

Using a Static Site for eCommerce

Web Engineering DUS Meetup

About Me



Marcus Weiner

@mraerino

Freelance Web Engineer

Shameless plug

KALKSPACE

- Coworking in Cologne
- Non-commercial
- Right side of the Rhine - *Schäl Sick*
- kalk.space

**Is it possible to build a web shop without a
product database?**

Static Sites

JAMStack

Have you ever built a static site?

Marina

Marina

→ 30 years old

→ Mother

→ Blogging

Marina

- Built her blog using Hugo
- Knows some HTML, CSS, Javascript
- Wants to sell some ebooks & merch

eCommerce API

POST /orders

```
{
  "items": [{
    sku: "my-fantastic-ebook",
    amount: 2
  }, {
    sku: "a-very-special-ebook",
    amount: 1
  }],
  "email": "first-customer@example.com",
}
```

200 OK

```
{  
  order_id: "1f9d1e1b-6e66-4bb3-a280-6783f20d64d2",  
  payment_status: "pending",  
  amount: 2000 // 20.00 €  
}
```

```
<- POST /payments
```

```
{  
  order_id: "1f9d1e1b-6e66-4bb3-a280-6783f20d64d2",  
  stripe_token: "54922b22-5176-11e9-8647-d663bd873d93"  
}
```

```
-> 200 OK
```

Now some DreamCode

Ordering

```
export const order = (cart, email) => {  
  const order = await shop.order({  
    items: cart,  
    email,  
  })  
  const stripeToken = await frontend.askCreditCard(order.amount);  
  await shop.pay({  
    orderId: order.id,  
    paymentProvider: 'stripe',  
    stripeToken,  
  })  
}
```

Implementation

Calculating Prices?

/shop/my-ebooks

```
<script class="shop-product" type="application/json">
{
  "sku": "my-fantastic-ebook",
  "price": 500, // 5€
}
</script>
```

POST /orders

```
{  
  "items": [  
    {  
      sku: "my-fantastic-ebook",  
      path: "/shop/my-ebooks",  
      amount: 2  
    }, {  
      sku: "a-very-special-ebook",  
      path: "/shop/my-ebooks",  
      amount: 1  
    }  
  ],  
  "email": "first-customer@example.com",  
}
```



Moar Features!

Admin Interface

- Generic
- Frontend App
- Uses the API

User Authentication

→ Uses JWTs!

→ Users are created on first order

Downloads

```
<script class="shop-product" type="application/json">
{
  "sku": "my-fantastic-ebook",
  "price": 500, // 5€
  "downloads": ["https://example.com/ebook"],
}
</script>
```

```
const order = await shop.getOrder(orderId)
const downloads = await order.getDownloads()
assert.equals(
  downloads[0].url,
  "https://example.com/ebook?token=eyJhbGciOi..."
)
```

Stock

```
<script class="shop-product" type="application/json">
{
  "sku": "awesome-t-shirt",
  "price": 1500, // 15€
  "stock": {
    "amount": 76,
    "last_updated": "2019-03-22T15:35:55Z",
  }
}
</script>
```




Extensibility

Cloud Functions

Using the API

Webhooks

Large Enterprises?

Any Product Storage

→ legacy ERP?

→ single source of thruth

Any Website Stack

- Developers can focus on website experience
 - Static workflows scale to large teams

GoCommerce

<https://github.com/netlify/gocommerce>

Thank you!

Ask me anything!