

# Development part-2

Cloud Application Development - E-commerce Application on IBM  
Cloud Foundry

## Backend Development Overview

E-commerce websites require backend development to function properly. The backend is the data access layer that holds products, orders, and customer information. It helps websites operate by processing, storing, and transferring their data to and from the customer-facing side. Backend development is a complex process that requires careful consideration of several key components and functionalities. Here is an overview of the essential aspects of e-commerce backend development:

**Compatibility:** All parts of the tech stack, including databases, applications, microservices, plug-ins, and the frontend, should work well together to create a successful e-commerce platform.

**Scalability:** The ability to add or subtract resources without significant costs enables efficient growth.

**Efficiency:** Optimizing the tech stack and considering the time and resources needed to build the platform is essential to get the most out of your technology budget.

**Frugality:** Standing up a robust backend can be expensive, so it's important to consider where your budget will get the best return on investment.

**Security:** Security is always vital, especially if you're storing customer data. Meeting the highest e-commerce standards is crucial to avoid disastrous consequences.

Developers can use various technologies to build an e-commerce backend, including SQL-based database management systems, node JS, and PHP. Backend providers like Back4app, Firebase, and Parse offer highly flexible solutions and amazing features to make the e-commerce app development process easier. Frontend development ensures that an e-commerce website is functional and visually appealing by prioritizing the user experience. Frontend developers use programming languages like JavaScript, HTML, and CSS to create and manage a website's frontend. Understanding the difference between the e-commerce backend and frontend and how they work together can help you design a seamless omnichannel shopping experience for your customers.

## BackEnd Using Tools :

- JAVASCRIPT
- REACT JS

## Code :

### HTML code :

```
<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```

<meta http-equiv="X-UA-Compatible" content="ie=edge">

<title> CONTENT DETAILS | E-COMMERCE WEBSITE BY EDYODA </title>

<!-- favicon -->

<link rel="icon"
href="https://yt3.ggpht.com/a/AGF-l78km1YyNXmF0r3-0CycCA0HLA_i6zYn_8NZEg=s
900-c-k-c0xffffffff-no-rj-mo" type="image/gif" sizes="16x16">

<!-- <link rel="stylesheet" href="/box1.css"> -->

<link rel="stylesheet" href="css/contetDetails.css">

<link href="https://fonts.googleapis.com/css?family=Lato&display=swap"
rel="stylesheet">

<!-- header links -->

<script src="https://kit.fontawesome.com/4a3b1f73a2.js"></script>

<link href="https://fonts.googleapis.com/css?family=Lato&display=swap"
rel="stylesheet">

</head>

<body>

<!-- HEADER -->

<div id="1"></div>

<script>

load("header.html");

function load(url)
{

```

```
req = new XMLHttpRequest();
req.open("GET", url, false);
req.send(null);

document.getElementById(1).innerHTML = req.responseText;
}

</script>

<div id="containerProduct">

<!-- JS rendered code -->

</div>

<script src="/contentDetails.js"></script>

<!-- FOOTER -->

<div id="4"></div>

<script>

load("footer.html");

function load(url)
{
req = new XMLHttpRequest();
req.open("GET", url, false);
req.send(null);

document.getElementById(4).innerHTML = req.responseText;
}
```

```
</script>

</body>

</html>
```

## JS code :

```
console.clear()

let id = location.search.split('?')[1]
console.log(id)

if (document.cookie.indexOf(',counter=') >= 0)
{
let counter = document.cookie.split(',')[1].split('=')[1]
document.getElementById("badge").innerHTML = counter
}

function dynamicContentDetails(ob)
{
let mainContainer = document.createElement('div')
mainContainer.id = 'containerD'
document.getElementById('containerProduct').appendChild(mainContainer);
```

```
let imageSectionDiv = document.createElement('div')

imageSectionDiv.id = 'imageSection'

let imgTag = document.createElement('img')

imgTag.id = 'imgDetails'

//imgTag.id = ob.photos

imgTag.src = ob.preview

imageSectionDiv.appendChild(imgTag)

let productDetailsDiv = document.createElement('div')

productDetailsDiv.id = 'productDetails'

// console.log(productDetailsDiv);

let h1 = document.createElement('h1')

let h1Text = document.createTextNode(ob.name)

h1.appendChild(h1Text)

let h4 = document.createElement('h4')

let h4Text = document.createTextNode(ob.brand)

h4.appendChild(h4Text)
```

```
console.log(h4);

let detailsDiv = document.createElement('div')
detailsDiv.id = 'details'

let h3DetailsDiv = document.createElement('h3')
let h3DetailsText = document.createTextNode('Rs ' + ob.price)
h3DetailsDiv.appendChild(h3DetailsText)

let h3 = document.createElement('h3')
let h3Text = document.createTextNode('Description')
h3.appendChild(h3Text)

let para = document.createElement('p')
let paraText = document.createTextNode(ob.description)
para.appendChild(paraText)

let productPreviewDiv = document.createElement('div')
productPreviewDiv.id = 'productPreview'

let h3ProductPreviewDiv = document.createElement('h3')
let h3ProductPreviewText = document.createTextNode('Product Preview')
```

```
h3ProductPreviewDiv.appendChild(h3ProductPreviewText)

productPreviewDiv.appendChild(h3ProductPreviewDiv)

let i;

for(i=0; i<ob.photos.length; i++)
{
let imgTagProductPreviewDiv = document.createElement('img')

imgTagProductPreviewDiv.id = 'previewImg'

imgTagProductPreviewDiv.src = ob.photos[i]

imgTagProductPreviewDiv.onclick = function(event)
{
console.log("clicked" + this.src)

imgTag.src = ob.photos[i]

document.getElementById("imgDetails").src = this.src
}

productPreviewDiv.appendChild(imgTagProductPreviewDiv)
}

let buttonDiv = document.createElement('div')

buttonDiv.id = 'button'

let buttonTag = document.createElement('button')
```



```
buttonDiv.appendChild(buttonTag)

buttonText = document.createTextNode('Add to Cart')

buttonTag.onclick = function()
{
    let order = id+" "

    let counter = 1

    if(document.cookie.indexOf(',counter=')>=0)
    {
        order = id + " " + document.cookie.split(',')[0].split('=')[1]

        counter = Number(document.cookie.split(',')[1].split('=')[1]) + 1
    }

    document.cookie = "orderId=" + order + ",counter=" + counter

    document.getElementById("badge").innerHTML = counter

    console.log(document.cookie)
}

buttonTag.appendChild(buttonText)


console.log(mainContainer.appendChild(imageSectionDiv));

mainContainer.appendChild(imageSectionDiv)

mainContainer.appendChild(productDetailsDiv)
```

```
productDetailsDiv.appendChild(h1)

productDetailsDiv.appendChild(h4)

productDetailsDiv.appendChild(detailsDiv)

detailsDiv.appendChild(h3DetailsDiv)

detailsDiv.appendChild(h3)

detailsDiv.appendChild(para)

productDetailsDiv.appendChild(productPreviewDiv)

productDetailsDiv.appendChild(buttonDiv)


return mainContainer
}


// BACKEND CALLING


let httpRequest = new XMLHttpRequest()

{

httpRequest.onreadystatechange = function()

{

if(this.readyState === 4 && this.status == 200)
```

```
{
  console.log('connected!!');

  let contentDetails = JSON.parse(this.responseText)

  {
    console.log(contentDetails);

    dynamicContentDetails(contentDetails)
  }
}

else
{
  console.log('not connected!');
}

}

}

httpRequest.open('GET',
'https://5d76bf96515d1a0014085cf9.mockapi.io/product/'+id, true)

httpRequest.send()
```

## Output :


UCEReditZs

CLOTHING ACCESSORIES

Q

Search for Clothing and Accessories

0



### Men Black MAMGP T7 Sweat Sporty Jacket




Puma

Rs 7999

**Description**

Black Solid Sporty Jacket, Has A Mock Collar, 3 Pockets, Zip Closure, Long Sleeves, Straight Hem, Unlined Lining

**Product Preview**



Add to Cart


UCEReditZs

CLOTHING ACCESSORIES

Q

Search for Clothing and Accessories

2



### Men Black MAMGP T7 Sweat Sporty Jacket




Puma

Rs 7999

**Description**

Black Solid Sporty Jacket, Has A Mock Collar, 3 Pockets, Zip Closure, Long Sleeves, Straight Hem, Unlined Lining

**Product Preview**



Add to Cart



## Unisex Black Galaxy Fit Fitness Band

Samsung

Rs 9990

### Description

Keep Up Your Routine For Even Longer. With Plenty Of Power To Last Throughout Your Daily Workouts, You Can Pack In Those Longer Hours To Reach Your Fitness Goals. What's More, You Won't Have To Worry About Disruptive Battery Discharge When Tracking Your Activity.

### Product Preview



Add to Cart

# Thank you !