



PaymentRequest API

Enhancing mobile ecommerce with quick simple checkout

SC Kim



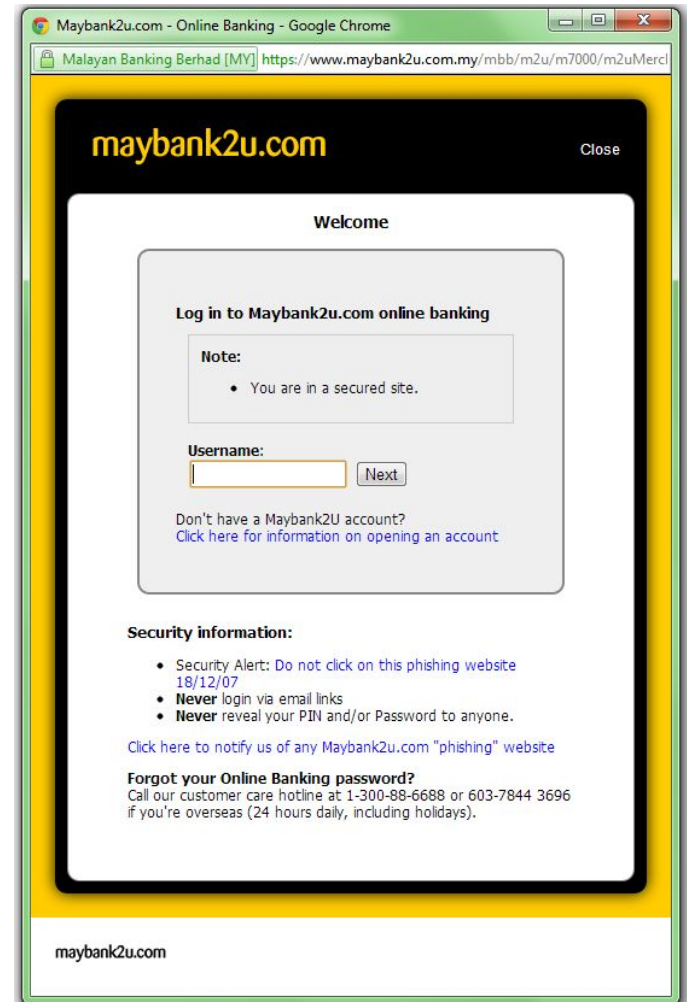
Before We Start

1. 10secs Introduction - Former Google Student Ambassador SEA 14/15, Full Stack Freelancer (Web & Mobile), Master (Computer Science) pursuing Augmented Reality Studies, Firebase Keynote Speaker MOSCMY 2016
2. How many of us here are online shoppers E.g Lazada, Lelong, Taobao
3. Do you purchase items via mobile apps or websites with external payment popup/redirects such as Maybank2U, CIMBClicks?
4. Do you remember 16digits credit card number, CVC, expiration without glimpse?
5. Any ecommerce developers?



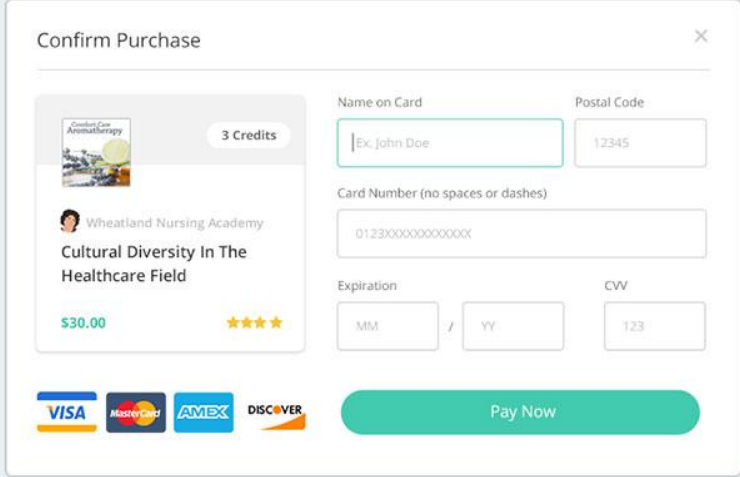
Dilemma 1 : Maybank2U

1. Add items to cart
2. Checkout cart
3. Start payment
4. Open Browser popup > Maybank2U
5. Login, Secret Key, Wait OTP
6. Done Payment



Dilemma 2 : Custom Ecommerce Website

1. Add items to cart
2. Checkout cart
3. Start payment
4. Redirect to their embeded payment website,
5. Key in
 - a. Name
 - b. 16 digits,
 - c. CVC,
 - d. Expiration
6. Wait OTP
7. Done Payment



A screenshot of a 'Confirm Purchase' payment form. The form is titled 'Confirm Purchase' with a close button (X) in the top right corner. It features a summary of the purchase on the left and payment details on the right. The summary includes a product image, '3 Credits', the merchant name 'Wheatland Nursing Academy', the item description 'Cultural Diversity In The Healthcare Field', a price of '\$30.00', and a five-star rating. Below the summary are logos for VISA, MasterCard, AMEX, and DISCOVER. The payment details section includes fields for 'Name on Card' (with placeholder 'Ex. John Doe'), 'Postal Code' (with placeholder '12345'), 'Card Number (no spaces or dashes)' (with placeholder '0123XXXXXXX0000000000'), 'Expiration' (with MM and YY sub-fields), and 'CVC' (with placeholder '123'). A large green 'Pay Now' button is at the bottom right.

Source: <http://bashooka.com>



Dilemma 3 - More, more and more payment key-in

The image displays four mobile app screenshots illustrating payment flows:

- Best Buy:** Shows the 'Order Summary' for \$36.62. The payment section asks 'How do you want to pay today?' with options for 'Credit Card or Debit Card' (selected), 'PayPal', and 'Best Buy Gift Card or Savings Code'. A promotional banner for the My Best Buy Credit Card is also visible.
- Newegg:** Shows the 'Secure Checkout' process. The 'Payment Methods' section has a dropdown menu set to 'Credit Card'. Below it, the 'Add New Credit Card' form is shown with fields for Cardholder Name, Card Number, and Expiration Date. A 'SAVE' button is present.
- Amazon:** Shows the 'Add a credit or debit card' screen. It prompts the user to 'Enter your credit card information' with fields for Name on card, Card number, and Expiration date. A 'Use this card across all of Amazon' checkbox is also present.
- Google Store:** Shows the 'Add new credit or debit card' screen. It prompts the user to enter card information with fields for Card number, MM / YY, CVC, Cardholder name, United States (US), and ZIP code. Error messages like 'Card number is required' and 'Cardholder name is required' are visible.

Users abandon mobile purchases **twice** as often as desktop purchases. Why?

&

66 % fewer conversion rates on mobile compared to desktop



Manual



Tedious



Slow

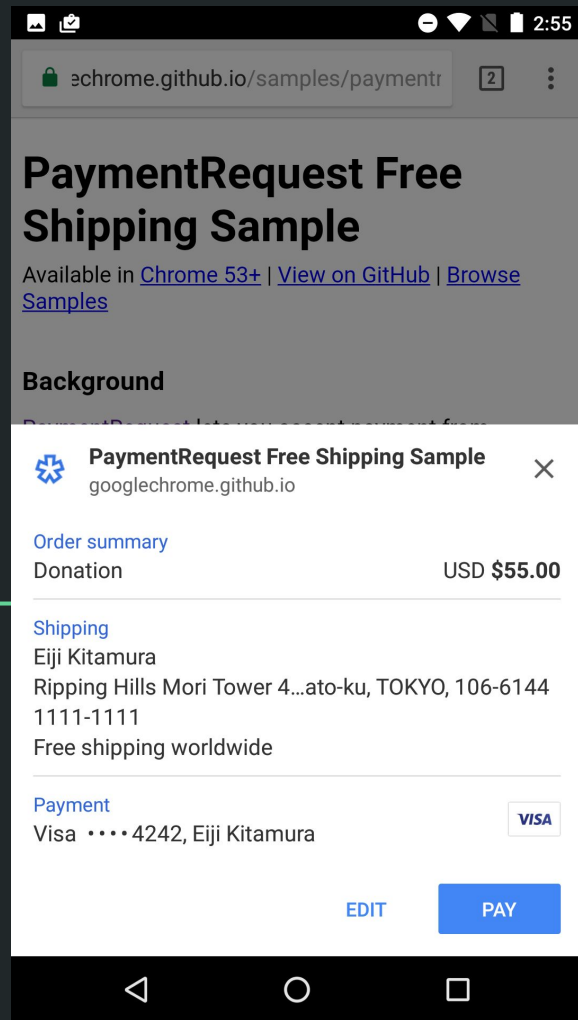


N-taps

Here comes PaymentRequest API

Seamless, easy-to-use checkout experience

<https://w3c.github.io/browser-payment-api/>



101 About PaymentRequest API

- Payment Request is a new API for the open web that makes checkout flows easier, faster and consistent on shopping sites.
- User agents (e.g., browsers) **facilitate the payment flow** between merchant and user.
- Collaborated with **Web Payments Working Group** (Google, Mozilla, Samsung, Alibaba, Microsoft, and more)
- NOT a new payment method
- NOT going to be payment processor/gateway, but work together with those



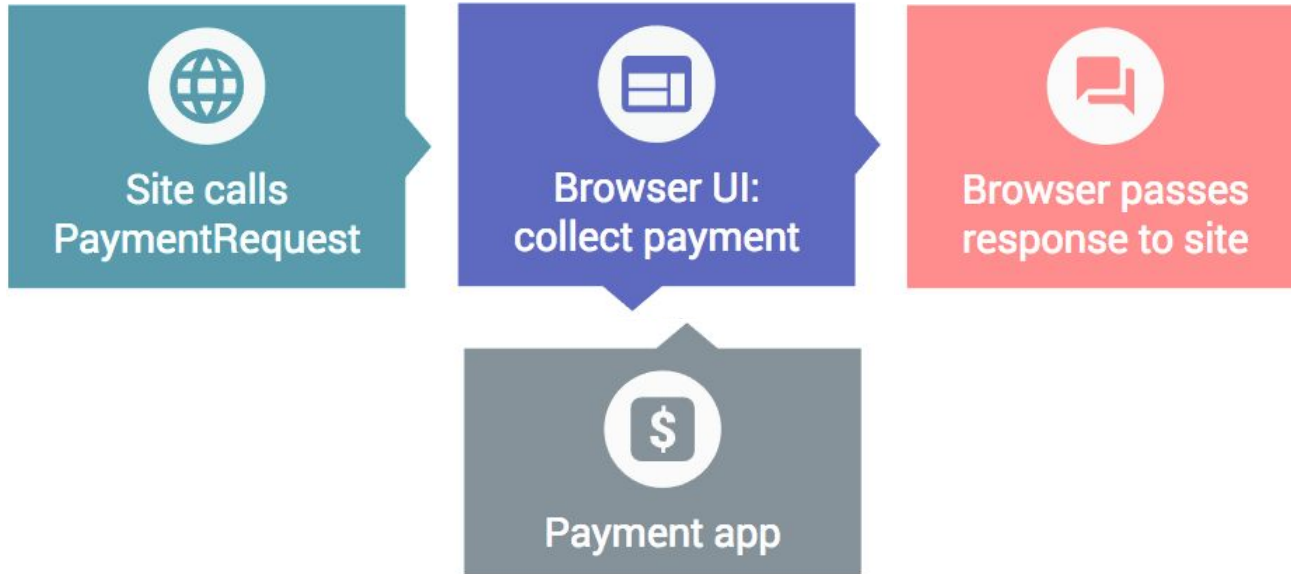
Microsoft

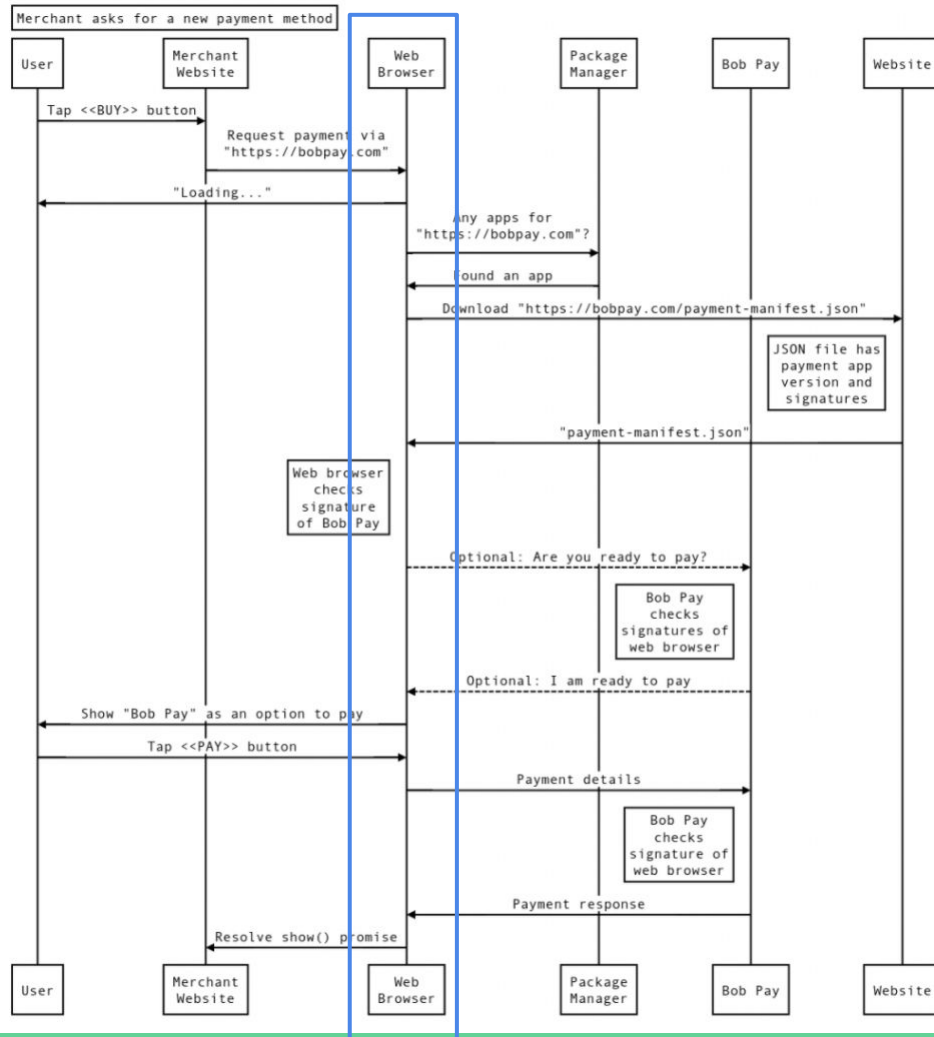


Alipay™



How Does It Works ?





Too Simple ?

Here's Behind-The-Scene which usual processes dealt by online merchant developers

```
<manifest package="com.bobpay.app">
  <service android:name=".IsReadyToPayService"
    android:enabled="true"
    android:exported="true">
    <intent-filter>
      <action android:name="org.chromium.intent.action.IS_READY_TO_PAY" />
    </intent-filter>
  </service>
  <activity android:name=".PaymentActivity"
    android:exported="true">
    <intent-filter>
      <action android:name="org.chromium.intent.action.PAY" />
    </intent-filter>
    <meta-data android:name="org.chromium.default_payment_method_name"
      android:value="https://bobpay.com/put/optional/path/here" />
  </activity>
</manifest>
```

Let's say 'Bob Pay'
as creditcard app in
Android

When Intent calls BobPay to start its service

```
Bundle extras = new Bundle();  
extras.putString("key", "value");  
intent.putExtras(extras);
```

What is being purchased ?
Which vendor ?
For how much money is contained ?

Once payment validated, Bob Pay send another intent back to PaymentRequest API

```
Intent result = new Intent();  
Bundle extras = new Bundle();  
extras.putString("key", "value");  
result.putExtras(extras);  
setResult(RESULT_OK, result); // Change to RESULT_CANCELED on  
failure.  
finish(); // Close the payment activity.
```



Wait, what if the BobPay in your phone is not Real BobPay app ?

Payment Method Manifest Identifier JSON file

(machine-readable by the browser)



payment-manifest.json

EXAMPLE 1

```
// payment-manifest.json
{
  // Android:
  "android": [{
    "package": "xyz.bobay.app",
    "version": 10,
    "sha256_cert_fingerprints": ["14:6D:E9:83:C5:73:06:50"],
    "platforms": {
      "play": "https://play.google.com/store/apps/details?id=com.bobspayments.app1"
    }
  }],
  // Web:
  "name": "Bob's Payments",
  "icons": [{
    "src": "icon/lowres.webp",
    "sizes": "48x48",
    "type": "image/webp"
  }, {
    "src": "icon/lowres",
    "sizes": "48x48"
  }, {
    "src": "icon/hd_hi.ico",
    "sizes": "72x72 96x96 128x128 256x256"
  }, {
    "src": "icon/hd_hi.svg",
    "sizes": "257x257"
  }],
  // iOS:
  "ios": {...},
  // Other supported payment apps:
  "externally_supported_apps": ["https://alicepay.com/pay"]
}
```



Supports for PaymentRequest

Google

May 10th, support for third-party Android payment apps are now enabled by default in Chromium since the web payment manifest prerequisite was approved in the blink-dev branch (Blink is the name of the rendering engine used by Chrome).

May 18th, PaymentRequest will be likely shipped its support with Chrome in coming keynote “The future of web payments” by Zach Koch, one of proposer for web payment standardization

MALAYSIA TIME: 18th May 11:30PM

The future of web payments



Thu. May 18, 8:30AM - 9:30AM

Stage 6

Content level: Intermediate

Creating a pain-free checkout experience on the web doesn't have to be hard. Come learn how we're saving the world from annoying checkout forms and driving mobile conversions with the new Payment Request API. We'll cover how you can use the Payment Request API to build a best-in-class checkout experience as well how third party payment providers can integrate directly into this new surface.



Zach Koch
Google



Let's Begin Code



PART 1: Payment Methods

Tell which payment methods (VISA, Android Pay, Credit Cards, AliPay, Paypal and etc.)

// Supported payment methods

```
var methodData = [{  
  supportedMethods: ['basic-card']  
  data: {  
    merchantID: '3123123133',  
    supportedNetworks: [  
      'visa', 'mastercard', 'amex', 'discover',  
      'diners', 'jcb', 'unionpay'  
    ]  
  }  
}];
```



PART 2: TRANSACTION INFO

How much you want to pay, which currency, details of list items of purchase

// Checkout details

```
var transactionDetails = {  
  displayItems: [{  
    label: 'Food', amount: { currency: 'MYR', value: '5.00' }  
  }, {  
    label: 'Company Sponsor', amount: { currency: 'MYR', value: '-5.00' }  
  }],  
  total: {  
    label: 'Total due', amount: { currency: 'MYR', value : '0.00' }  
  }  
};
```

NOTE
PaymentRequest
knows no math

1 + 1 = 3? Possibly



PART 2 EXTRA : TRANSACTION DETAILS

Shipping options can be displayed by default

```
// ShippingOptions

...

total: {

  label: 'Total due', amount: { currency: 'MYR', value : '0.00' }

}

shippingOptions: [{

  Id:'free', label:'FREE SHIPPING', amount:'MYR', selected: true

}]

};
```



PART 3: EXTRA INFO

- User Address
- Shipping Support
- Name
- Email
- Phone
- Shipping Type such as COD, pickup, shipping

Optional but useful

```
// Checkout details  
var options = {  
    requestShipping: true,  
    requestPayerPhone: true,  
    requestPayerName: true,  
    shippingType: 'shipping'  
};
```



Done setup, now begin the process !



1. Instantiaze Payment Request

// 1. Create a `PaymentRequest` instance

```
var request = new PaymentRequest(methodData, transactionDetails, options);  
request.addEventListener('shippingaddresschange', function(evt){  
    evt.updateWith(Promise.resolve(newTransactionDetails));  
});
```



2. Show the native UI with `.show()` &

```
request.show().then(result => {  
  // POST the payment information to the server  
  return fetch('/pay', {  
    method: 'POST',  
    credentials: 'include',  
    headers: {  
      'Content-Type': 'application/json'  
    },  
    body: JSON.stringify(result.toJSON())  
  })  
  ...  
})
```



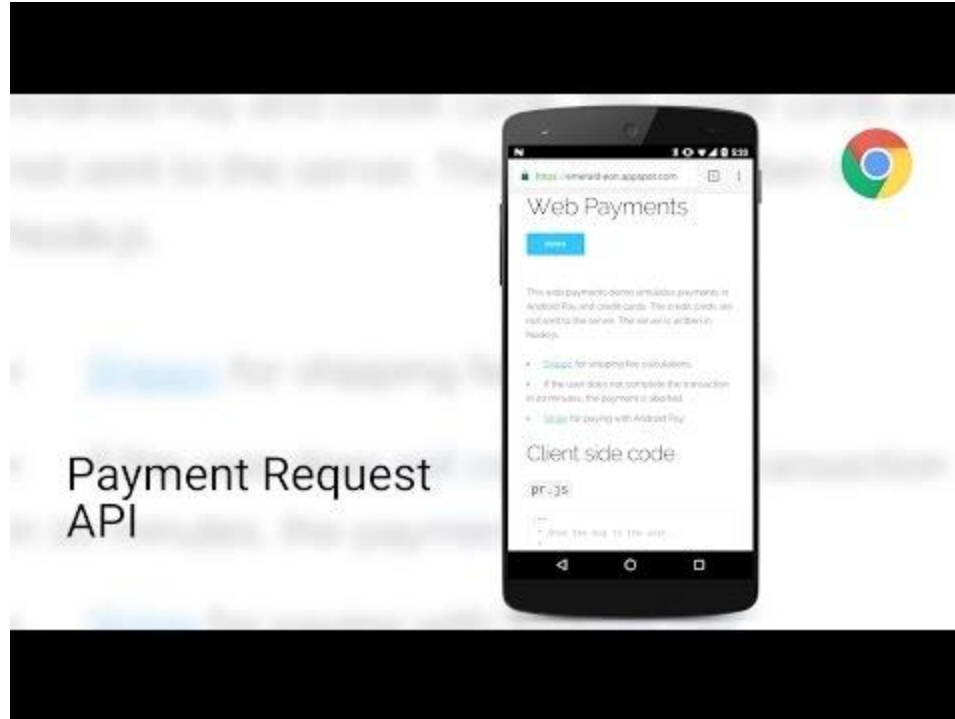
4. DISPLAY PAYMENT RESULT

...

```
}).then(response => {  
    // 4. Display payment results  
    if (response.status === 200) {  
        // Payment successful  
        return result.complete('success');  
    } else {  
        // Payment failure  
        return result.complete('fail');  
    }  
}).catch(() => {  
    return result.complete('fail');  
});
```



SUMMARY OF PAYMENTREQUEST FLOW



REFERENCES:

APPSPOT DEMO (MOBILE) : <https://emerald-eon.appspot.com/> OR
<https://goo.gl/Q5uZ9g>

MOSC + SLIDES : <http://lanyrd.com/2017/moscmysfrgxd/#link-cfydz>

W3C DOCS : <https://w3c.github.io/browser-payment-api/>

W3C GITHUB : <https://github.com/w3c/browser-payment-api/>

GOOGLE DEVELOPERS PAYMENT REQUEST API :

<https://developers.google.com/web/fundamentals/discovery-and-monetization/payment-request/>



K.I.T

Facebook | SC Kim <https://fb.com/johnny.ksc>

Twitter | @saychun_kim

GooglePlus | +SCKim

Gmail | <mailto:saychunkim@gmail.com>

Portfolio | <https://saychunkim.github.io>



</keynote>

Thank you !

