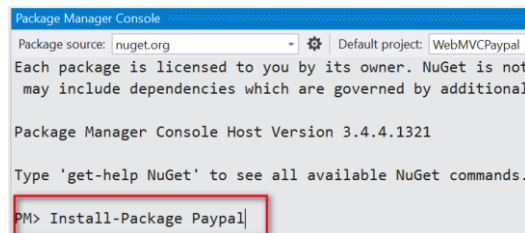
- Lưu **Client ID** và **Client Secret key** vào file text document (txt) để làm các bước tiếp theo

1.2 Tích hợp vào ứng dụng website (lưu ý: bước này đã có sẵn website bán hàng cần tích hợp thanh toán paypal hoặc tự tạo ứng dụng website ASP.NET MVC để thực hiện)

- Tạo mới project ASP NET MVC Web Application → đặt tên ứng dụng.
- Thêm thư viện **Paypal SDK for .NET** thông qua **NuGet Package Manager** hoặc thêm từ viện PayPal.dll, Newtonsoft.Json.dll từ thư mục <<thu-vien>> kèm theo bài hướng dẫn
- Chọn Tools -> Library Package Manager -> Package Manager Console



- Install-Package Paypal



1.3 Cấu hình file web.config bao gồm các thông tin thanh toán từ paypal

- **Client ID và Client Secret key** lấy ở bước trên (môi trường sandbox)

```
<configSections>
  <!-- For more information on Entity Framework configuration, visit http://go.microsoft.com/fwlink/?LinkID=237468 -->
  <section name="entityFramework"
    type="System.Data.Entity.Internal.ConfigFile.EntityFrameworkSection, EntityFramework,
    Version=6.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089" requirePermission="false" />
  <section name="paypal" type="PayPal.SDKConfigHandler, PayPal" />
</configSections>
```

- Code

```
<section name="paypal" type="PayPal.SDKConfigHandler, PayPal" />
```

- Tiếp theo


```

<configuration>
  <configSections>...</configSections>
  <connectionStrings>...</connectionStrings>
  <appSettings>...</appSettings>
  <system.web>...</system.web>
  <system.webServer>...</system.webServer>
  <runtime>...</runtime>
  <entityFramework>...</entityFramework>
  <system.codedom>...</system.codedom>
  <!-- PayPal SDK settings -->
  <paypal>
    <settings>
      <add name="mode" value="sandbox"/>
      <add name="connectionTimeout" value="360000"/>
      <add name="requestRetries" value="1"/>
      <add name="clientId"
        value="AS_zpr0T5vVRMzRNxXkiI6TUMXDTzC_KF8Rm2tZGokWVWWFZsQ67ikyv85wkr8FRp
        r"/>
      <add name="clientSecret" value="EAqWbRkqKRjQgyLDtk3Lh1dNlxXM79Wqd-
        KuVMcLMUeWzEm1B1Rdtb7B4otTdZ1K1hIZaE9-jaFenfQS"/>
    </settings>
  </paypal>
</configuration>

```

Sandbox chỉ là môi trường kiểm thử, trước khi triển khai ứng dụng thanh toán trên paypal cần đổi mode "live" và thay đổi clientId và clientSecret key

- Code

```

<paypal>
  <settings>
    <add name="mode" value="sandbox" />
    <add name="connectionTimeout" value="360000" />
    <add name="requestRetries" value="1" />
    <add name="clientId" value="Giá trị clientId" />
    <add name="clientSecret" value="Giá trị Secret" />
  </settings>
</paypal>

```

1.4 Tạo controller thực hiện thanh toán qua **Paypal** đặt tên **PaypalController.cs**

```

namespace WebMVCPaypal.Controllers
{
    public class PaypalController : Controller
    {
        private PayPal.Api.Payment payment;
        // GET: Paypal
        public ActionResult Index()
        {
            return View();
        }

        public ActionResult PaymentWithPaypal()...

        private Payment CreatePayment(APIContext apiContext, string returnUrl)...

        private Payment ExecutePayment(APIContext apiContext, string payerId, string
            paymentId)...
    }
}

```

```

public ActionResult PaymentWithPaypal()
{
    //getting the apiContext as earlier
    APIContext apiContext = Configuration.GetAPIContext();

    try
    {
        string payerId = Request.Params["PayerID"];

        if (string.IsNullOrEmpty(payerId))
        {
            //this section will be executed first because PayerID doesn't exist
            //it is returned by the create function call of the payment class

            // Creating a payment
            // baseUrl is the url on which paypal sendsback the data.
            // So we have provided URL of this controller only
            string baseURI = Request.Url.Scheme + "://" + Request.Url.Authority
+
                "/Paypal/PaymentWithPayPal?";

            //guid we are generating for storing the paymentID received in
session
            //after calling the create function and it is used in the payment
execution

            var guid = Convert.ToString((new Random()).Next(100000));

            //CreatePayment function gives us the payment approval url
            //on which payer is redirected for paypal account payment

```

```
var createdPayment = this.CreatePayment(apiContext, baseURI +
"guid=" + guid);

//get links returned from paypal in response to Create function
call

var links = createdPayment.links.GetEnumerator();

string paypalRedirectUrl = null;

while (links.MoveNext())
{
    Links lnk = links.Current;

    if (lnk.rel.ToLower().Trim().Equals("approval_url"))
    {
        //saving the paypalredirect URL to which user will be
redirected for payment
        paypalRedirectUrl = lnk.href;
    }
}

// saving the paymentID in the key guid
Session.Add(guid, createdPayment.id);

return Redirect(paypalRedirectUrl);
}
else
{
    // This section is executed when we have received all the payments
parameters

    // from the previous call to the function Create

    // Executing a payment

    var guid = Request.Params["guid"];

    var executedPayment = ExecutePayment(apiContext, payerId,
Session[guid] as string);

    if (executedPayment.state.ToLower() != "approved")
    {
        return View("FailureView");
    }
}
}
catch (Exception ex)
{
    Logger.Log("Error" + ex.Message);
    return View("FailureView");
}

return View("SuccessView");
}
```

```
private Payment CreatePayment(APIContext apiContext, string returnUrl)
{
    var itemList = new ItemList() { items = new List<Item>() };
    //Các giá trị bao gồm danh sách sản phẩm, thông tin đơn hàng
    //Sẽ được thay đổi bằng hành vi thao tác mua hàng trên website

    itemList.items.Add(new Item()
    {
        //Thông tin đơn hàng
        name = "Item Name",
        currency = "USD",
        price = "5",
        quantity = "1",
        sku = "sku"
    });
    //Hình thức thanh toán qua paypal
    var payer = new Payer() { payment_method = "paypal" };

    // Configure Redirect Urls here with RedirectUrls object
    var redirUrls = new RedirectUrls()
    {
        cancel_url = returnUrl,
        return_url = returnUrl
    };
    //các thông tin trong đơn hàng
    var details = new Details()
    {
        tax = "1",
        shipping = "1",
        subtotal = "5"
    };
    //Đơn vị tiền tệ và tổng đơn hàng cần thanh toán
    var amount = new Amount()
    {
        currency = "USD",
        total = "7", // Total must be equal to sum of shipping, tax and
subtotal.
        details = details
    };

    var transactionList = new List<Transaction>();
    //Tất cả thông tin thanh toán cần đưa vào transaction
    transactionList.Add(new Transaction()
    {
        description = "Transaction description.",
        invoice_number = "your invoice number",
        amount = amount,
        item_list = itemList
    });

    this.payment = new Payment()
    {
        intent = "sale",
        payer = payer,
        transactions = transactionList,
        redirect_urls = redirUrls
    };
}
```

```
        // Create a payment using a APIContext
        return this.payment.Create(apiContext);
    }
```

```
private Payment ExecutePayment(APIContext apiContext, string payerId, string paymentId)
{
    var paymentExecution = new PaymentExecution() { payer_id = payerId };
    this.payment = new Payment() { id = paymentId };
    return this.payment.Execute(apiContext, paymentExecution);
}
```

1.5 Tạo thư mục /Helper, thêm 2 file [Configuration.cs](#) và [Logger.cs](#)

- Configuration.cs

```
public static class Configuration
{
    public readonly static string ClientId;
    public readonly static string ClientSecret;

    // Static constructor for setting the readonly static members.
    static Configuration()
    {
        var config = GetConfig();
        ClientId = config["clientId"];
        ClientSecret = config["clientSecret"];
    }

    // Create the configuration map that contains mode and other optional
    // configuration details.
    public static Dictionary<string, string> GetConfig()
    {
        return PayPal.Api.ConfigManager.Instance.GetProperties();
    }

    // Create accessToken
    private static string GetAccessToken()
    {
        // ###AccessToken
        // Retrieve the access token from
        // OAuthTokenCredential by passing in
        // ClientID and ClientSecret
        // It is not mandatory to generate Access Token on a per call basis.
        // Typically the access token can be generated once and
        // reused within the expiry window
        string accessToken = new OAuthTokenCredential(ClientId, ClientSecret,
            GetConfig()).GetAccessToken();
        return accessToken;
    }

    // Returns APIContext object
    public static APIContext GetAPIContext()
    {

```

```
// ### Api Context
// Pass in a `APIContext` object to authenticate
// the call and to send a unique request id
// (that ensures idempotency). The SDK generates
// a request id if you do not pass one explicitly.
APIContext apiContext = new APIContext(GetAccessToken());
apiContext.Config = GetConfig();

// Use this variant if you want to pass in a request id
// that is meaningful in your application, ideally
// a order id.
// String requestId = Long.toString(System.nanoTime());
// APIContext apiContext = new APIContext(GetAccessToken(), requestId );

return apiContext;
}
}
```

- Logger.cs

```
public class Logger
{
    public static string LogDirectoryPath = Environment.CurrentDirectory;

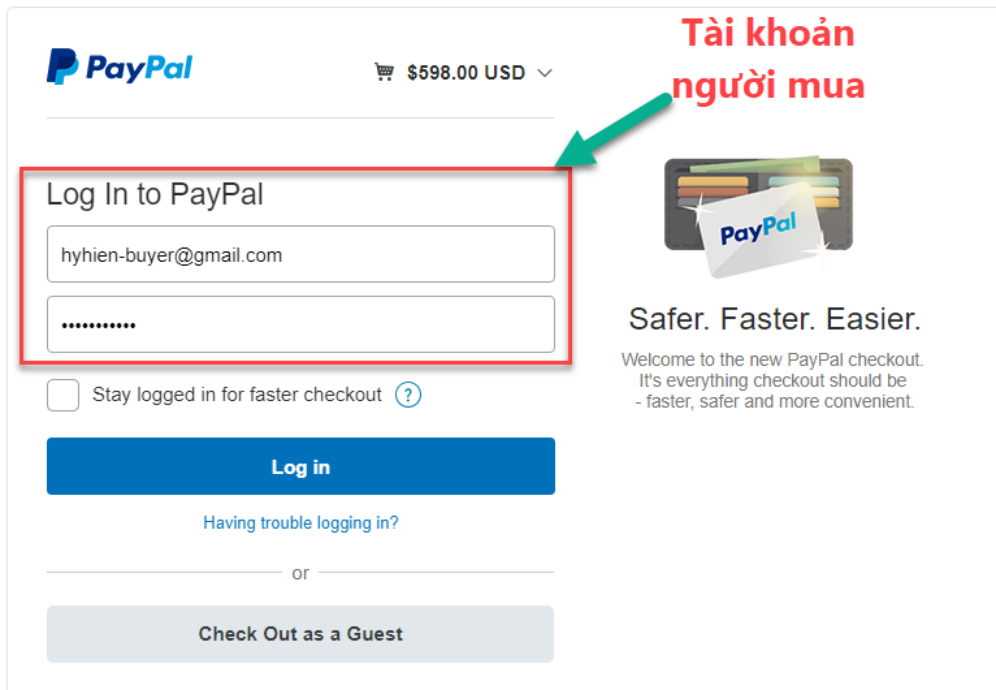
    public static void Log(String lines)
    {
        // Write the string to a file.append mode is enabled so that the log
        // lines get appended to test.txt than wiping content and writing the log

        try
        {
            System.IO.StreamWriter file = new
System.IO.StreamWriter(LogDirectoryPath + "\\Error.log", true);
            file.WriteLine(DateTime.Now.ToString("dd/MM/yyyy HH:mm:ss") + " --> " +
lines);
            file.Close();
        }
        catch
        {
        }
    }
}
```

1.6 Chạy thử ứng dụng tích hợp

- [Http:// localhost:60099/paypal/PaymentWithPaypal](http://localhost:60099/paypal/PaymentWithPaypal)

test facilitator's Test Store



Tài khoản người mua

Log In to PayPal

hyhien-buyer@gmail.com

.....

☐ Stay logged in for faster checkout [?](#)

Log in

[Having trouble logging in?](#)

or

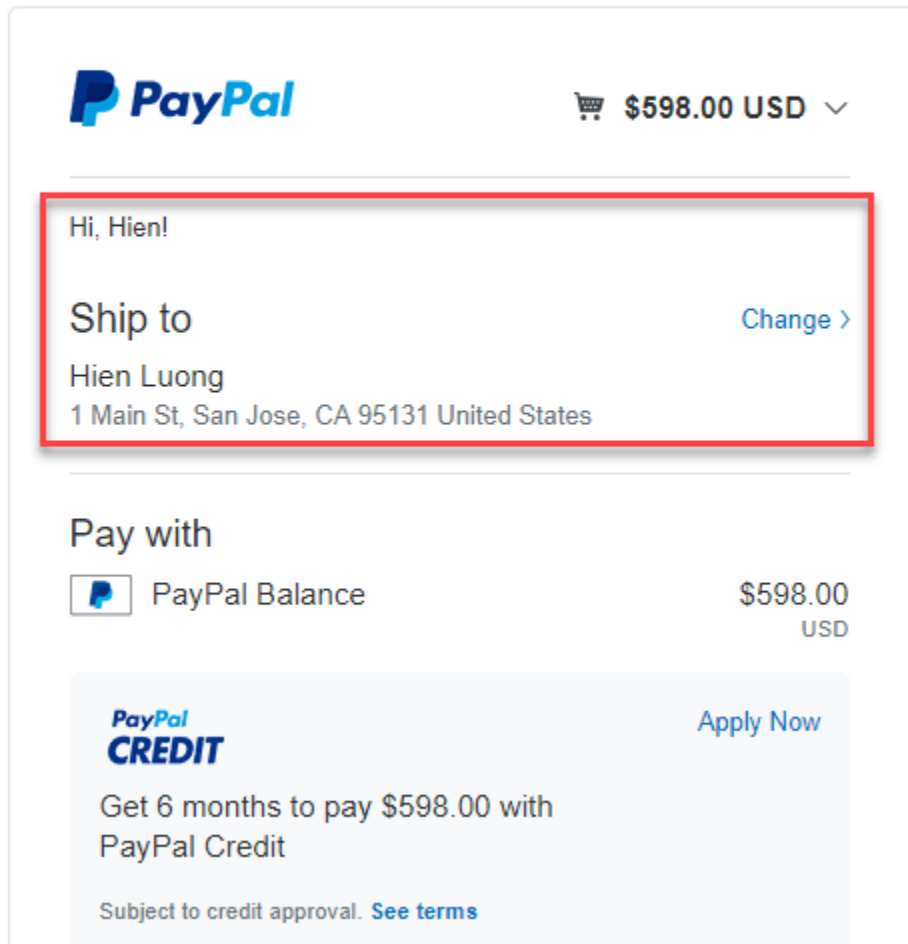
Check Out as a Guest

Safer. Faster. Easier.

Welcome to the new PayPal checkout.
It's everything checkout should be
- faster, safer and more convenient.

- Xử lý

test facilitator's Test Store



PayPal

Shopping cart icon: \$598.00 USD

Hi, Hien!

Ship to [Change >](#)

Hien Luong
1 Main St, San Jose, CA 95131 United States

Pay with

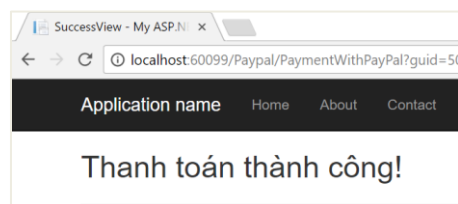
PayPal Balance \$598.00 USD

PayPal CREDIT [Apply Now](#)

Get 6 months to pay \$598.00 with PayPal Credit

Subject to credit approval. [See terms](#)

- Kết quả



- Đăng nhập vào <https://www.sandbox.paypal.com/> để kiểm tra đơn hàng
Sử dụng tài khoản người bán + người mua để kiểm tra:



hyhien-facilitator@gmail.com

.....

Log In

Having trouble logging in?

or

Sign Up

- Xem thống kê đơn hàng

PayPal

Summary

Money

Activity

Reports

Tools

More

Activity

Activity
(including balance & fees)

Charts & Graphs

Email Address

Search

Active

All transactions

All currencies

Past 30 days


	Date	Type	Name	Payment	Gross	Fee	Net	Balance
	5:59 AM	Payment from	test buyer	Completed	\$7.00 USD	-\$0.50	\$6.50	\$13.00 USD
	1:04 AM	Payment from	test buyer	Completed	\$7.00 USD	-\$0.50	\$6.50	\$6.50 USD

- Sau khi Kiểm thử thành công trong môi trường Sandbox, chuyển đổi thành môi trường thật để thanh toán paypal (sinh viên tự thực hiện theo mô tả ở phần trên)

1.7 Bổ sung: Sinh viên hiện thực các màn hình

- Thiết kế màn hình xem giỏ hàng, người dùng chọn hình thức thanh toán qua paypal

[← Mua thêm sản phẩm khác](#)Giỏ hàng của bạn



Apple Macbook Air MMEF2ZP/A i5 5250U/8GB/128GB
• Tặng Balo thegioididong (Nếu không có quà sẽ hoàn 100,000đ)
• Trả góp 0% với HSBC HOẶC Tặng PMH 500,000đ

Xóa

Màu: Bạc

-

2

+

Tổng tiền: **47.980.000đ**
[Sử dụng mã giảm giá](#)

☒ Anh ☐ Chị

Nguyễn Đình Ánh

phone

Số điện thoại ít nhất là 10 số

Để được phục vụ nhanh hơn, hãy chọn thêm:
☐ Địa chỉ giao hàng ☐ Nhận tại siêu thị

THANH TOÁN KHI NHẬN HÀNG
Xem hàng trước, không mua không sao

THANH TOÁN ONLINE
Bằng thẻ ATM, Visa, Master

- Hệ thống nhận các giá trị của sản phẩm từ CSDL và thông tin khách hàng để thanh toán online qua paypal như hướng dẫn tại các phần trước.