IERG4210 2014-15 Term 2 Tutorial 10

Wenrui Diao

Department of Information Engineering

The Chinese University of Hong Kong

Outline

1. Hints to Assignment Phase 5

- Reminder:
- The classroom of the tutorial on Tuesday has been changed to NAH213, so you could attend the tutorial without moving.

Assignment Phase 5

- Secure Checkout Flow – using PayPal

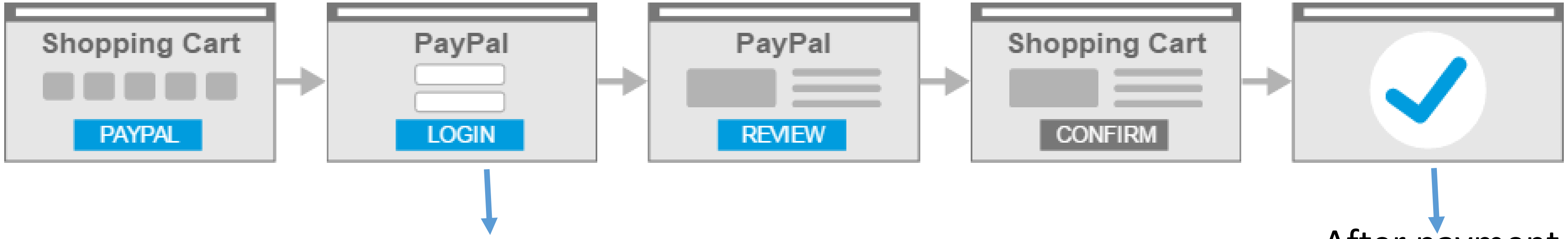
1. Create test accounts at PayPal developer website
2. Create a payment table
3. REST API based PayPal checkout
4. Data checks + Error handle

- For more details, please check http://ierg4210.github.io/web/assign-spec/AssignmentMarkingChecklist5.0.pdf

Accept PayPal with the Payments API

- How it works

1. Get an OAuth access token (client_id, client_secret)
2. Add PayPal buttons to your site → Create a payment → Receive a payment call response
3. Get payment approval and return user
4. Execute payment request



Redirect to PayPal

After payment, redirect to your website

PayPal Sandbox

- The PayPal Sandbox mirrors the features found on the PayPal production servers.

- All Sandbox transactions are tracked on the Sandbox test site (https://www.sandbox.paypal.com), just like live transactions are tracked in the PayPal live site (https://www.paypal.com).

- Assignment Phase 5 is based on Sandbox

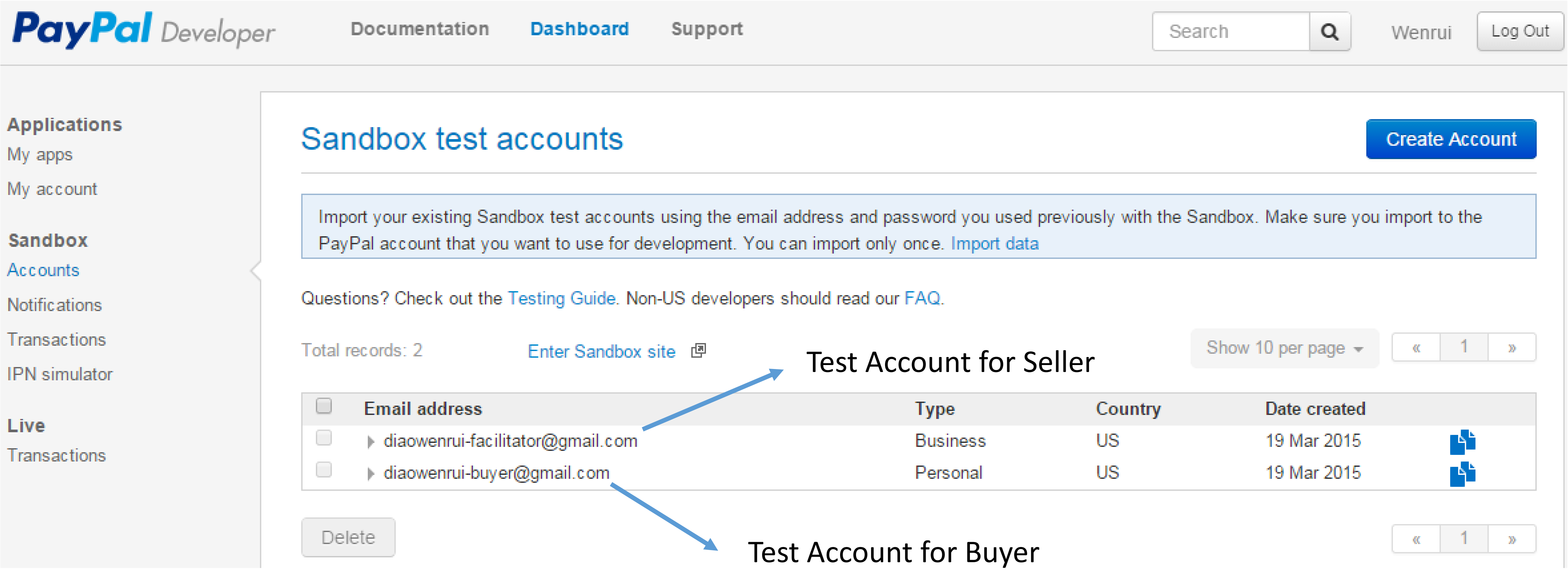
Create test accounts at PayPal developer website

- Sign up PayPal developer account
- https://developer.paypal.com/

- After registration, two test accounts have been created by default
- Virtual PayPal accounts that exist only in the Sandbox environment

Create test accounts at PayPal developer website

- Dashboard → Sandbox → Accounts



Sandbox test accounts

Import your existing Sandbox test accounts using the email address and password you used previously with the Sandbox. Make sure you import to the PayPal account that you want to use for development. You can import only once. Import data

Questions? Check out the Testing Guide. Non-US developers should read our FAQ.

Total records: 2 Enter Sandbox site

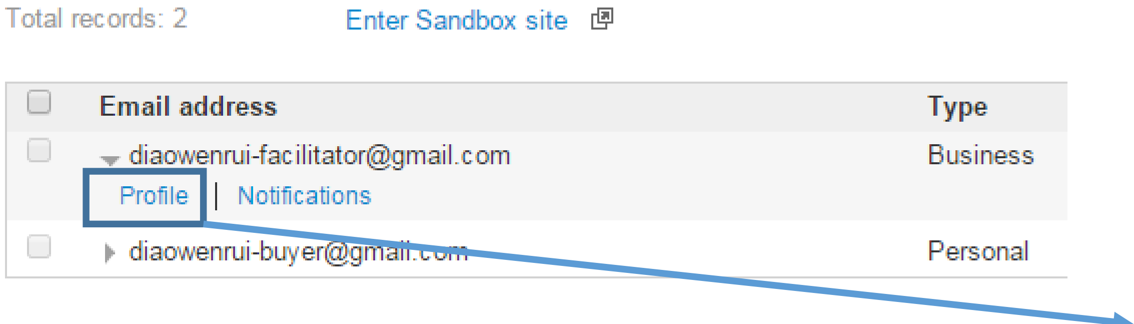
□	Email address	Type	Country	Date created
□	diaowenrui-facilitator@gmail.com	Business	US	19 Mar 2015
□	diaowenrui-buyer@gmail.com	Personal	US	19 Mar 2015

Test Account for Seller

Test Account for Buyer

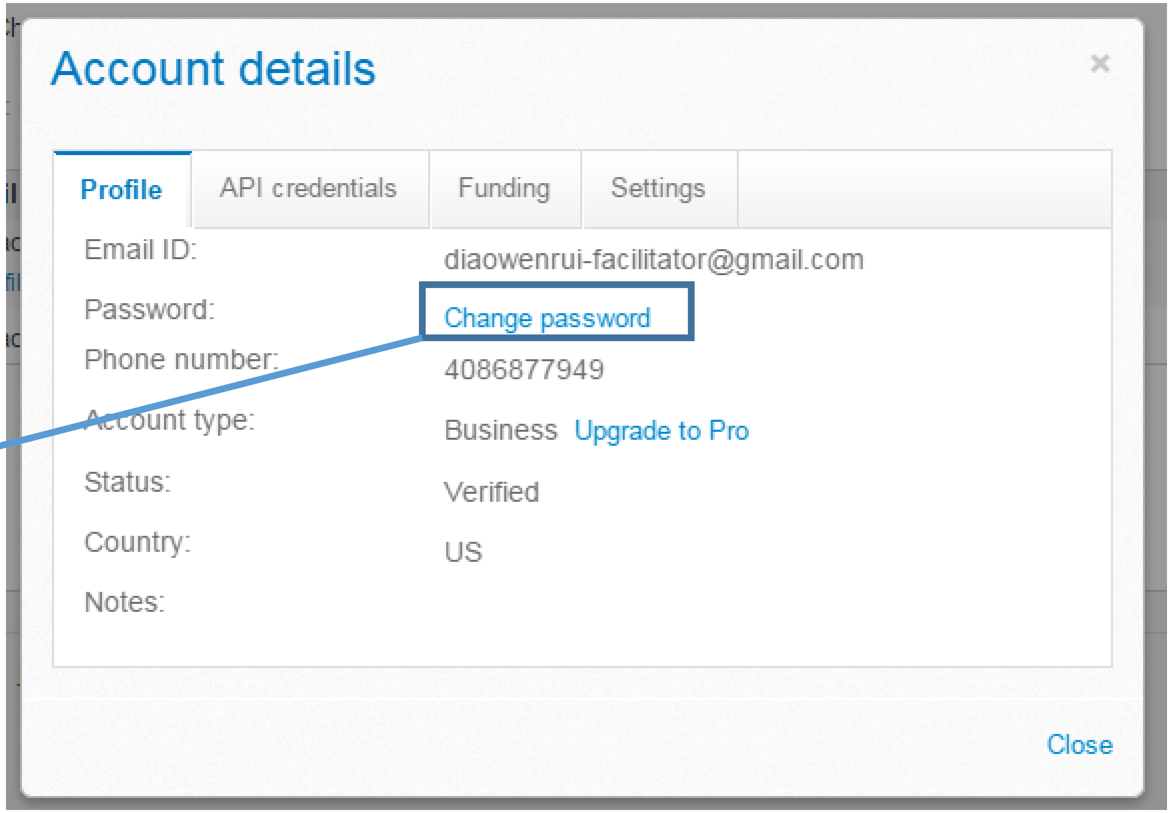
Create test accounts at PayPal developer website

- Change settings of test accounts to suit for your requirements.



Total records: 2 Enter Sandbox site

□	Email address	Type
□	diaowenrui-facilitator@gmail.com Profile | Notifications	Business
□	diaowenrui-buyer@gmail.com	Personal



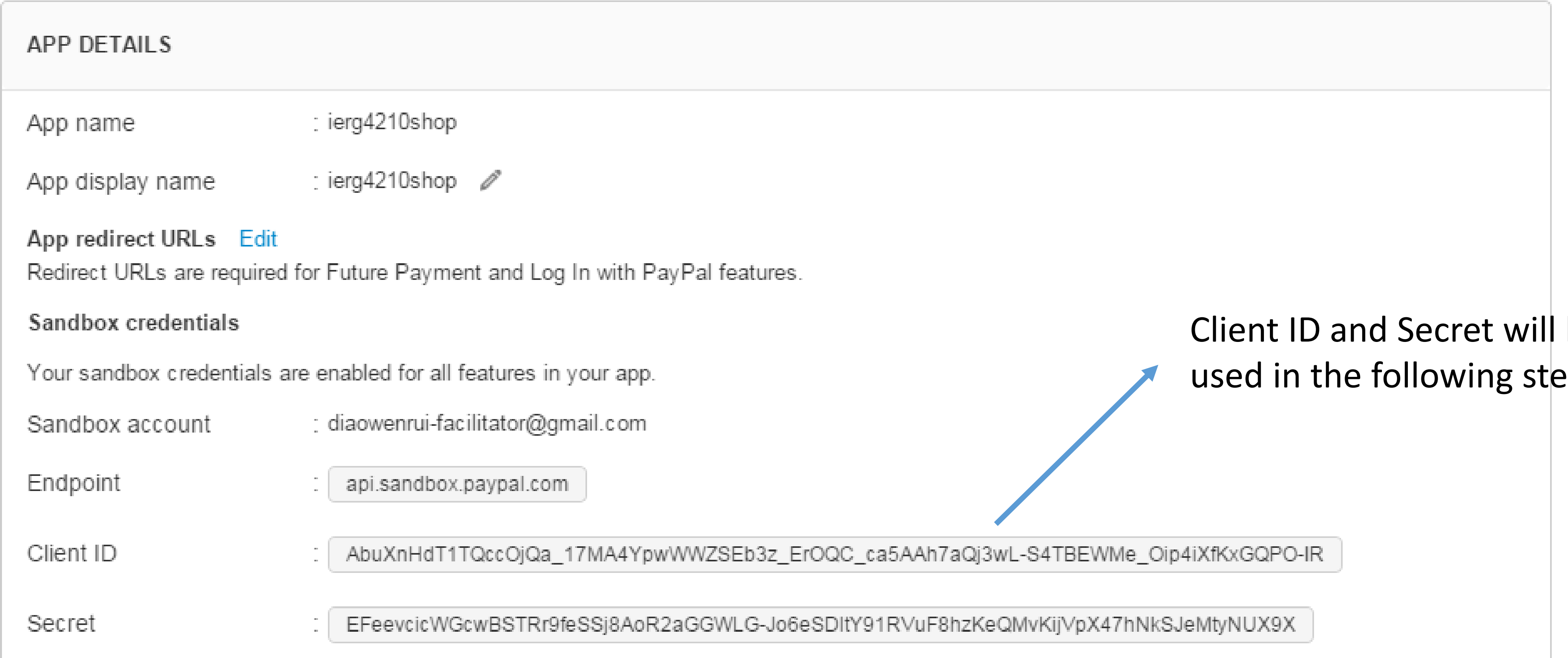
Account details

Email ID:	diaowenrui-facilitator@gmail.com
Password:	Change password
Phone number:	4086877949
Account type:	Business Upgrade to Pro
Status:	Verified
Country:	US
Notes:	

Change to a new password (both seller and buyer) which will be used in the following steps.

Create test accounts at PayPal developer website

- Dashboard → My apps → Create App



APP DETAILS

App name : ierg4210shop

App display name : ierg4210shop

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 : diaowenrui-facilitator@gmail.com

Endpoint : api.sandbox.paypal.com

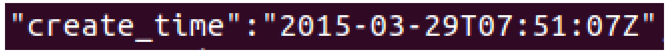
Client ID : AbuXnHdT1TQccOjQa_17MA4YpwWWZSEb3z_ErOQC_ca5AAh7aQj3wL-S4TBEWMe_Oip4iXfKxGQPO-IR

Secret : EFeevcicWGcwBSTRr9feSSj8AoR2aGGWLG-Jo6eSDltY91RVuF8hzKeQMvKijVpX47hNkSJeMtyNUX9X

Client ID and Secret will be used in the following steps.

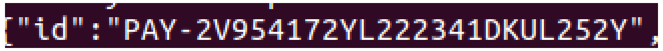
Create a payment table

- Create a payment table
- -- Required columns: payid (primary key), userid, paymentId, state, dateCreated
- -- Required index: (userid, paymentId), (state)



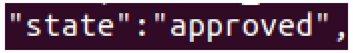
```
"create_time":"2015-03-29T07:51:07Z"
```

Generated by PayPal



```
{"id":"PAY-2V954172YL222341DKUL252Y",
```

Payment Status, got from PayPal



```
"state":"approved",
```

- You could add more columns
- Think twice what columns are needed in the table

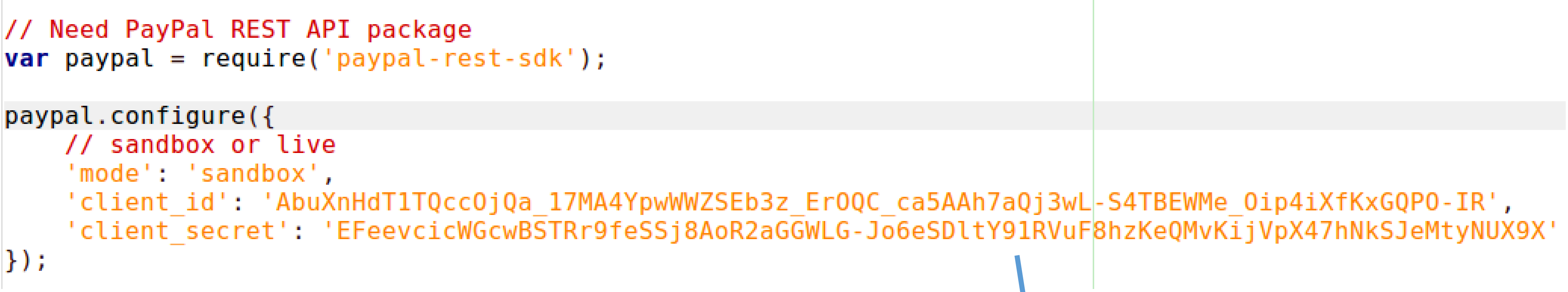
REST API based PayPal checkout

- Based on PayPal REST API
- $npm install --save paypal-rest-sdk

If you are interested in RESTful style, read http://en.wikipedia.org/wiki/Representational_state_transfer

- Reference:
- paypal-rest-sdk: https://www.npmjs.com/package/paypal-rest-sdk
- REST API: https://developer.paypal.com/docs/api/
- Sample Code (prepared by Dr. Fung): https://gist.github.com/adon-at-work/48e77447fb0513ec375c

REST API based PayPal checkout



```
// Need PayPal REST API package
var paypal = require('paypal-rest-sdk');

paypal.configure({
    // sandbox or live
    'mode': 'sandbox',
    'client_id': 'AbuXnHdT1TQccOjQa_17MA4YpwWWZSEb3z_ErOQC_ca5AAh7aQj3wL-S4TBEWMe_Oip4iXfKxGQPO-IR',
    'client_secret': 'EFeevcicWGcwBSTRr9feSSj8AoR2aGGWLG-Jo6eSDltY91RVuF8hzKeQMvKijVpX47hNkSJeMtyNUX9X'
});
```

Change client_id and client_secret to your own, see page 9.

REST API based PayPal checkout

- Submit the shopping list to /checkout (i.e., with item’s pids and qtys)
- Checkout page hosted at /checkout
  - Create a payment token using paypal.payment.create()
- Error page hosted at /checkout/error
- Thank you page hosted at /checkout/thankyou

REST API based PayPal checkout

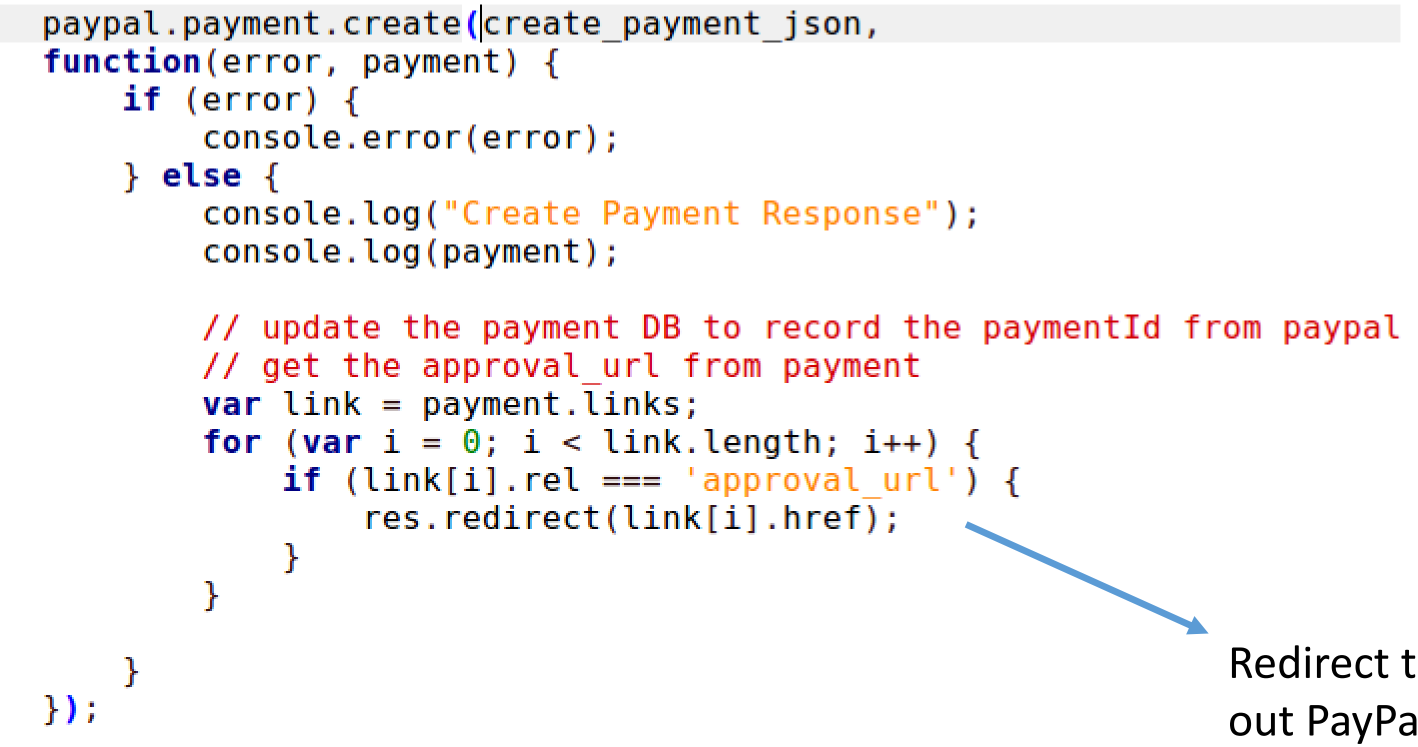
- You should study the code in phase 3b (shopping cart update) before doing this task
- Only pid and quantity should be gotten from client. Others should be saved on server side (Prevent tampering)
- You need to apply some input checks on Checkout page / Error page / Thank you page

Such as whether requested products == paid products?

Such as a product which doesn’t exist in your database

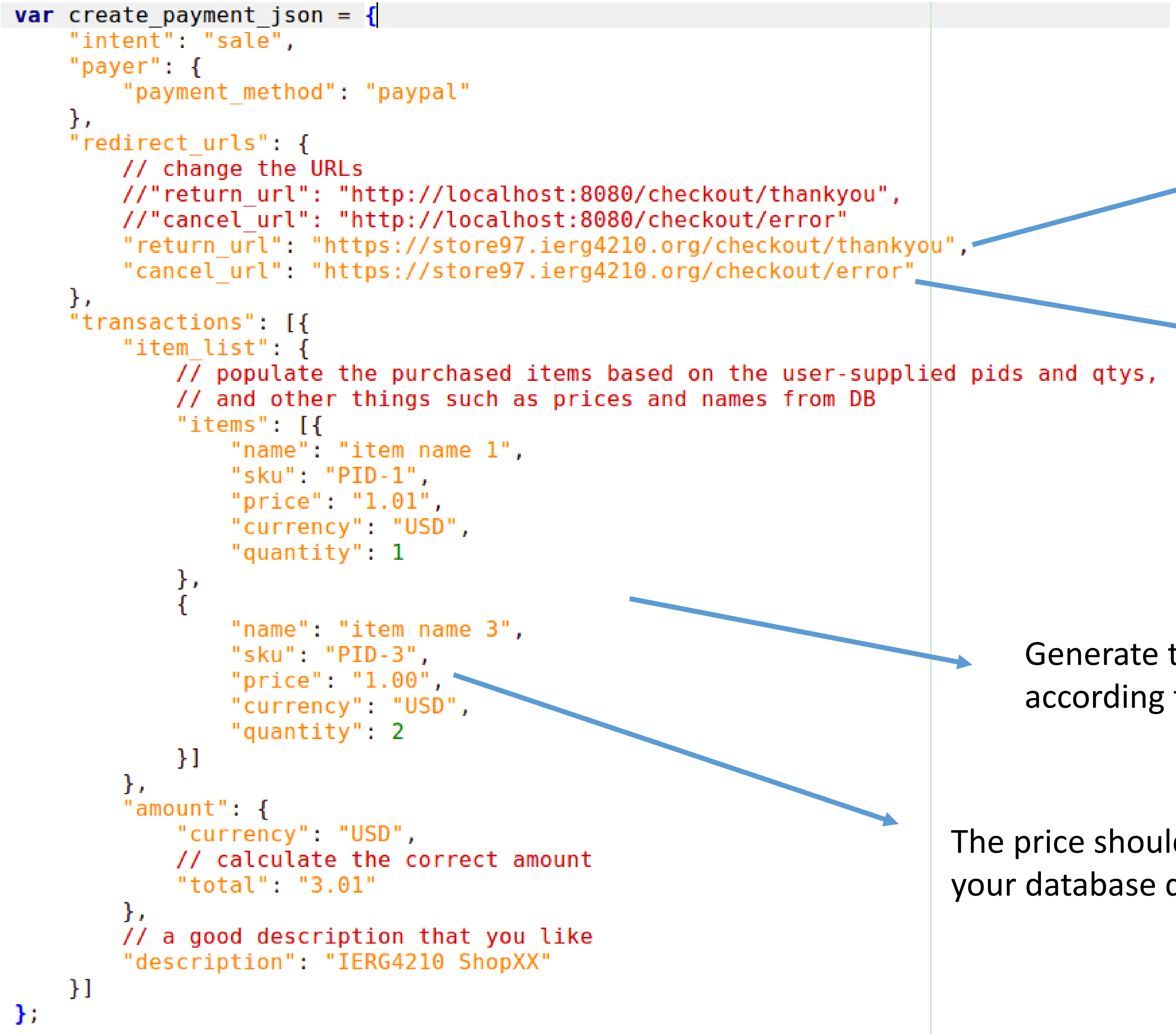
REST API based PayPal checkout -- Checkout page

- In app.get('/checkout', function(req, res), { … });



```
paypal.payment.create(create_payment_json,
function(error, payment) {
    if (error) {
        console.error(error);
    } else {
        console.log("Create Payment Response");
        console.log(payment);

        // update the payment DB to record the paymentId from paypal
        // get the approval_url from payment
        var link = payment.links;
        for (var i = 0; i < link.length; i++) {
            if (link[i].rel === 'approval_url') {
                res.redirect(link[i].href);
            }
        }

    }
});
```

Redirect to approval_url to carry out PayPal payment operation



```
var create_payment_json = {
    "intent": "sale",
    "payer": {
        "payment_method": "paypal"
    },
    "redirect_urls": {
        // change the URLs
        //"return_url": "http://localhost:8080/checkout/thankyou",
        //"cancel_url": "http://localhost:8080/checkout/error"
        "return_url": "https://store97.ierg4210.org/checkout/thankyou",
        "cancel_url": "https://store97.ierg4210.org/checkout/error"
    },
    "transactions": [{
        "item_list": {
            // populate the purchased items based on the user-supplied pids and qtys,
            // and other things such as prices and names from DB
            "items": [{
                "name": "item name 1",
                "sku": "PID-1",
                "price": "1.01",
                "currency": "USD",
                "quantity": 1
            },
            {
                "name": "item name 3",
                "sku": "PID-3",
                "price": "1.00",
                "currency": "USD",
                "quantity": 2
            }]
        },
        "amount": {
            "currency": "USD",
            // calculate the correct amount
            "total": "3.01"
        },
        // a good description that you like
        "description": "IERG4210 ShopXX"
    }]
};
```

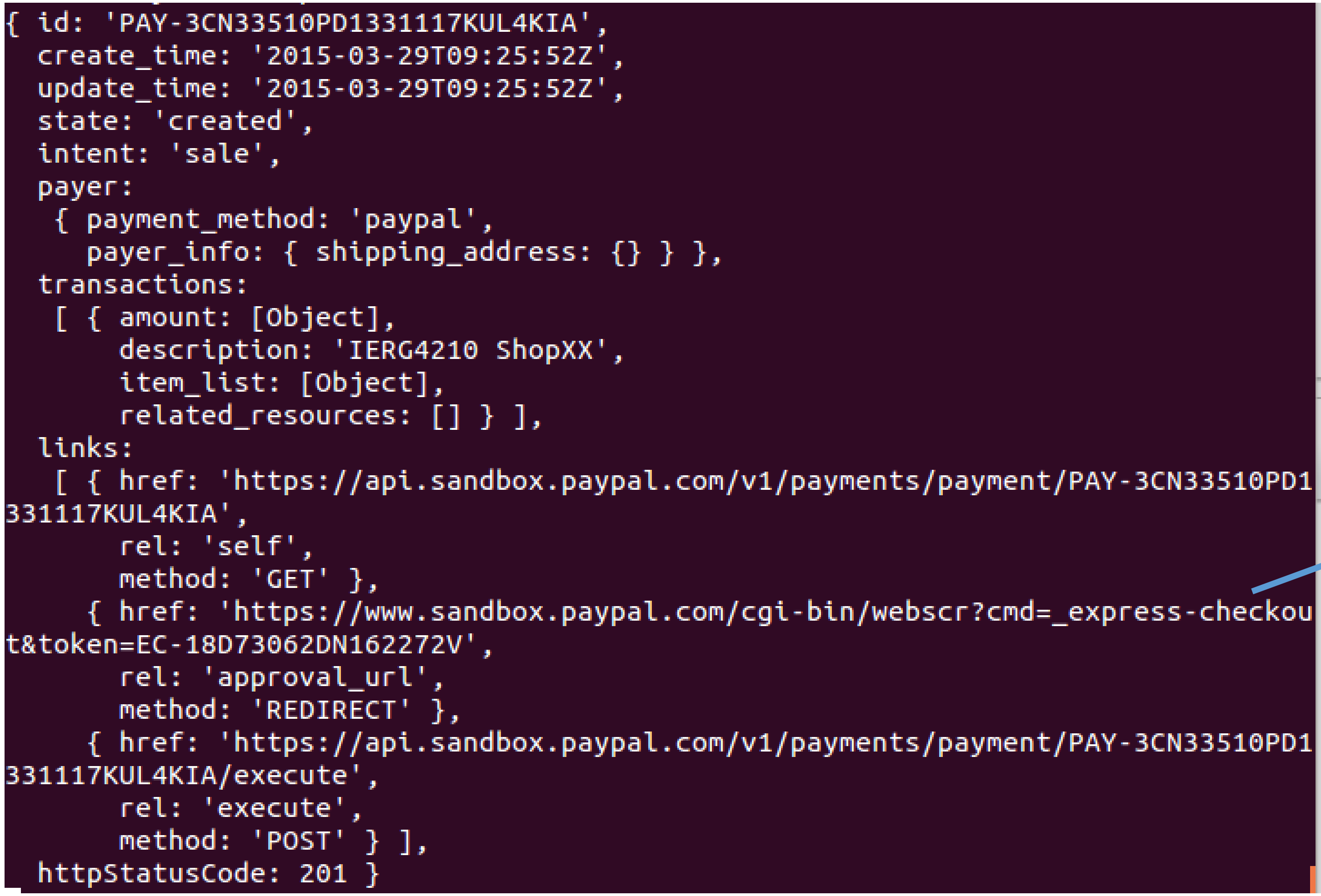
Redirect after payment success

Redirect after payment fail

Generate transactions dynamically
according to items in your shopping cart

The price should come from
your database directly

REST API based PayPal checkout
-- Response of paypal.payment.create()



```
{ id: 'PAY-3CN33510PD1331117KUL4KIA',
  create_time: '2015-03-29T09:25:52Z',
  update_time: '2015-03-29T09:25:52Z',
  state: 'created',
  intent: 'sale',
  payer:
   { payment_method: 'paypal',
     payer_info: { shipping_address: {} } },
  transactions:
   [ { amount: [Object],
       description: 'IERG4210 ShopXX',
       item_list: [Object],
       related_resources: [] } ],
  links:
   [ { href: 'https://api.sandbox.paypal.com/v1/payments/payment/PAY-3CN33510PD1
331117KUL4KIA',
       rel: 'self',
       method: 'GET' },
     { href: 'https://www.sandbox.paypal.com/cgi-bin/webscr?cmd=_express-checkou
t&token=EC-18D73062DN162272V',
       rel: 'approval_url',
       method: 'REDIRECT' },
     { href: 'https://api.sandbox.paypal.com/v1/payments/payment/PAY-3CN33510PD1
331117KUL4KIA/execute',
       rel: 'execute',
       method: 'POST' } ],
  httpStatusCode: 201 }
```

approval_url

REST API based PayPal checkout
-- Error page

- If payment error, PayPal will redirect to your error page like: https://store97.ierg4210.org/checkout/error?token=EC-6FV02294NK967922C

- Requirement: Give a [Retry] hyperlink to the payment URL again

REST API based PayPal checkout -- Error page

- Solution:
- Construct an approval_url:

- var token = req.query.token;

- <a href="https://www.sandbox.paypal.com/cgi-bin/webscr?cmd=_express-checkout&amp;token={{token}}"> Retry Payment </a>
- For example: https://www.sandbox.paypal.com/cgi-bin/webscr?cmd=_express-checkout&token=EC-6FV02294NK967922C

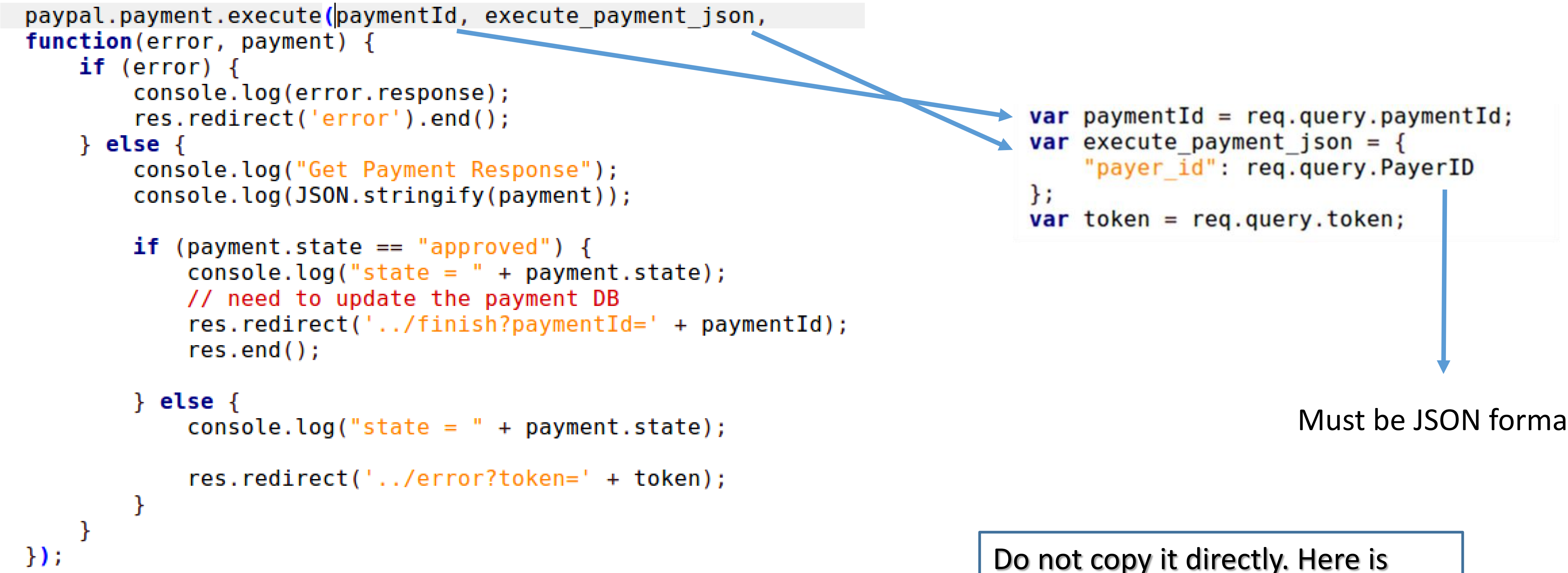
REST API based PayPal checkout
-- Thank you page

- If payment success, PayPal will redirect to your thank you page, like:
- https://store97.ierg4210.org/checkout/thankyou?paymentId=PAY-1J122343002417042KUL46OI&token=EC-50P06565XB3038503&PayerID=Z7MZW2WQMMEJA

- Claim the payment by using paypal.payment.execute()

REST API based PayPal checkout
-- Invoke paypal.payment.execute()

- In app.get('/checkout/thankyou', function(req, res) { …});



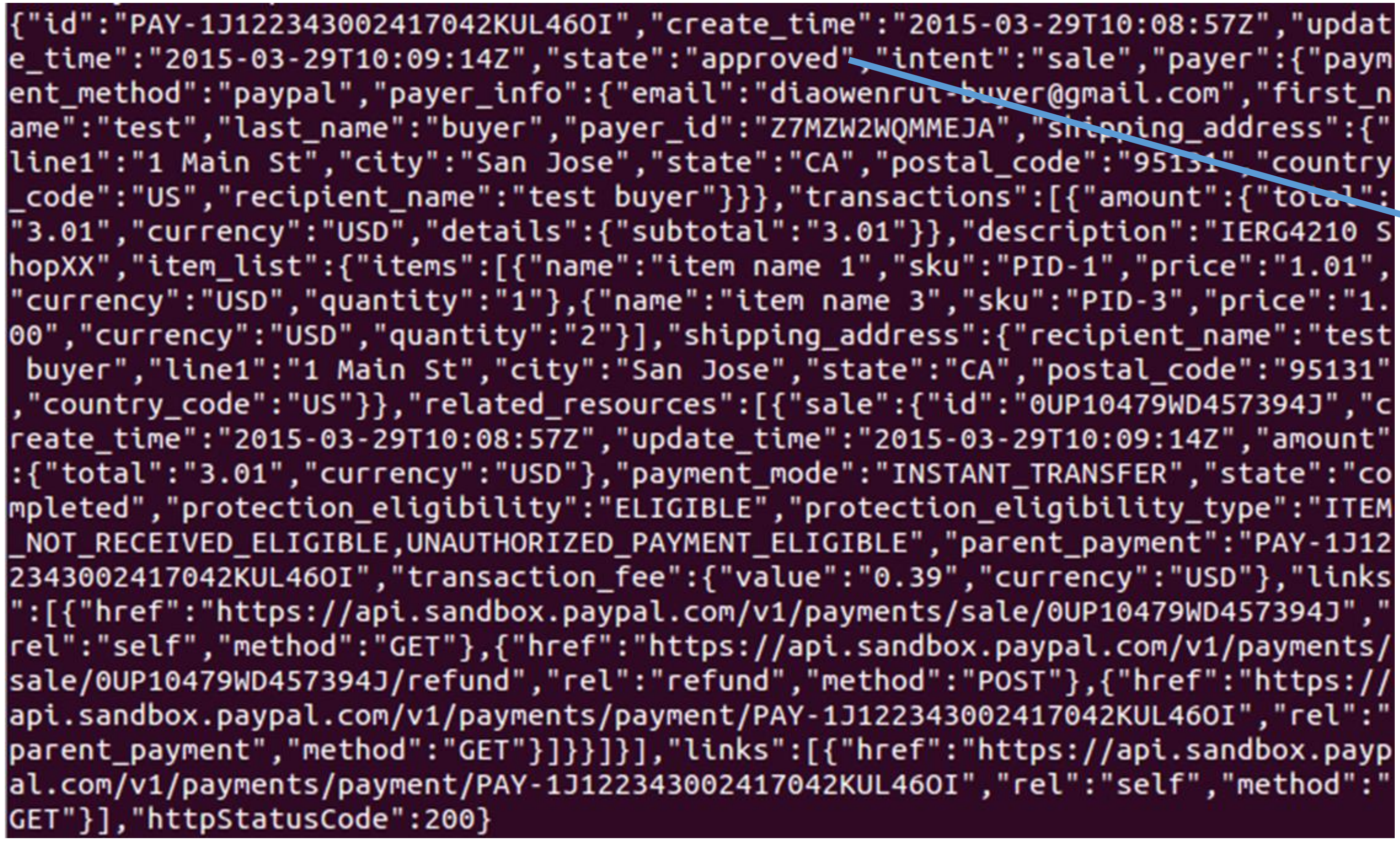
```
paypal.payment.execute(paymentId, execute_payment_json,
function(error, payment) {
    if (error) {
        console.log(error.response);
        res.redirect('error').end();
    } else {
        console.log("Get Payment Response");
        console.log(JSON.stringify(payment));

        if (payment.state == "approved") {
            console.log("state = " + payment.state);
            // need to update the payment DB
            res.redirect('../finish?paymentId=' + paymentId);
            res.end();

        } else {
            console.log("state = " + payment.state);

            res.redirect('../error?token=' + token);
        }
    }
});
```

```
var paymentId = req.query.paymentId;
var execute_payment_json = {
    "payer_id": req.query.PayerID
};
var token = req.query.token;
```

Must be JSON format

Do not copy it directly. Here is only an example.

REST API based PayPal checkout
-- Response of paypal.payment.execute()



```
{"id":"PAY-1J122343002417042KUL46OI","create_time":"2015-03-29T10:08:57Z","updat
e_time":"2015-03-29T10:09:14Z","state":"approved","intent":"sale","payer":{"paym
ent_method":"paypal","payer_info":{"email":"diaowenrui-buyer@gmail.com","first_n
ame":"test","last_name":"buyer","payer_id":"Z7MZW2WQMMEJA","shipping_address":{"
line1":"1 Main St","city":"San Jose","state":"CA","postal_code":"95131","country
_code":"US","recipient_name":"test buyer"}}},"transactions":[{"amount":{"total":
"3.01","currency":"USD","details":{"subtotal":"3.01"}},"description":"IERG4210 S
hopXX","item_list":{"items":[{"name":"item name 1","sku":"PID-1","price":"1.01",
"currency":"USD","quantity":"1"},{"name":"item name 3","sku":"PID-3","price":"1.
00","currency":"USD","quantity":"2"}],"shipping_address":{"recipient_name":"test
 buyer","line1":"1 Main St","city":"San Jose","state":"CA","postal_code":"95131"
,"country_code":"US"}},"related_resources":[{"sale":{"id":"0UP10479WD457394J","c
reate_time":"2015-03-29T10:08:57Z","update_time":"2015-03-29T10:09:14Z","amount"
:{"total":"3.01","currency":"USD"},"payment_mode":"INSTANT_TRANSFER","state":"co
mpleted","protection_eligibility":"ELIGIBLE","protection_eligibility_type":"ITEM
_NOT_RECEIVED_ELIGIBLE,UNAUTHORIZED_PAYMENT_ELIGIBLE","parent_payment":"PAY-1J12
2343002417042KUL46OI","transaction_fee":{"value":"0.39","currency":"USD"},"links
":[{"href":"https://api.sandbox.paypal.com/v1/payments/sale/0UP10479WD457394J","
rel":"self","method":"GET"},{"href":"https://api.sandbox.paypal.com/v1/payments/
sale/0UP10479WD457394J/refund","rel":"refund","method":"POST"},{"href":"https://
api.sandbox.paypal.com/v1/payments/payment/PAY-1J122343002417042KUL46OI","rel":"
parent_payment","method":"GET"}]}}]}],"links":[{"href":"https://api.sandbox.payp
al.com/v1/payments/payment/PAY-1J122343002417042KUL46OI","rel":"self","method":"
GET"}],"httpStatusCode":200}
```

approved

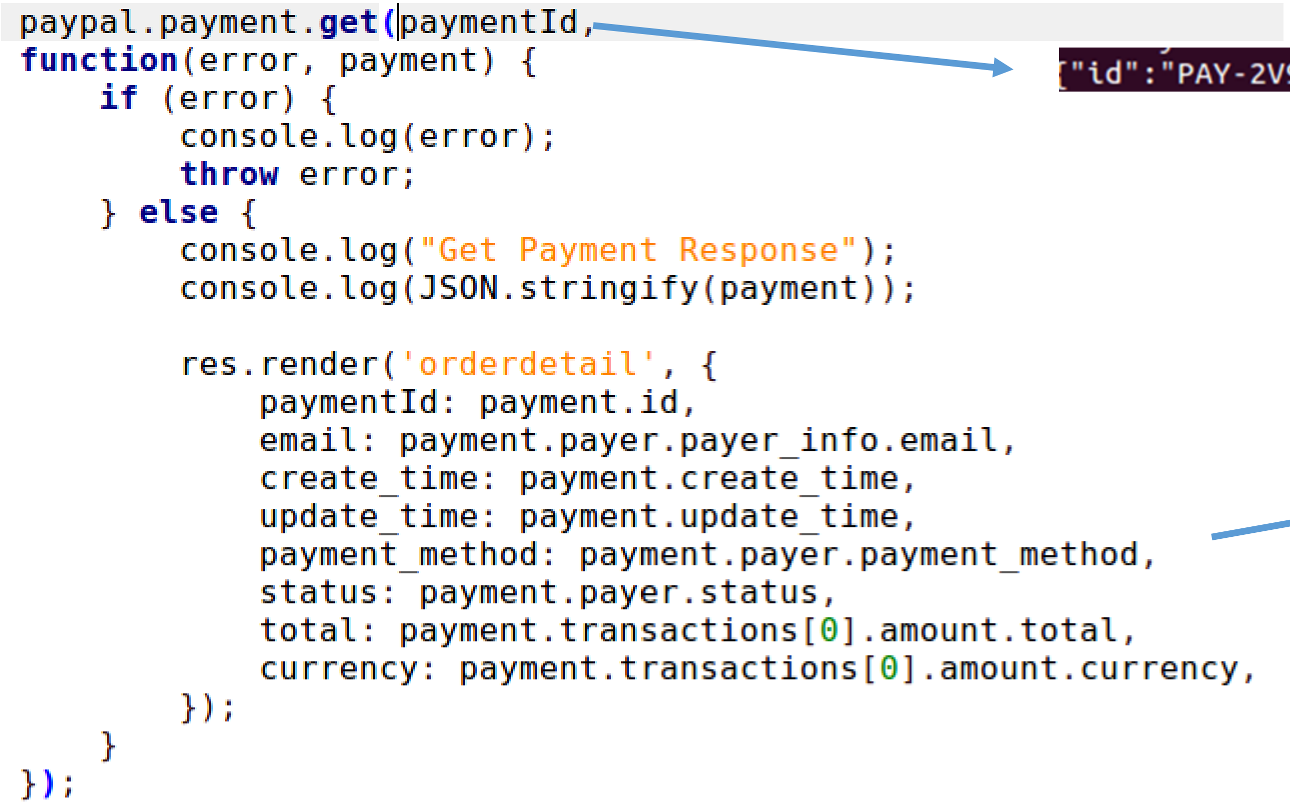
REST API based PayPal checkout -- Account page

- Display a table showing the user-specific paymentId and state in chronological order
- -- Database query

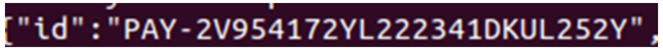
- When clicked a paymentId, Retrieve the items purchased using paypal.payment.get()

- Remember to also clear the local storage of shopping cart

REST API based PayPal checkout
-- Invoke paypal.payment.get()



```
paypal.payment.get(paymentId,
function(error, payment) {
    if (error) {
        console.log(error);
        throw error;
    } else {
        console.log("Get Payment Response");
        console.log(JSON.stringify(payment));

        res.render('orderdetail', {
            paymentId: payment.id,
            email: payment.payer.payer_info.email,
            create_time: payment.create_time,
            update_time: payment.update_time,
            payment_method: payment.payer.payment_method,
            status: payment.payer.status,
            total: payment.transactions[0].amount.total,
            currency: payment.transactions[0].amount.currency,
        });
    }
});
```

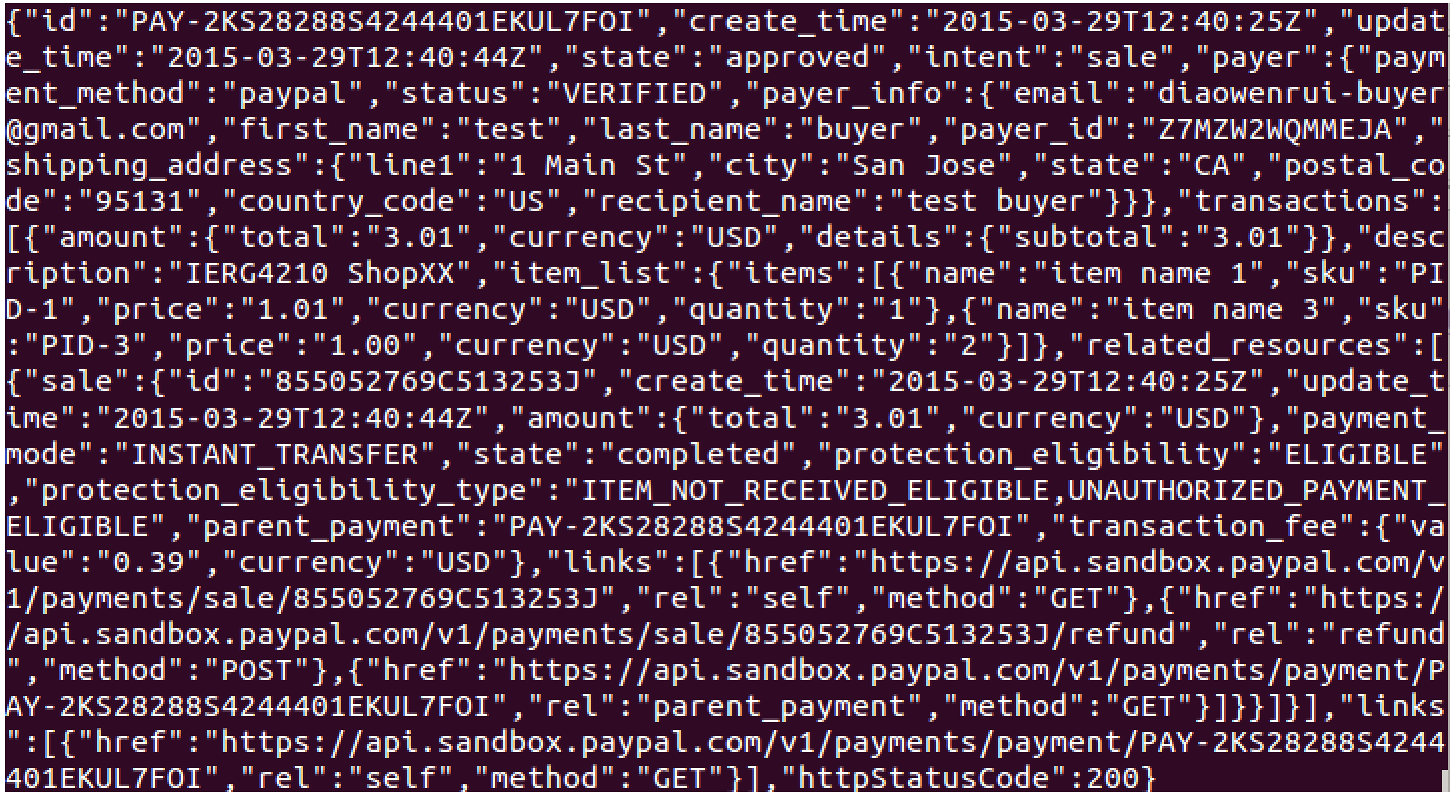


```
{"id":"PAY-2V954172YL222341DKUL252Y",
```

Change them according
to your requirement

Do not copy it directly. Here is
only an example.

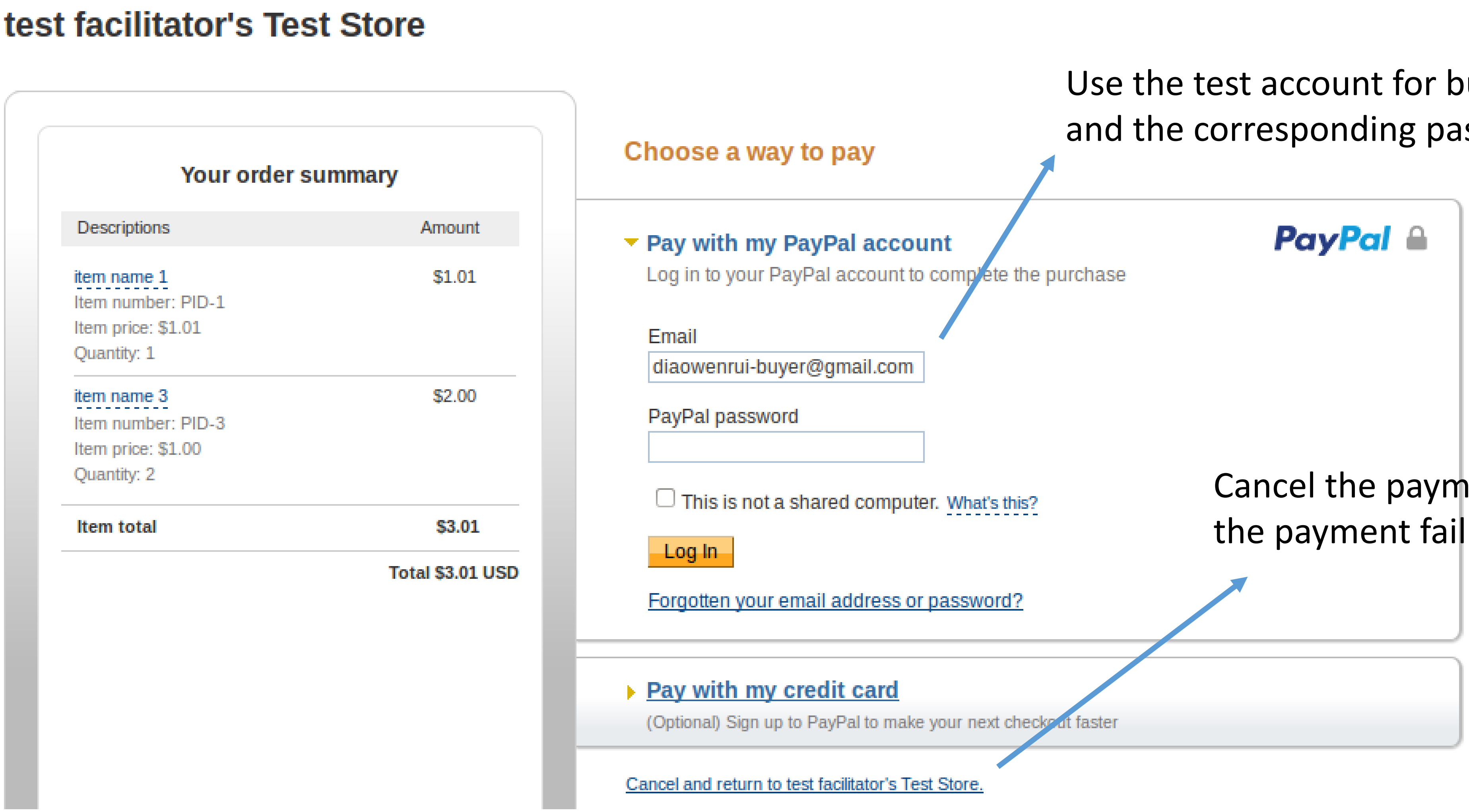
REST API based PayPal checkout
-- Response of paypal.payment.get()



```
{"id":"PAY-2KS28288S4244401EKUL7FOI","create_time":"2015-03-29T12:40:25Z","updat
e_time":"2015-03-29T12:40:44Z","state":"approved","intent":"sale","payer":{"paym
ent_method":"paypal","status":"VERIFIED","payer_info":{"email":"diaowenrui-buyer
@gmail.com","first_name":"test","last_name":"buyer","payer_id":"Z7MZW2WQMMEJA","
shipping_address":{"line1":"1 Main St","city":"San Jose","state":"CA","postal_co
de":"95131","country_code":"US","recipient_name":"test buyer"}}},"transactions":
[{"amount":{"total":"3.01","currency":"USD","details":{"subtotal":"3.01"}},"desc
ription":"IERG4210 ShopXX","item_list":{"items":[{"name":"item name 1","sku":"PI
D-1","price":"1.01","currency":"USD","quantity":"1"},{"name":"item name 3","sku"
:"PID-3","price":"1.00","currency":"USD","quantity":"2"}]},"related_resources":[
{"sale":{"id":"855052769C513253J","create_time":"2015-03-29T12:40:25Z","update_t
ime":"2015-03-29T12:40:44Z","amount":{"total":"3.01","currency":"USD"},"payment_
mode":"INSTANT_TRANSFER","state":"completed","protection_eligibility":"ELIGIBLE"
,"protection_eligibility_type":"ITEM_NOT_RECEIVED_ELIGIBLE,UNAUTHORIZED_PAYMENT_
ELIGIBLE","parent_payment":"PAY-2KS28288S4244401EKUL7FOI","transaction_fee":{"va
lue":"0.39","currency":"USD"},"links":[{"href":"https://api.sandbox.paypal.com/v
1/payments/sale/855052769C513253J","rel":"self","method":"GET"},{"href":"https:/
/api.sandbox.paypal.com/v1/payments/sale/855052769C513253J/refund","rel":"refund
","method":"POST"},{"href":"https://api.sandbox.paypal.com/v1/payments/payment/P
AY-2KS28288S4244401EKUL7FOI","rel":"parent_payment","method":"GET"}]}}]}],"links
":[{"href":"https://api.sandbox.paypal.com/v1/payments/payment/PAY-2KS28288S4244
401EKUL7FOI","rel":"self","method":"GET"}],"httpStatusCode":200}
```

Resolve it carefully,
many items….

Test -- Checkout



test facilitator's Test Store

Your order summary

Descriptions	Amount
item name 1 Item number: PID-1 Item price: $1.01 Quantity: 1	$1.01
item name 3 Item number: PID-3 Item price: $1.00 Quantity: 2	$2.00
Item total	**$3.01**
	Total $3.01 USD

Choose a way to pay

Pay with my PayPal account

Log in to your PayPal account to complete the purchase

Email: diaowenrui-buyer@gmail.com

PayPal password:

□ This is not a shared computer. What's this?

Pay with my credit card

(Optional) Sign up to PayPal to make your next checkout faster

Cancel and return to test facilitator's Test Store.

Use the test account for buyer and the corresponding password

Cancel the payment to test the payment failure case

Just A Reminder

- Make sure you have enough time to do it.
- It is difficult to finish the whole phase in two/three days.

- Make sure you have finished previous phases about session management and shopping cart update.

Demo and Q&A

- PayPal Official Reference:
- Sandbox overview: https://developer.paypal.com/docs/classic/lifecycle/sb_overview/
- Accept PayPal with the Payments API: https://developer.paypal.com/docs/integration/web/web-checkout/
- Samples for PayPal-node-SDK: https://github.com/paypal/PayPal-node-SDK/tree/master/samples
- Rest API Sample app in Node.js: https://github.com/paypal/rest-api-sample-app-nodejs