

Phase 4
Development Part 2

E-Commerce Application on IBM Cloud Foundry



Introduction about Phase 4

In this part we will continue building our project.

Continue building the e-commerce platform by implementing user authentication, shopping cart, and checkout functionality. Implement user registration and authentication features using a backend server (e.g. Node.js). Implement shopping cart functionality, calculate the total, and enable a smooth checkout process.

HTML Script for User Registration and Authentication

Front-end (HTML, CSS, JavaScript):

```
html
<!DOCTYPE html>
<html>
<head>
  <title>User Registration and Authentication</title>
</head>
<body>
  <h1>User Registration</h1>
  <form id="registrationForm">
    <label for="username">Username:</label>
    <input type="text" id="username" required><br>
    <label for="password">Password:</label>
    <input type="password" id="password" required><br>
    <button type="submit">Register</button>
  </form>

  <h1>User Login</h1>
  <form id="loginForm">
    <label for="loginUsername">Username:</label>
    <input type="text" id="loginUsername" required><br>
    <label for="loginPassword">Password:</label>
    <input type="password" id="loginPassword" required><br>
    <button type="submit">Login</button>
  </form>

  <p id="message"></p>

  <script src="script.js"></script>
</body>
</html>
```

Backend (Node.js)

```
javascript
const express = require('express');
```

```
const mongoose = require('mongoose');
const bcrypt = require('bcrypt');
const app = express();
const port = 3000;

mongoose.connect('mongodb://localhost/ecommerce_db', { useNewUrlParser: true,
useUnifiedTopology: true });

const User = mongoose.model('User', {
  username: String,
  password: String
});

app.use(express.json());

app.post('/register', async (req, res) => {
  const { username, password } = req.body;
  const hashedPassword = await bcrypt.hash(password, 10);

  const user = new User({
    username,
    password: hashedPassword
  });
  user.save()
    .then(() => {
      res.send('User registered successfully');
    })
    .catch(error => {
      res.status(500).send('Error registering user');
    });
});

app.post('/login', async (req, res) => {
  const { username, password } = req.body;

  const user = await User.findOne({ username });

  if (user && await bcrypt.compare(password, user.password)) {
    res.send('Login successful');
  } else {
    res.status(401).send('Login failed');
  }
});

app.listen(port, () => {
  console.log(`Server is running on port ${port}`);
});
```

OUTPUT

User Registration

Username:
Password:

User Login

Username:
Password:

HTML and CSS for check out page

Html Script

```
html
<!DOCTYPE html>
<html>
<head>
  <title>E-commerce Checkout</title>
  <link rel="stylesheet" type="text/css" href="styles.css">
</head>
<body>
  <div class="checkout-container">
    <h2>Checkout</h2>
    <form>
      <label for="name">Name</label>
      <input type="text" id="name" name="name" required>

      <label for="email">Email</label>
      <input type="email" id="email" name="email" required>

      <label for="address">Shipping Address</label>
      <textarea id="address" name="address" required></textarea>

      <label for="credit-card">Credit Card</label>
      <input type="text" id="credit-card" name="credit-card" required>

      <label for="expiration-date">Expiration Date</label>
      <input type="text" id="expiration-date" name="expiration-date"
required>

      <label for="cvv">CVV</label>
```

```

        <input type="text" id="cvv" name="cvv" required>

        <input type="submit" value="Place Order">
    </form>
</div>
</body>
</html>

```

CSS Script

```

css
body {
    font-family: Arial, sans-serif;
    background-color: #f2f2f2;
    margin: 0;
    padding: 0;
}

.checkout-container {
    max-width: 400px;
    margin: 0 auto;
    background-color: #fff;
    padding: 20px;
    border: 1px solid #ddd;
    border-radius: 5px;
}

form {
    display: flex;
    flex-direction: column;
}

label {
    font-weight: bold;
}

input[type="text"],
input[type="email"],
textarea {
    width: 100%;
    padding: 10px;
    margin-bottom: 10px;
    border: 1px solid #ddd;
    border-radius: 5px;
}

input[type="submit"] {
    background-color: #007bff;
    color: #fff;
    padding: 10px 20px;
}

```

```
border: none;
border-radius: 5px;
cursor: pointer;
}

input[type="submit"]:hover {
  background-color: #0056b3;
}
```

OUTPUT

Checkout

Name	<input type="text"/>	Email	<input type="text"/>	Shipping Address	<input type="text"/>	Credit Card	<input type="text"/>	Expiration Date	<input type="text"/>
	<input type="text"/>	CVV	<input type="text"/>	<input type="button" value="Place Order"/>					