Check out the future of checkouts

Introduction to the Payment Request API

Christian Ost at BarcelonaJS in June 2019

Web Payments is an emerging web standard […] to simplify online payments and enable a broader set of players to participate easily in the payments ecosystem on the web.

https://developers.google.com/web/fundamentals/payments/

> […] It is not a new way for paying for things; rather, it's a way for users to select their preferred way of paying for things, and make that information available to a merchant.

https://developer.mozilla.org/en-US/docs/Web/API/Payment_Request_API

Why do we need a standard around paying online?

Christian Ost

Developer at Typeform

@_christianost | christianost.de

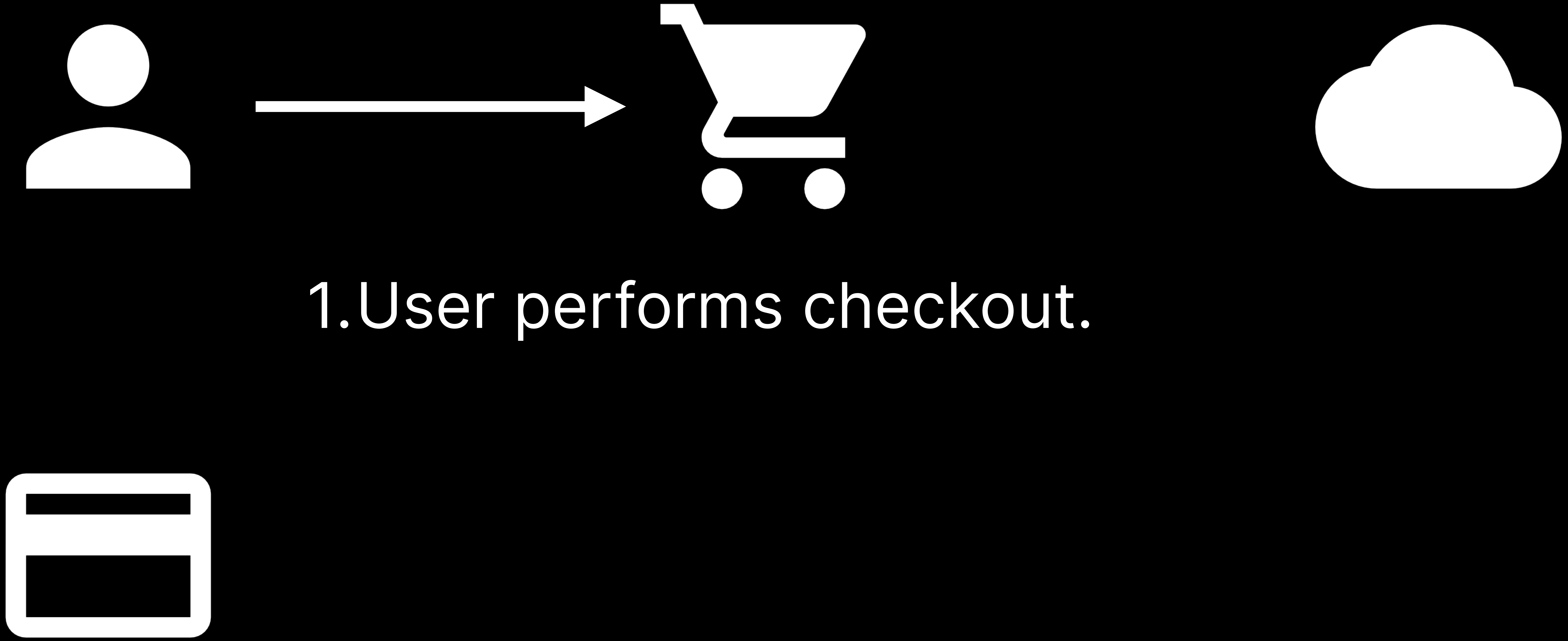
typeform.com

1. How will paying online work?
2. How can I use it?
3. How will it look like?

1. How will paying online work?

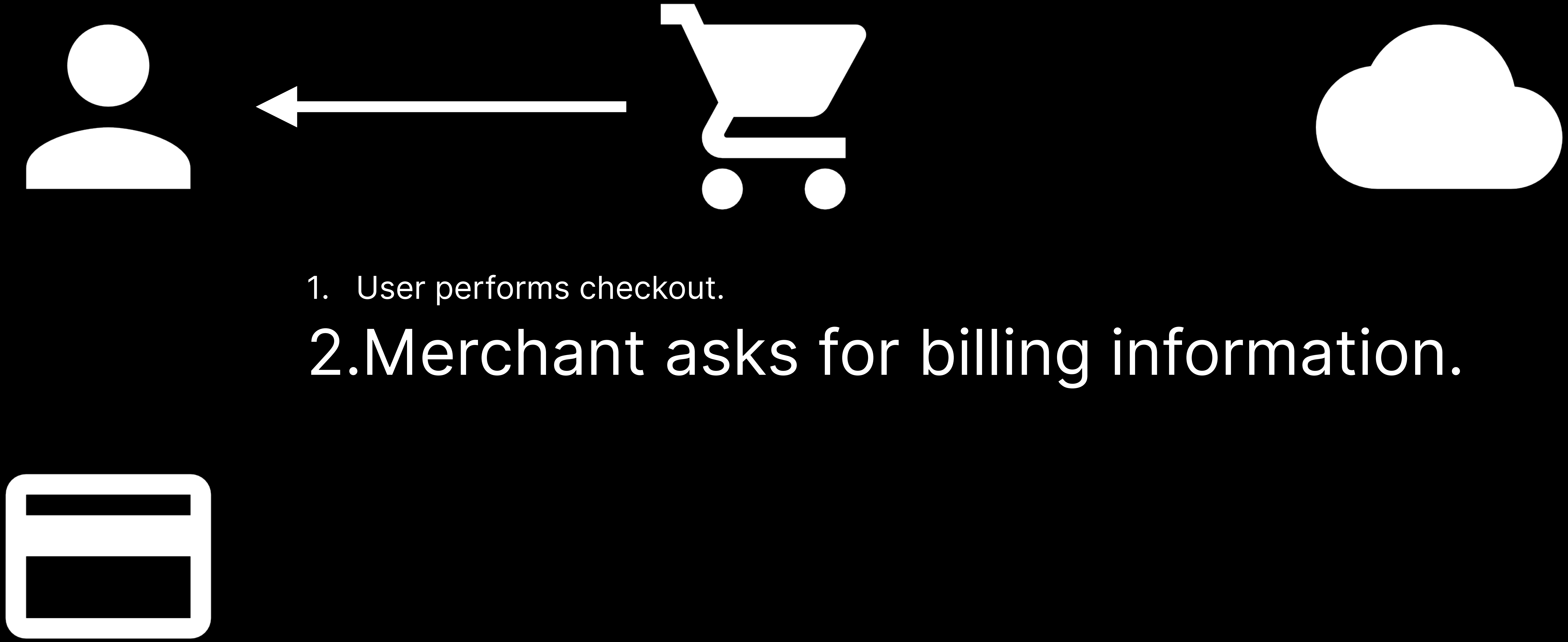
2. How can I use it?
3. How will it look like?

https://developers.google.com/web/fundamentals/payments/



1.User performs checkout.

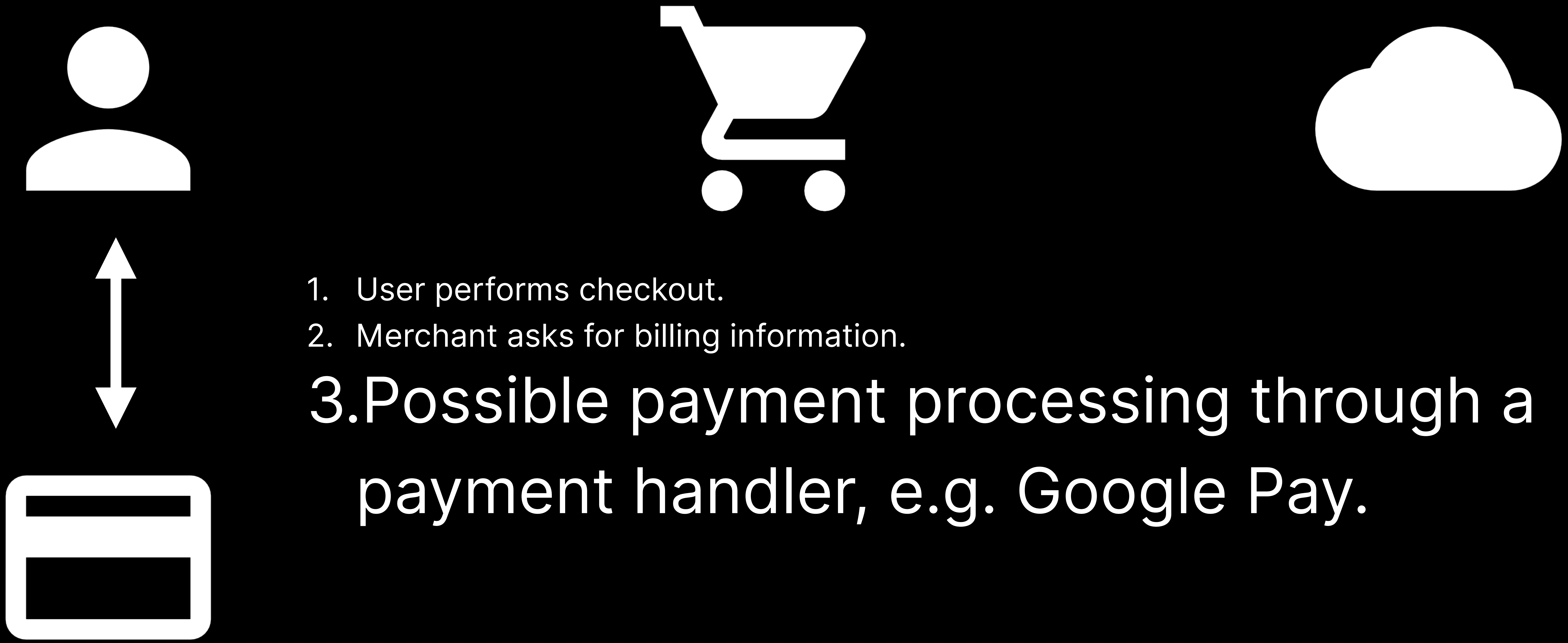
https://developers.google.com/web/fundamentals/payments/



1. User performs checkout.

2. Merchant asks for billing information.

https://developers.google.com/web/fundamentals/payments/



1. User performs checkout.
2. Merchant asks for billing information.
3. Possible payment processing through a payment handler, e.g. Google Pay.

https://developers.google.com/web/fundamentals/payments/

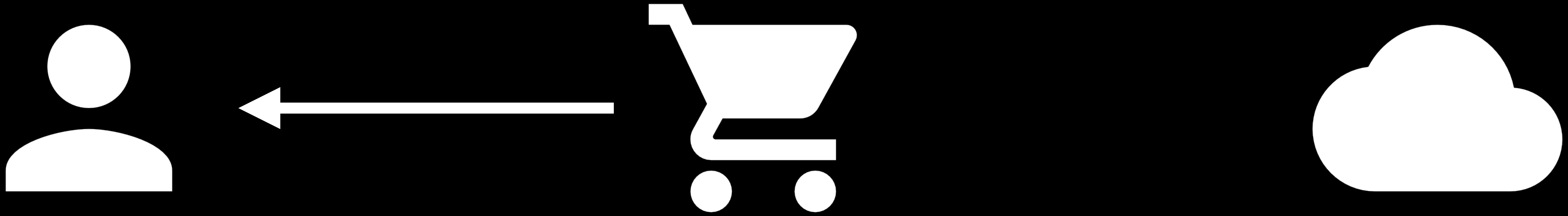
1. User performs checkout.
2. Merchant asks for billing information.
3. Possible payment processing through a payment handler, e.g. Google Pay.

4. Merchant sends payment information to Payment Service Provider.

https://developers.google.com/web/fundamentals/payments/

1. User performs checkout.
2. Merchant asks for billing information.
3. Possible payment processing through a payment handler, e.g. Google Pay.
4. Merchant sends payment information to Payment Service Provider.
5. Payment Service Provider returns outcome.

https://developers.google.com/web/fundamentals/payments/



1. User performs checkout.
2. Merchant asks for billing information.
3. Possible payment processing through a payment handler, e.g. Google Pay.
4. Merchant sends payment information to Payment Service Provider.
5. Payment Service Provider returns outcome.
6. Success or failure of the payment is returned to user.

https://developers.google.com/web/fundamentals/payments/

1. User performs checkout.
2. Merchant asks for billing information.
3. Possible payment processing through a payment handler, e.g. Google Pay.
4. Merchant sends payment information to Payment Service Provider.
5. Payment Service Provider returns outcome.
6. Success or failure of the payment is returned to user.

https://developers.google.com/web/fundamentals/payments/

To enable this flow, the Web Payment Ecosystem is built on top of multiple APIs.

Web Payment Ecosystem

- Payment Request API (native checkout flow)
- Payment Handler API (payment handling by 3rd parties)
- Payment Method Identifiers (e.g. `basic-card`)
- Payment Method Manifest (describes how payments are handled)

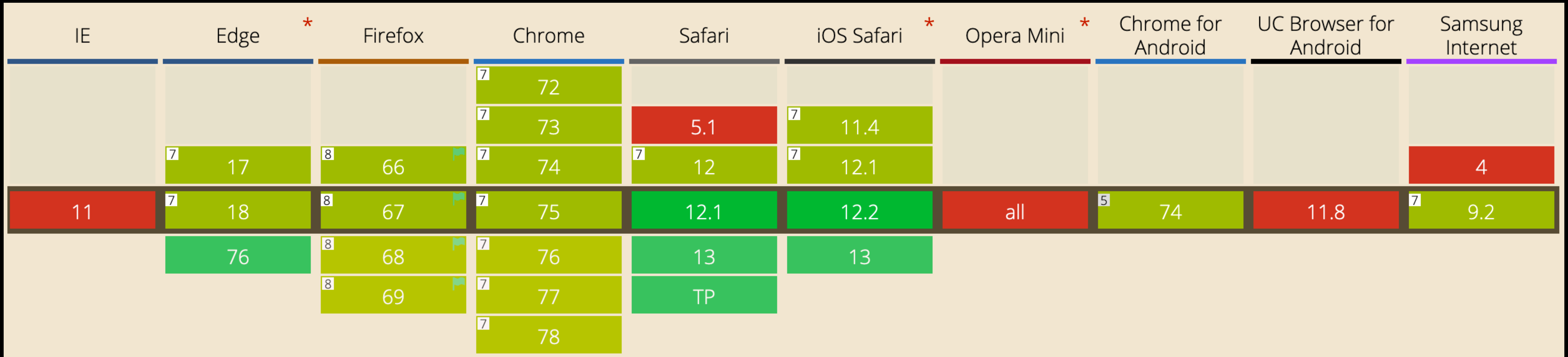
Web Payment Ecosystem

- Payment Request API (native checkout flow)
- Payment Handler API (payment handling by 3rd parties)
- Payment Method Identifiers (e.g. `basic-card`)
- Payment Method Manifest (describes how payments are handled)

1. How will paying online work?

2. **How can I use it?**

3. How will it look like?



IE	Edge *	Firefox	Chrome	Safari	iOS Safari *	Opera Mini *	Chrome for Android	UC Browser for Android	Samsung Internet
			[7] 72						
			[7] 73	5.1	[7] 11.4				
	[7] 17	[8] 66 ⚑	[7] 74	[7] 12	[7] 12.1				4
11	[7] 18	[8] 67 ⚑	[7] 75	12.1	12.2	all	[5] 74	11.8	[7] 9.2
	76	[8] 68 ⚑	[7] 76	13	13				
		[8] 69 ⚑	[7] 77	TP					
			[7] 78						

Basic Payment Request

```
if (!window.PaymentRequest) return;

const paymentRequest = new PaymentRequest(
  methodData,
  details,
  options
);
```

Payment methods data

```
// new PaymentRequest(methodData, details, options)

const methodData = [
  {
    supportedMethods: ‘basic-card',
    data: {
      supportedNetworks: ['visa', 'mastercard', ‘amex'],
      supportedTypes: ['debit', 'credit']
    }
  },
  {
    supportedMethods: 'https://apple.com/apple-pay',
    data: { /* dictionary configuring ApplePay integration */ }
  }
]
```

Available payment method identifiers

- basic-card
- basic-card-transfer
  tokenized-card
  interledger
  sepamail

- https://google.com/pay
  https://apple.com/apple-pay
  https://spay.samsung.com

see https://w3c.github.io/payment-method-id

Payment details

```
// new PaymentRequest(methodData, details, options)

const details = {
  displayItems: [
    {
      label: 'seggsy shirt',
      amount: { currency: 'EUR', value: '29.99' },
    },
    {
      label: 'seggsy hat',
      amount: { currency: 'EUR', value: '59.99' },
    },
  ],
  total: {
    label: 'Total',
    amount: { currency: 'EUR', value: '89.98' },
  },
};
```

Payment options

```
// new PaymentRequest(methodData, details, options)

const options = {
  requestShipping: true,
  requestPayerEmail: true,
  requestPayerPhone: true,
  requestPayerName: true,
  shippingType: 'shipping', // or 'delivery' or 'pickup'
};
```

1. How will paying online work?
2. How can I use it?
3. How will it look like?

Executing a payment request

```
try {
  const paymentRequest = new PaymentRequest(/* ... */)
  if (!await paymentRequest.canMakePayment()) {
    // consider showing an alternative payment flow
  }

  const response = await paymentRequest.show()

  // user accepted the payment, handle it

  response.complete('success') // or 'fail'
} catch(error) {
  // handle error
}
```

Live demonstration

Resources

https://developers.google.com/web/fundamentals/payments/

https://developer.mozilla.org/en-US/docs/Web/API/Payment_Request_API

https://www.w3.org/Payments/WG/

Thank you for listening.

Find a link to the slides on https://christianost.de.