



SEEM4540

Open Systems for
E-Commerce

Lecture 07 - Online Payment

PayPal

- PayPal is an American based e-commerce business allowing payments and money transfers to be made through the Internet.
- In 1998, Confinity (by Max Levchin, Peter Thiel, Luke Nosek, and Ken Howery) established. PayPal was a service name for money transfer service.
- In 1999, X.com (an online bank) established.
- In 2000, X.com merged with Confinity and later renamed as PayPal.
- In 2002, Paypal was acquired by eBay.
- In 2013, PayPal moved \$180 billion in 26 currencies across 193 nations, generating a total revenue of \$6.6 billion (41% of eBay's total profits).

PayPal for Developer

- PayPal provides four ways for you to interact with its database:
 - REST
 - Need some programming (this lecture)
 - Predefined Buttons
 - Simply add some PayPal buttons on your web page. No programming involved (next lecture)
 - Mobile SDK
 - For iOS (Objective C) and Android (Java) programmers
 - Classical APIs
 - Make use of non-RESTful interfaces to provide payment solutions. This is the traditional way to connect to PayPal.

PayPal Programming

- For PayPal (programming), please register two accounts:
 - PayPal account
 - Used for creating our own application (app).
 - <https://www.paypal.com/>
 - PayPal sandbox account
 - Used for testing. E.g. you do not want to use your own credit card to complete a real transaction for testing purpose. That's why we need it.
 - <https://www.sandbox.paypal.com/>
- Documentation:
 - <https://developer.paypal.com/docs/>

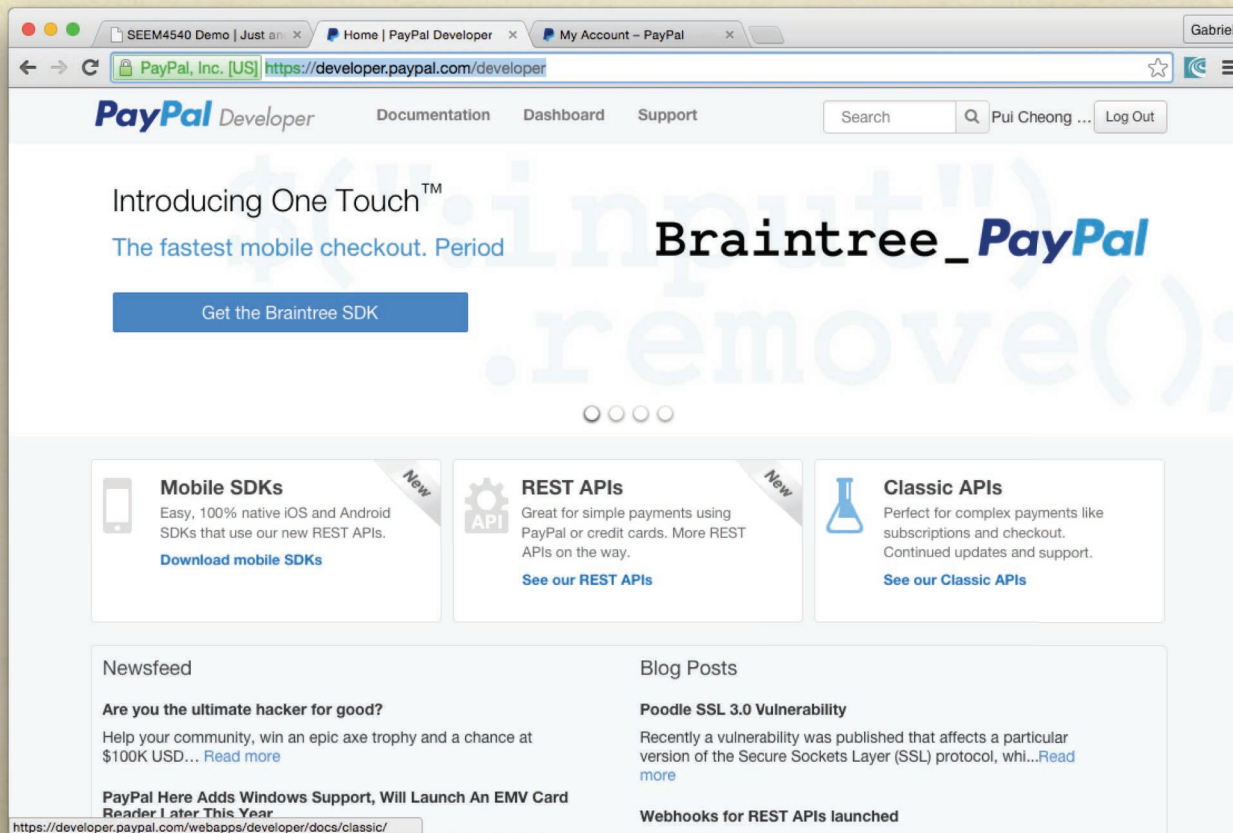
Sandbox

- In computer security, a sandbox is a security mechanism for separating running programs. It is often used to execute untested code, or untrusted programs from unverified third parties, suppliers, untrusted users and untrusted websites.



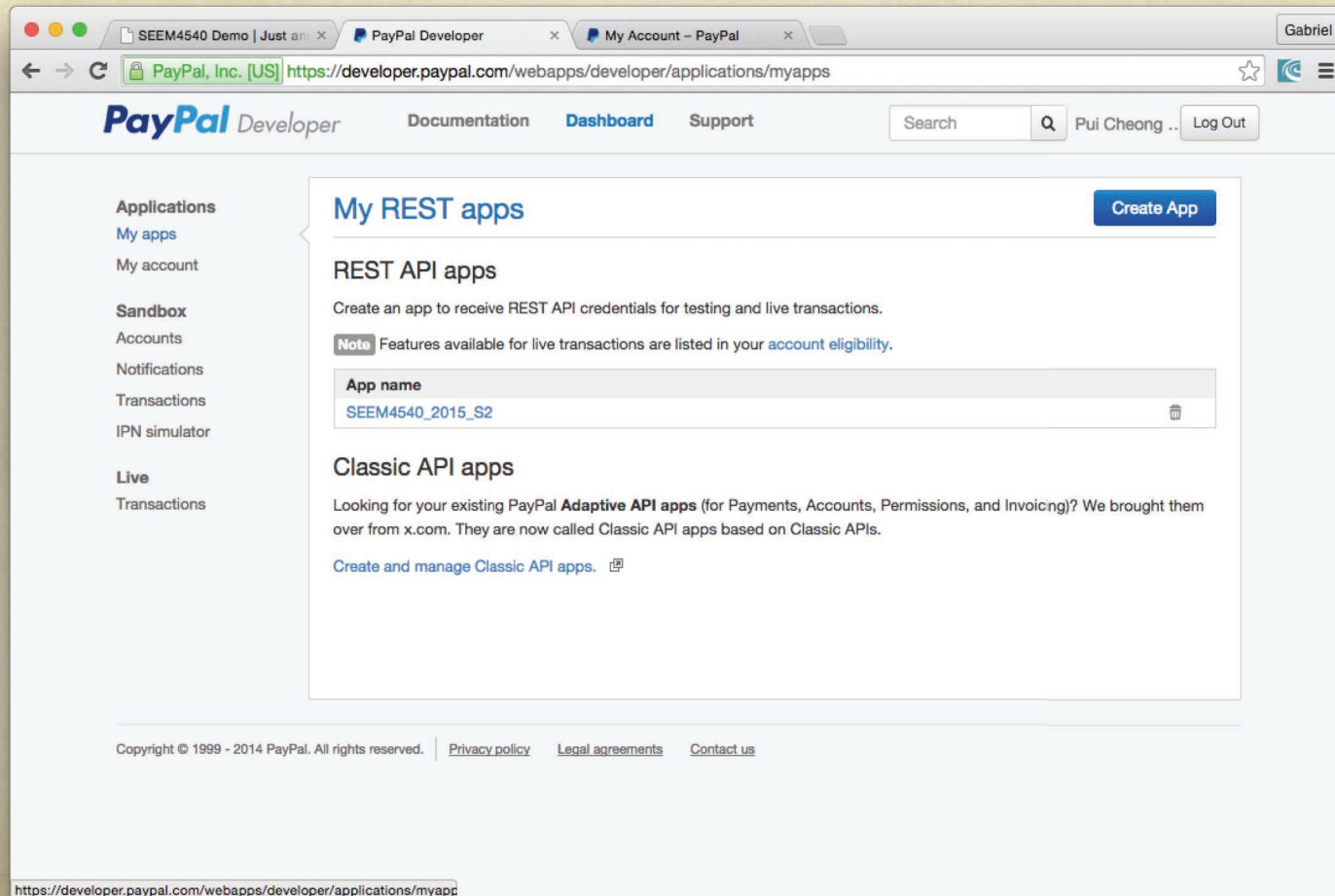
Developer Tools

- To create app, we need to go to the developer side:
- <https://developer.paypal.com/developer>



Developer Interface

○ The dashboard (classic UI)



The screenshot displays the PayPal Developer Dashboard in a browser window. The browser tabs include 'SEEM4540 Demo | Just an...', 'PayPal Developer', and 'My Account - PayPal'. The address bar shows 'https://developer.paypal.com/webapps/developer/applications/myapps'. The page header features the 'PayPal Developer' logo, navigation links for 'Documentation', 'Dashboard', and 'Support', a search bar, and a user profile for 'Pui Cheong ..' with a 'Log Out' button. The main content area is divided into a left sidebar and a main panel. The sidebar has sections for 'Applications' (with links for 'My apps' and 'My account'), 'Sandbox' (with links for 'Accounts', 'Notifications', 'Transactions', and 'IPN simulator'), and 'Live' (with a link for 'Transactions'). The main panel is titled 'My REST apps' and includes a 'Create App' button. Below this, there is a section for 'REST API apps' with a description and a note about live transaction eligibility. A table lists one application with the name 'SEEM4540_2015_S2'. Below the table is a section for 'Classic API apps' with a description and a link to 'Create and manage Classic API apps'. The footer contains copyright information and links for 'Privacy policy', 'Legal agreements', and 'Contact us'.

SEEM4540 Demo | Just an... PayPal Developer My Account - PayPal Gabriel

PayPal, Inc. [US] https://developer.paypal.com/webapps/developer/applications/myapps

PayPal Developer Documentation Dashboard Support Search Pui Cheong .. Log Out

Applications
My apps
My account

Sandbox
Accounts
Notifications
Transactions
IPN simulator

Live
Transactions

My REST apps

Create App

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](#).

App name
SEEM4540_2015_S2

Classic API apps

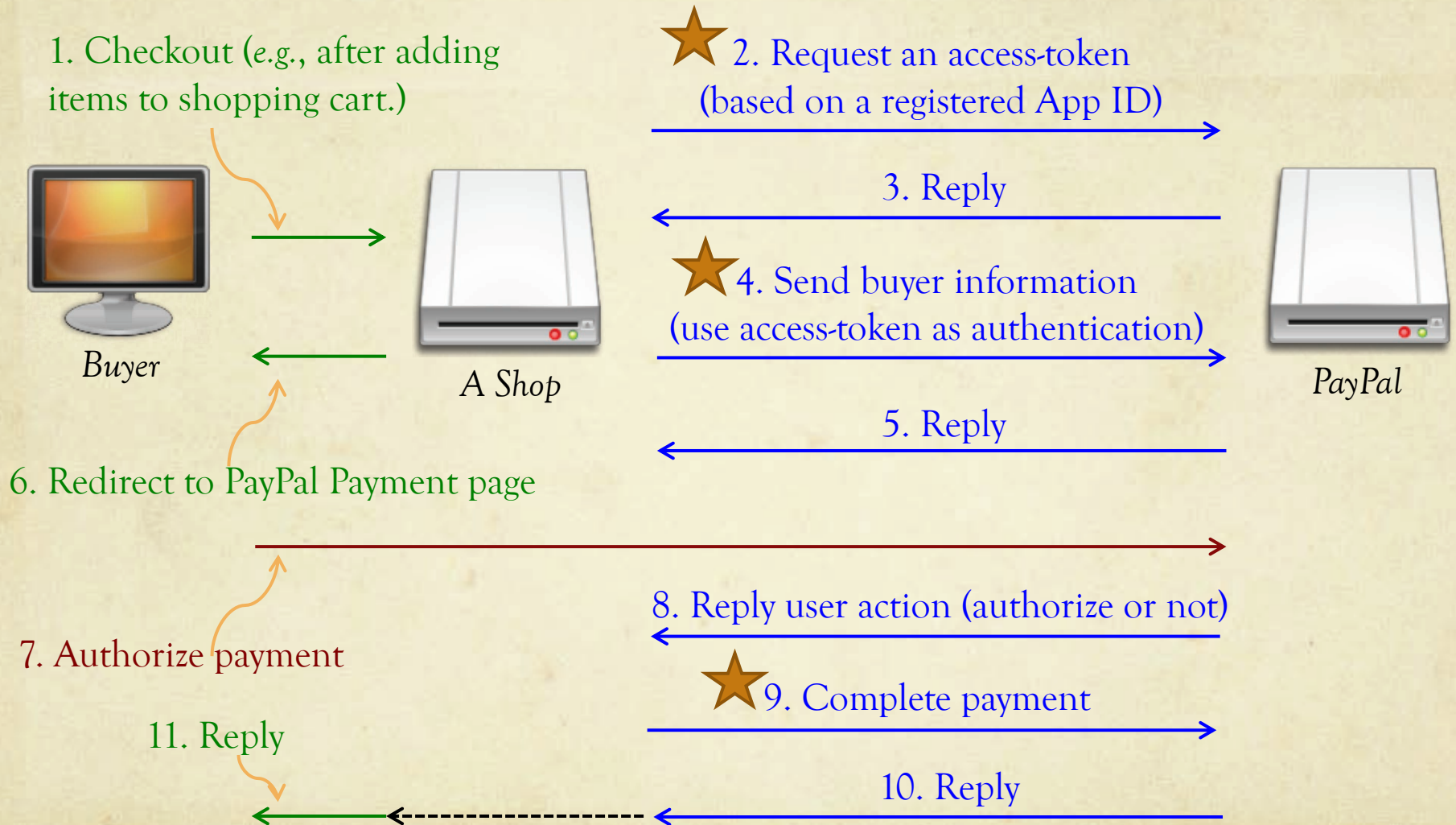
Looking for your existing PayPal **Adaptive API apps** (for Payments, Accounts, Permissions, and Invoicing)? We brought them over from x.com. They are now called Classic API apps based on Classic APIs.

[Create and manage Classic API apps.](#)

Copyright © 1999 - 2014 PayPal. All rights reserved. | [Privacy policy](#) [Legal agreements](#) [Contact us](#)

https://developer.paypal.com/webapps/developer/applications/myapp

PayPal with REST API Overview



PayPal with REST API

○ General steps:

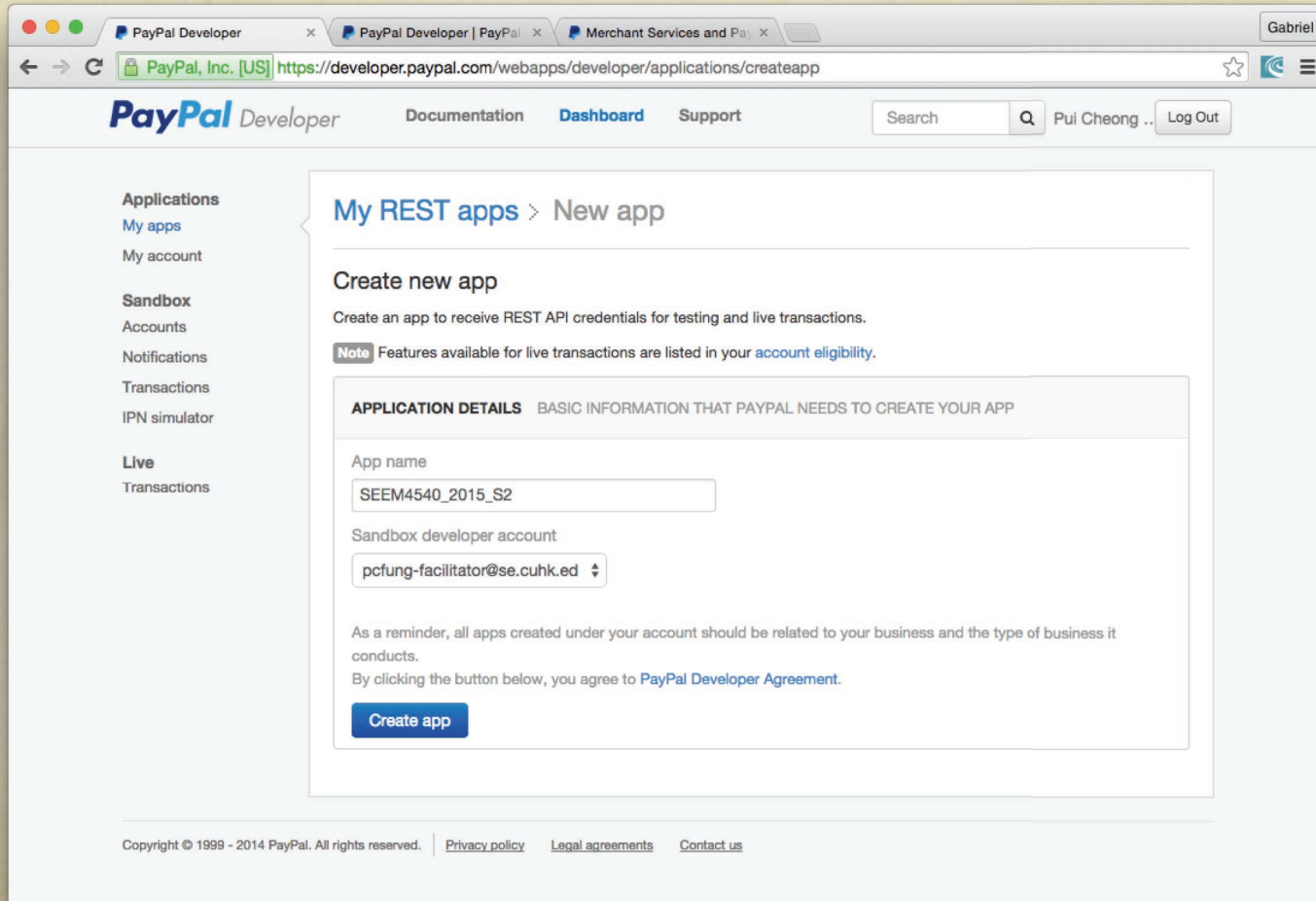
1. Create a PayPal application (a.k.a. PayPal app) from the online PayPal developer interface.
2. In your program, connect to your PayPal application and obtain an access token dynamically.
3. In your program, make API calls based on the dynamically generated access token.

Create a PayPal App

The screenshot shows the PayPal Developer dashboard. The browser address bar displays the URL <https://developer.paypal.com/apps/develop/applications/myapps>. The page header includes the PayPal Developer logo, navigation links for Documentation, Dashboard, and Support, a search bar, and a user profile for 'Pui Cheong' with a Log Out button. The left sidebar contains a menu with 'Applications' (circled in red), 'My apps', 'My account', 'Sandbox', 'Accounts', 'Notifications', 'Transactions', 'IPN simulator', 'Live', and 'Transactions'. The main content area is titled 'My REST apps' and features a 'Create App' button (circled in red). Below this, there is a section for 'REST API apps' with a description and a 'Note' about live transaction features. A form field for 'App name' is present with the placeholder text 'Create your first app to view it here'. Further down, there is a section for 'Classic API apps' with a description and a link to 'Create and manage Classic API apps'. The footer contains copyright information and links for Privacy policy, Legal agreements, and Contact us.

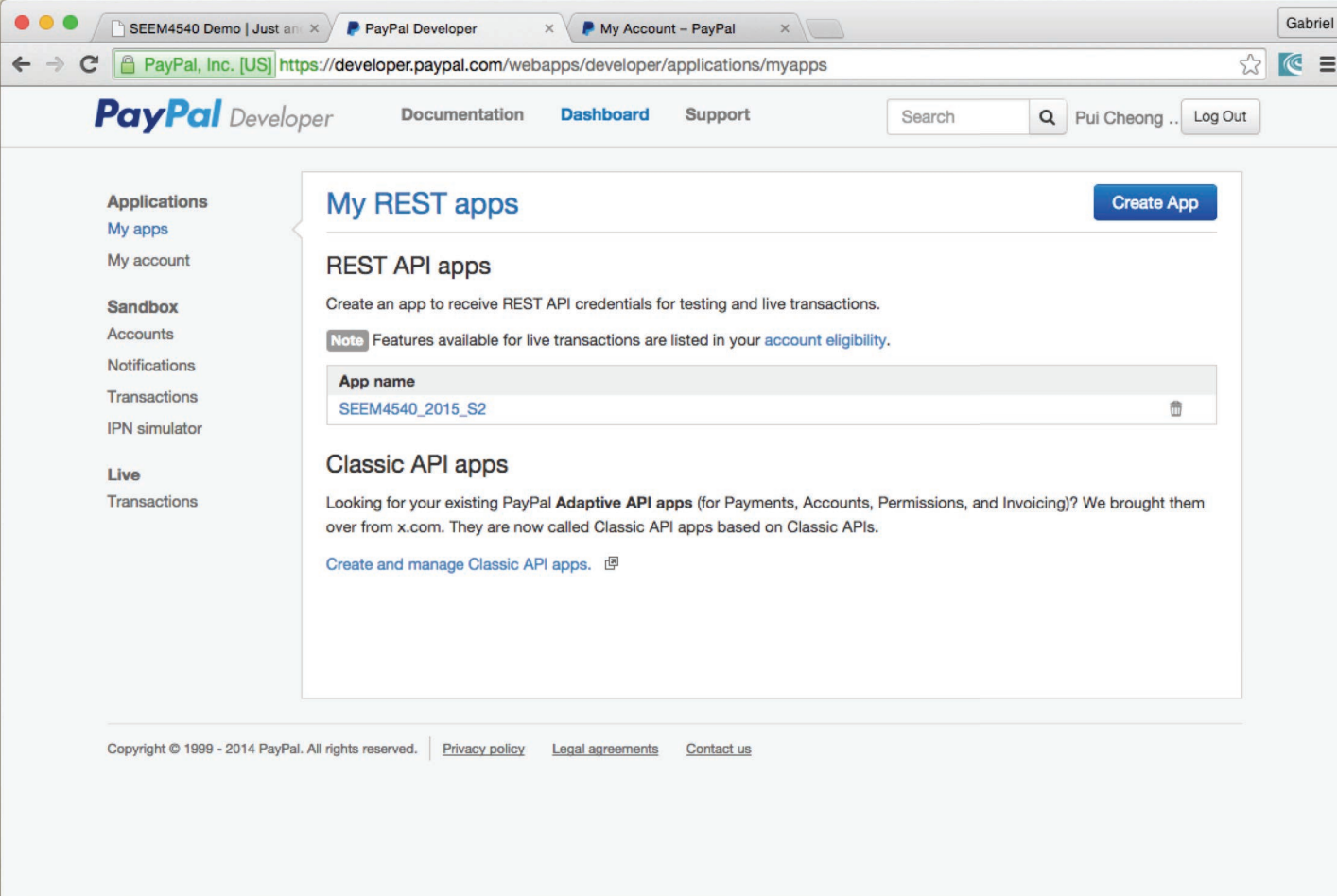
Create a PayPal App (cont'd)

- A sample screen shot



Create a PayPal App (cont'd)

○ A sample screen shot



The screenshot shows a web browser window with three tabs: "SEEM4540 Demo | Just an...", "PayPal Developer", and "My Account - PayPal". The address bar shows the URL "https://developer.paypal.com/webapps/developer/applications/myapps". The page header includes the "PayPal Developer" logo, navigation links for "Documentation", "Dashboard", and "Support", a search bar, and a user profile for "Pui Cheong .." with a "Log Out" button. The main content area is titled "My REST apps" and features a "Create App" button. Below this, there is a section for "REST API apps" with a description: "Create an app to receive REST API credentials for testing and live transactions." A note states: "Features available for live transactions are listed in your account eligibility." A table lists one app with the name "SEEM4540_2015_S2". Below the table is a section for "Classic API apps" with a description: "Looking for your existing PayPal Adaptive API apps (for Payments, Accounts, Permissions, and Invoicing)? We brought them over from x.com. They are now called Classic API apps based on Classic APIs." A link "Create and manage Classic API apps." is provided. The footer contains copyright information: "Copyright © 1999 - 2014 PayPal. All rights reserved." and links for "Privacy policy", "Legal agreements", and "Contact us".

Create a PayPal App (cont'd)

- A sample screen shot

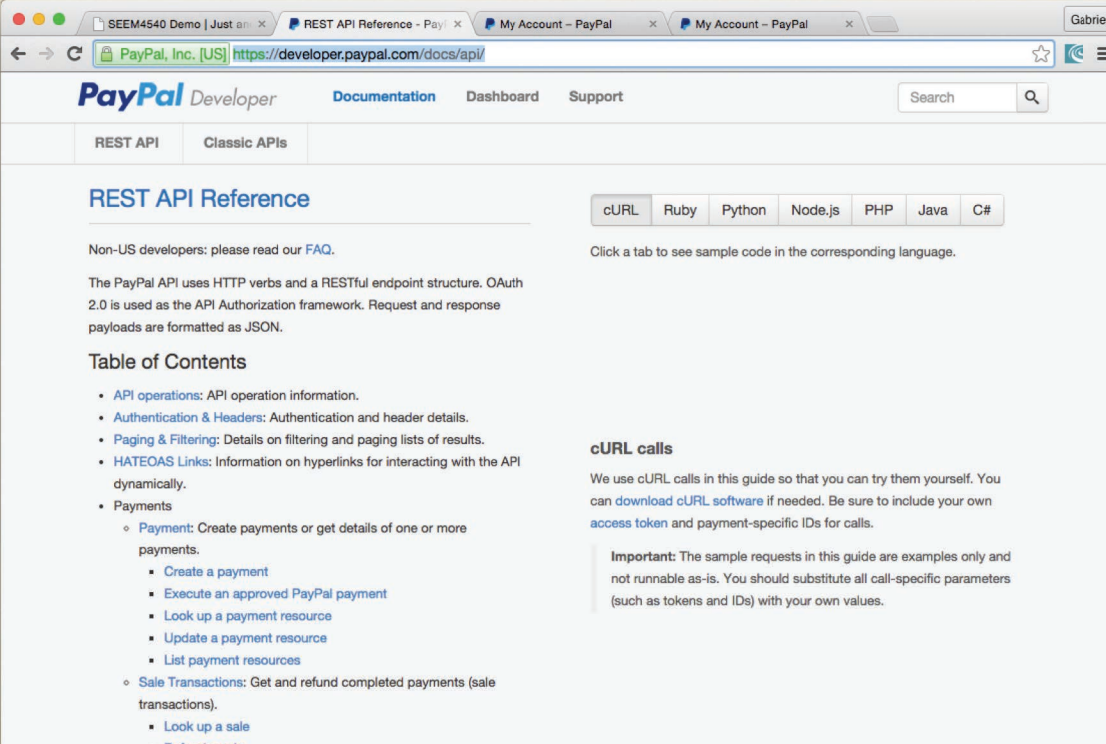
The screenshot shows the PayPal Developer dashboard for a user named Gabriel. The main content area displays the details for a REST app named "SEEM4540_2015_S2". A green notification box at the top states: "App successfully created. Use test credentials to test your application in Sandbox environment." Below this, the app name "SEEM4540_2015_S2" is shown with a folder icon. The "APP DETAILS" section includes the following information:

- App name: SEEM4540_2015_S2
- App display name: SEEM4540_2015_S2
- App redirect URLs: Edit
- Redirect URLs are required for Future Payment and Log In with PayPal features.
- Sandbox credentials: Your sandbox credentials are enabled for all features in your app.
- Sandbox account: pcfung-facilitator@se.cuhk.edu.hk
- Endpoint: [Redacted]
- Client ID: Ad80-khNk1ZyB-UiBVVth-Xmdri7I_hV_ntluu9_MfgJOx1lg...C57
- Secret: EJo2HiGeS8CeHSwLaN0YILyQnYOblisHXowMFSpgal17A...ZjDyVgKLGh7VQ1Py7m7IBB5TcsMEI

The Client ID and Secret fields are circled in red. A brown callout box with white text points to these fields, stating: "Client ID and Secret are used to obtain your access token".

Get an Access Token (cont'd)

- API:
 - <https://developer.paypal.com/docs/api/>
 - The following screen shows cURL implementation.



The screenshot shows a web browser window displaying the PayPal Developer REST API Reference page. The browser's address bar shows the URL <https://developer.paypal.com/docs/api/>. The page header includes the PayPal Developer logo and navigation links for Documentation, Dashboard, and Support. Below the header, there are tabs for REST API and Classic APIs. The main content area is titled "REST API Reference" and features a "Table of Contents" with links to various API operations. On the right side, there are tabs for different programming languages, with "cURL" selected. Below the tabs, there is a section titled "cURL calls" with an "Important" note.

REST API Reference

Non-US developers: please read our [FAQ](#).

The PayPal API uses HTTP verbs and a RESTful endpoint structure. OAuth 2.0 is used as the API Authorization framework. Request and response payloads are formatted as JSON.

Table of Contents

- [API operations](#): API operation information.
- [Authentication & Headers](#): Authentication and header details.
- [Paging & Filtering](#): Details on filtering and paging lists of results.
- [HATEOAS Links](#): Information on hyperlinks for interacting with the API dynamically.
- [Payments](#)
 - [Payment](#): Create payments or get details of one or more payments.
 - [Create a payment](#)
 - [Execute an approved PayPal payment](#)
 - [Look up a payment resource](#)
 - [Update a payment resource](#)
 - [List payment resources](#)
 - [Sale Transactions](#): Get and refund completed payments (sale transactions).
 - [Look up a sale](#)
 - [Refund a sale](#)

cURL calls

We use cURL calls in this guide so that you can try them yourself. You can [download cURL software](#) if needed. Be sure to include your own [access token](#) and payment-specific IDs for calls.

Important: The sample requests in this guide are examples only and not runnable as-is. You should substitute all call-specific parameters (such as tokens and IDs) with your own values.

cURL

- A software project providing a library and a command-line tool for transferring data using various protocols.
- Some people pronounce it as “see-U-R-L”, whereas some pronounce it as “crew”.
- A lot of related tools and framework, such as libcurl.

PayPal PHP SDK

- PayPal provides a full-featured SDK for PHP Developers!
- PayPal PHP SDK is the official Open Source PHP SDK for supporting PayPal Rest APIs. Checkout all the supporting documents, samples, codebase from:
 - <http://paypal.github.io/PayPal-PHP-SDK/>

// PayPal PHP SDK 

PayPal PHP SDK Installation

- Follow the instruction here:
 - <https://github.com/paypal/PayPal-PHP-SDK/wiki/Installation-Direct-Download>

PayPal PHP SDK – Your First Call

- To make sure you have installed everything correctly, please write a sample PHP file with the following content:
- Note: it is based on the instructions in:
 - <https://github.com/paypal/PayPal-PHP-SDK/wiki/Making-First-Call>

```
<?php
```

```
require __DIR__ . '/PayPal-PHP-SDK/autoload.php';
```

```
$clientId = "xxxxxxxx";
```

```
$secret = "xxxxxxxx";
```

```
$apiContext = new \PayPal\Rest\ApiContext(
```

```
    new \PayPal\Auth\OAuthTokenCredential($clientId, $secret)
```

```
);
```

PayPal PHP SDK – Your First Call (cont'd)

```
$creditCard = new \PayPal\Api\CreditCard();
$creditCard->setType("visa")
    ->setNumber("4417119669820331")
    ->setExpireMonth("11")
    ->setExpireYear("2019")
    ->setCvv2("012")
    ->setFirstName("Joe")
    ->setLastName("Shopper");
try {
    $creditCard->create($apiContext);
    echo "success";
}
catch (\PayPal\Exception\PayPalConnectionException $ex) {
    echo $ex->getData();
}
?>
```

PayPal PHP SDK – Your First Call (cont'd)

- If you install everything correct, you should get the word “success” on the screen.

Make API Calls

- With a valid access token, you're ready to make a request to web services via REST API interface.
- In the following, we will demonstrate how to make a payment transaction.
 - We need to create three pages:
 1. Process Payment page
 2. Confirm Payment page
 3. Cancel Payment page

Process Payment – payment_process.php (1)

```
<?php
```

```
require __DIR__.'\PayPal-PHP-SDK/autoload.php'; // include the SDK
```

```
use PayPal\Api\Amount; // use the PayPal resources
```

```
use PayPal\Api\Details;
```

```
use PayPal\Api\Item;
```

```
use PayPal\Api\ItemList;
```

```
use PayPal\Api\Payer;
```

```
use PayPal\Api\Payment;
```

```
use PayPal\Api\RedirectUrls;
```

```
use PayPal\Api\Transaction;
```

```
$clientId = "xxxxxxx"; // set your client ID and Secret
```

```
$secret = "xxxxxxxxx";
```

```
$apiContext = new \PayPal\Rest\ApiContext(
```

```
    new \PayPal\Auth\OAuthTokenCredential($clientId, $secret) // get the access token
```

```
);
```

Process Payment – payment_process.php (2)

```
$payer = new Payer(); // create a payer
```

```
$payer->setPaymentMethod("paypal");
```

```
$item1 = new Item(); // your items. Note: they should be get dynamically!
```

```
$item1->setName('Coffee Bean'); // for demonstration, we hard-code them
```

```
$item1->setCurrency('HKD');
```

```
$item1->setQuantity(1);
```

```
$item1->setPrice(75);
```

```
$item2 = new Item();
```

```
$item2->setName('Chinese Tea');
```

```
$item2->setCurrency('HKD');
```

```
$item2->setQuantity(3);
```

```
$item2->setPrice(200);
```

Process Payment – payment_process.php (3)

```
$itemList = new ItemList(); // combine all items into a list
```

```
$itemList->setItems(array($item1, $item2));
```

```
$details = new Details(); // add other details, such as shipping
```

```
$details->setShipping(200);
```

```
$details->setSubtotal(675); // make sure the amount (without shipping) is correct!
```

```
$amount = new Amount();
```

```
$amount->setCurrency("HKD");
```

```
$amount->setTotal(875); // make sure the amount (included shipping) is correct!
```

```
$amount->setDetails($details);
```

```
$transaction = new Transaction();
```

```
$transaction->setAmount($amount);
```

```
$transaction->setItemList($itemList);
```

```
$transaction->setDescription("Detailed description");
```

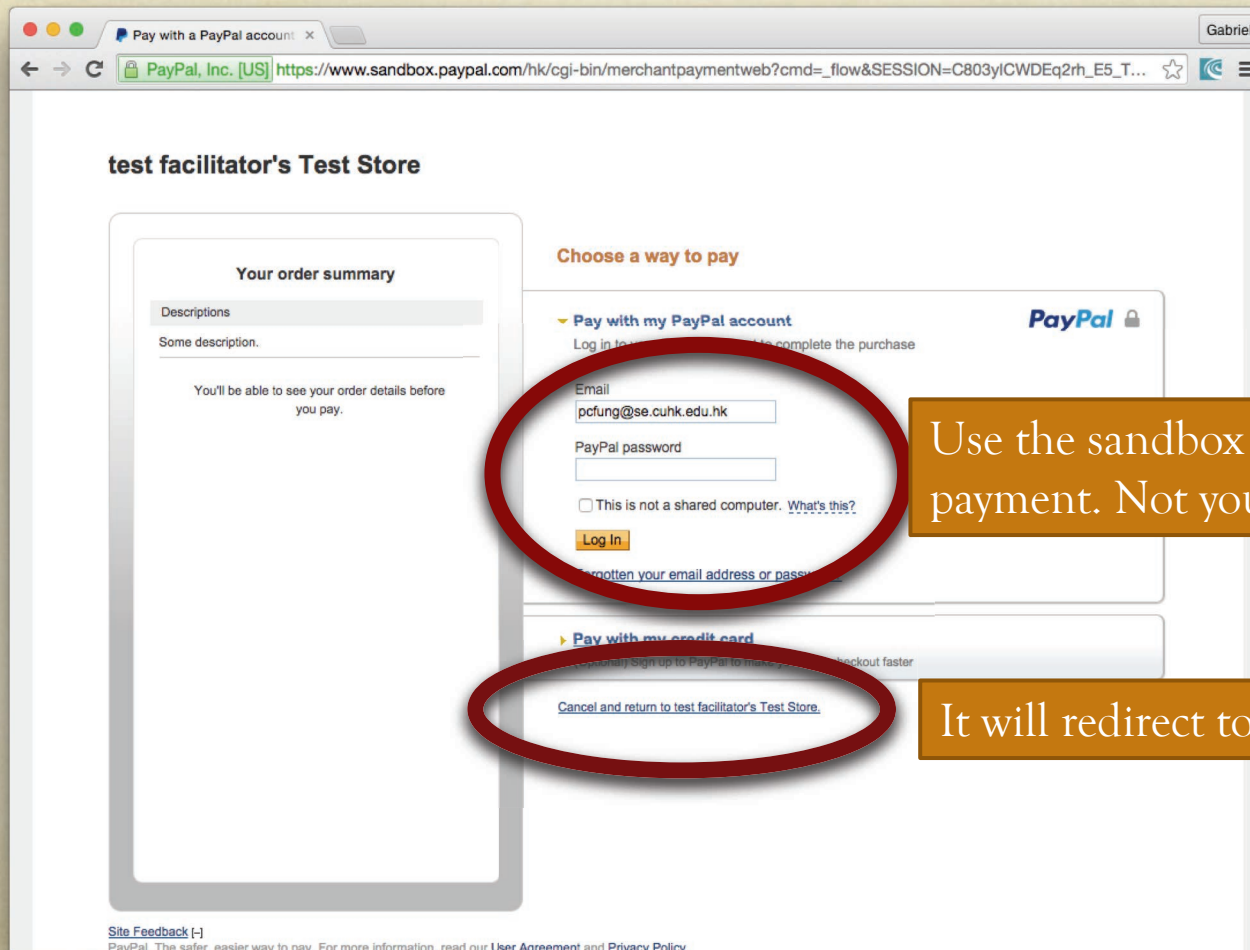

Process Payment – payment_process.php (4)

```
$redirectUrls = new RedirectUrls();
$redirectUrls->setReturnUrl("http://.../payment_confirm.php"); // make sure it exists
$redirectUrls->setCancelUrl("http://.../payment_cancel.php"); // make sure it exists

$payment = new Payment();
$payment->setIntent("sale");
$payment->setPayer($payer);
$payment->setRedirectUrls($redirectUrls);
$payment->setTransactions(array($transaction));
try {
    $payment->create($apiContext);
    $paymentLink = $payment->getApprovalLink(); // the link
    header("location: ".$paymentLink); // redirect to another php/html page
}
catch (Exception $ex) {
    exit($ex);
}
?>
```

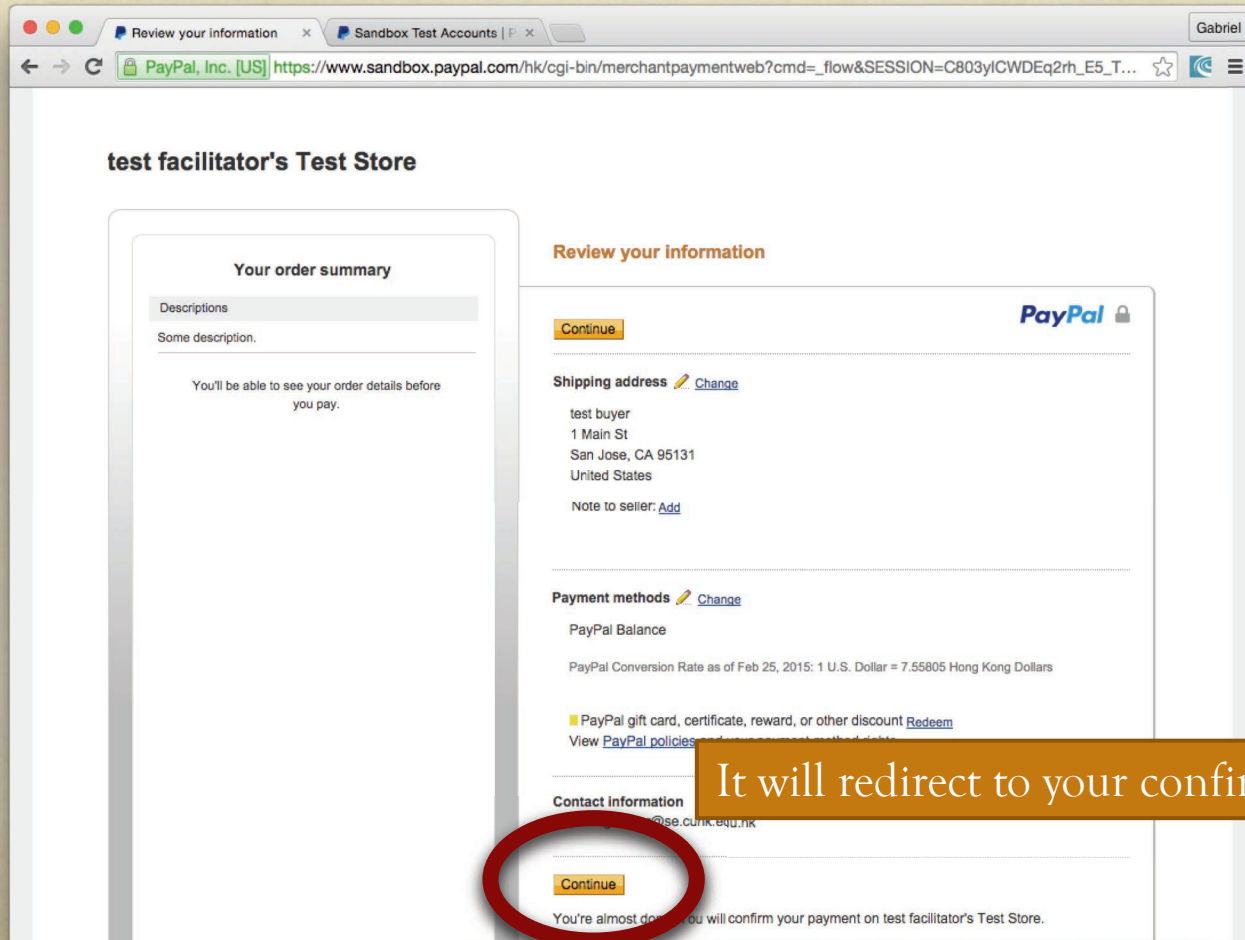
Process Payment – A Sample Output

- A sample output



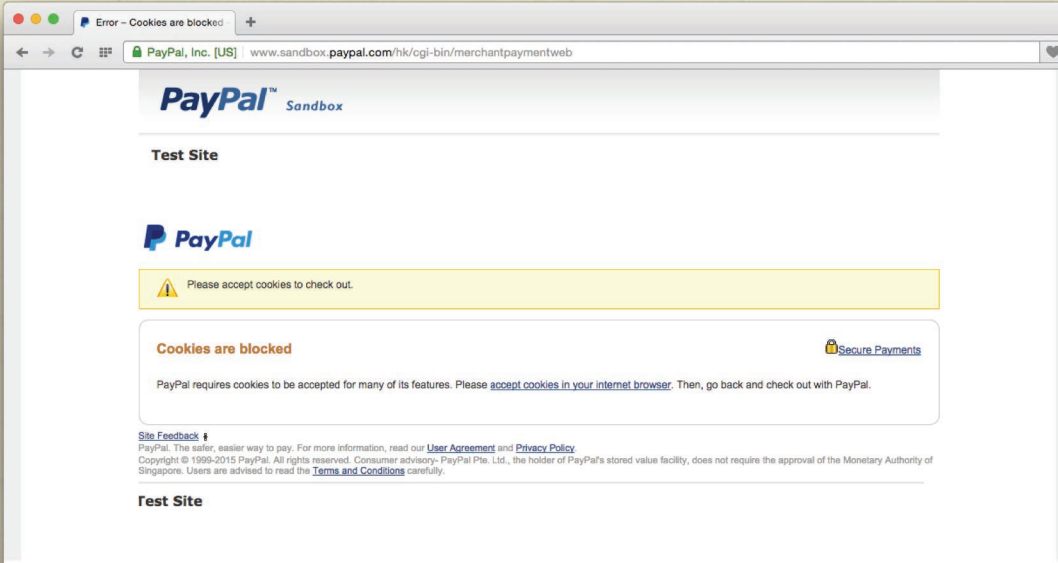
Process Payment – A Sample Output (cont'd)

- A sample output



Process Payment – Funny Issue 1

- When I was preparing the lecture note (in 2015 Semester), I have the following error in the whole night.



PayPal™ Sandbox

Test Site

PayPal

Please accept cookies to check out.

Cookies are blocked [Secure Payments](#)

PayPal requires cookies to be accepted for many of its features. Please [accept cookies in your internet browser](#). Then, go back and check out with PayPal.

[Site Feedback](#)

PayPal. The safer, easier way to pay. For more information, read our [User Agreement](#) and [Privacy Policy](#).
Copyright © 1999-2015 PayPal. All rights reserved. Consumer advisory: PayPal Pte. Ltd., the holder of PayPal's stored value facility, does not require the approval of the Monetary Authority of Singapore. Users are advised to read the [Terms and Conditions](#) carefully.

Test Site

After spending 3 hours for debugging, I give up.
It turns out that it is PayPal sandbox is down...
Not my problem... (News also reported this as well!)



Process Payment - Funny Issue 2

- When I was preparing the lecture note (in 2016 Semester), I have the following error:
 - SSL operation failed with code 1. OpenSSL Error messages:
error:14077410:SSL
routines:SSL23_GET_SERVER_HELLO:sslv3 alert handshake
failure in ...
- After spending an hour, finally, I realized that it is because PayPal updated its underlying SSL model and my computer not yet updated...



Cancel Payment

- To write a cancel payment page is very simple. For example:
 1. Create a file in paypal/payment_cancel.php
 2. Write the following content:

`You have canceled your payment`

Confirm Payment – payment_confirm.php (1)

```
<?php
require __DIR__.'\PayPal-PHP-SDK/autoload.php';

use PayPal\Api\ExecutePayment;
use PayPal\Api\Payment;
use PayPal\Api\PaymentExecution;

$clientId = "xxxxxxx";
$secret = "xxxxxxxxx";
$apiContext = new \PayPal\Rest\ApiContext(
    new \PayPal\Auth\OAuthTokenCredential($clientId, $secret)
);

$paymentId = $_GET['paymentId'];
$payment = Payment::get($paymentId, $apiContext);

$execution = new PaymentExecution();
$execution->setPayerId($_GET['PayerID']);
```

Confirm Payment – payment_confirm.php (2)

```
try{
    $result = $payment->execute($execution, $apiContext);
    $payment = Payment::get($paymentId, $apiContext);
    echo "Thank you for your payment"; // or redirect to another page using header(...)
}
catch (Exception $ex) {
    exit($ex);
}
?>
```


PayPal Payment

- Try to combine everything together by yourself!



Shopping Cart

- Now the only thing we left is to create a shopping cart for people to buy products.
- We will discuss this next.

