**NANMUDHALVAN**

**-** **CAD101 Cloud Application Development - Group 1**

**COLLEGE NAME: JEPPIAAR INSTITUTE OF TECHNOLOGY**

**TITLE: 2106-E-commerece Application on IBM Cloud Foundry**

**TEAM NAME: Proj\_228508\_Team\_2**

**TEAM MEMBERS:**

1. **S.Jagadesh**
2. **S.Ihsan Ahamed**
3. **R.Aravindhan**
4. **S.Guhan**
5. **P.Manoj**

**Phase 4: Development Part 2**

In this part you will continue building your 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, Python).
* Implement shopping cart functionality, calculate the total, and enable a smooth checkout process

**OBJECTIVIES:**

In this technology projects you will continue building your project using IBM Cloud Foundry. Perform different functions as per project requirement. After performing the relevant activities create a document around it and share the same for assessment.

**USER AUTHENTICATION**

pip install Flask Flask-Login

**IMPLEMENTATON OF USER AUTHENTICATION**

from flask import Flask, render\_template, redirect, url\_for, request

from flask\_login import LoginManager, UserMixin, login\_user, login\_required, logout\_user, current\_user

app = Flask(\_\_name\_\_)

app.secret\_key = 'your\_secret\_key' # Change this to a strong, random secret key

login\_manager = LoginManager()

login\_manager.init\_app(app)

# Mock user data for demonstration

users = {'user1': {'password': 'password1'}, 'user2': {'password': 'password2'}}

class User(UserMixin):

def \_\_init\_\_(self, id):

self.id = id

@login\_manager.user\_loader

def load\_user(user\_id):

return User(user\_id)

@app.route('/login', methods=['GET', 'POST'])

def login():

if request.method == 'POST':

username = request.form['username']

password = request.form['password']

if users.get(username) and users[username]['password'] == password:

user = User(username)

login\_user(user)

return redirect(url\_for('dashboard'))

return render\_template('login.html')

@app.route('/logout')

@login\_required

def logout():

logout\_user()

return redirect(url\_for('login'))

@app.route('/dashboard')

@login\_required

def dashboard():

return f'Welcome, {current\_user.id}! This is your dashboard.'

if \_\_name\_\_ == '\_\_main\_\_':

app.run(debug=True)

**SHOPPING CART**

**Setup flask application**

from flask import Flask, render\_template, request, session

app = Flask(\_\_name\_\_)

app.secret\_key = 'your\_secret\_key' # Change this to a strong, random secret key

# Sample products (for demonstration)

products = [

{'id': 1, 'name': 'Product 1', 'price': 10},

{'id': 2, 'name': 'Product 2', 'price': 20},

{'id': 3, 'name': 'Product 3', 'price': 30}

]

@app.route('/')

def index():

return render\_template('index.html', products=products)

@app.route('/add\_to\_cart', methods=['POST'])

def add\_to\_cart():

product\_id = int(request.form['product\_id'])

product = next((p for p in products if p['id'] == product\_id), None)

if product:

cart = session.get('cart', [])

cart.append(product)

session['cart'] = cart

return redirect('/')

@app.route('/cart')

def view\_cart():

cart = session.get('cart', [])

total\_price = sum(item['price'] for item in cart)

return render\_template('cart.html', cart=cart, total\_price=total\_price)

if \_\_name\_\_ == '\_\_main\_\_':

app.run(debug=True)

**create html template**

**1.index.html**

<!DOCTYPE html>

<html>

<head>

<title>Shopping Cart</title>

</head>

<body>

<h1>Products</h1>

<ul>

{% for product in products %}

<li>

{{ product.name }} - ${{ product.price }}

<form action="/add\_to\_cart" method="post">

<input type="hidden" name="product\_id" value="{{ product.id }}">

<input type="submit" value="Add to Cart">

</form>

</li>

{% endfor %}

</ul>

</body>

</html>

**2.cart.html**

<!DOCTYPE html>

<html>

<head>

<title>Shopping Cart</title>

</head>

<body>

<h1>Shopping Cart</h1>

<ul>

{% for item in cart %}

<li>{{ item.name }} - ${{ item.price }}</li>

{% endfor %}

</ul>

<p>Total Price: ${{ total\_price }}</p>

</body>

</html>

**CHECKOUT FUNCTIONALITIES**

**update Flask app**

from flask import Flask, render\_template, request, session, redirect, url\_for

app = Flask(\_\_name\_\_)

app.secret\_key = 'your\_secret\_key'

# Mock products for demonstration

products = [

{'id': 1, 'name': 'Product 1', 'price': 10},

{'id': 2, 'name': 'Product 2', 'price': 20},

{'id': 3, 'name': 'Product 3', 'price': 30}

]

@app.route('/')

def index():

return render\_template('index.html', products=products)

@app.route('/add\_to\_cart', methods=['POST'])

def add\_to\_cart():

product\_id = int(request.form['product\_id'])

product = next((p for p in products if p['id'] == product\_id), None)

if product:

cart = session.get('cart', [])

cart.append(product)

session['cart'] = cart

return redirect('/')

@app.route('/cart')

def view\_cart():

cart = session.get('cart', [])

total\_price = sum(item['price'] for item in cart)

return render\_template('cart.html', cart=cart, total\_price=total\_price)

@app.route('/checkout', methods=['GET', 'POST'])

def checkout():

if request.method == 'POST':

# Process payment (mocked)

cart = session.get('cart', [])

total\_price = sum(item['price'] for item in cart)

session.pop('cart', None) # Clear the cart after checkout

return render\_template('confirmation.html', total\_price=total\_price)

return render\_template('checkout.html')

if \_\_name\_\_ == '\_\_main\_\_':

app.run(debug=True)

**create html templates**

**(i)checkout.html**

<!DOCTYPE html>

<html>

<head>

<title>Checkout</title>

</head>

<body>

<h1>Checkout</h1>

<form action="/checkout" method="post">

<label for="name">Name:</label>

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

<label for="address">Address:</label>

<input type="text" id="address" name="address" required><br><br>

<input type="submit" value="Submit Order">

</form>

</body>

</html>

**(ii)confirmation.html**

<!DOCTYPE html>

<html>

<head>

<title>Order Confirmation</title>

</head>

<body>

<h1>Order Confirmation</h1>

<p>Your order has been placed successfully!</p>

<p>Total Price: ${{ total\_price }}</p>

</body>

</html>

**USER REGISTRATION(BACK END)**

**Update Flask app**

from flask import Flask, render\_template, request, redirect, url\_for, flash

app = Flask(\_\_name\_\_)

app.secret\_key = 'your\_secret\_key'

# Mock user data (for demonstration)

users = {}

@app.route('/')

def index():

return render\_template('index.html')

@app.route('/register', methods=['GET', 'POST'])

def register():

if request.method == 'POST':

username = request.form['username']

password = request.form['password']

if username in users:

flash('Username already exists. Please choose a different username.', 'error')

else:

users[username] = {'password': password}

flash('Registration successful! Please log in.', 'success')

return redirect(url\_for('login'))

return render\_template('register.html')

@app.route('/login')

def login():

return render\_template('login.html')

if \_\_name\_\_ == '\_\_main\_\_':

app.run(debug=True)

**create html template**

**(i)register.html**

<!DOCTYPE html>

<html>

<head>

<title>Register</title>

</head>

<body>

<h1>Register</h1>

{% with messages = get\_flashed\_messages() %}

{% if messages %}

<ul>

{% for message in messages %}

<li>{{ message }}</li>

{% endfor %}

</ul>

{% endif %}

{% endwith %}

<form action="/register" method="post">

<label for="username">Username:</label>

<input type="text" id="username" name="username" required><br><br>

<label for="password">Password:</label>

<input type="password" id="password" name="password" required><br><br>

<input type="submit" value="Register">

</form>

</body>

</html>

**(iii)login.html**

<!DOCTYPE html>

<html>

<head>

<title>Login</title>

</head>

<body>

<h1>Login</h1>

<form action="/login" method="post">

<label for="username">Username:</label>

<input type="text" id="username" name="username" required><br><br>

<label for="password">Password:</label>

<input type="password" id="password" name="password" required><br><br>

<input type="submit" value="Login">

</form>

</body>

</html>

**AUTHENTICATIN(BACKEND)**

**Update Flask app**

from flask import Flask, render\_template, request, redirect, url\_for, flash

app = Flask(\_\_name\_\_)

app.secret\_key = 'your\_secret\_key'

# Mock user data (for demonstration)

users = {'user1': {'password': 'password1'}, 'user2': {'password': 'password2'}}

@app.route('/')

def index():

return render\_template('index.html')

@app.route('/login', methods=['POST'])

def login():

username = request.form['username']

password = request.form['password']

if users.get(username) and users[username]['password'] == password:

flash('Login successful!', 'success')

return redirect(url\_for('dashboard'))

else:

flash('Invalid username or password. Please try again.', 'error')

return redirect(url\_for('index'))

@app.route('/dashboard')

def dashboard():

return render\_template('dashboard.html')

if \_\_name\_\_ == '\_\_main\_\_':

app.run(debug=True)

**create html template**

**(i)index.html**

<!DOCTYPE html>

<html>

<head>

<title>Login</title>

</head>

<body>

<h1>Login</h1>

{% with messages = get\_flashed\_messages() %}

{% if messages %}

<ul>

{% for message in messages %}

<li>{{ message }}</li>

{% endfor %}

</ul>

{% endif %}

{% endwith %}

<form action="/login" method="post">

<label for="username">Username:</label>

<input type="text" id="username" name="username" required><br><br>

<label for="password">Password:</label>

<input type="password" id="password" name="password" required><br><br>

<input type="submit" value="Login">

</form>

</body>

</html>

**(ii)dashboard.html**

<!DOCTYPE html>

<html>

<head>

<title>Dashboard</title>

</head>

<body>

<h1>Welcome to the Dashboard</h1>

<p>This is a protected page.</p>

</body>

</html>

**SHOPPING CART FUNCTIONALITIES,TOTAL,SMOOTH CHECKOUT PROCESS**

**Update Flask app**

from flask import Flask, render\_template, request, redirect, url\_for, flash

app = Flask(\_\_name\_\_)

app.secret\_key = 'your\_secret\_key'

# Mock products for demonstration

products = [

{'id': 1, 'name': 'Product 1', 'price': 10},

{'id': 2, 'name': 'Product 2', 'price': 20},

{'id': 3, 'name': 'Product 3', 'price': 30}

]

@app.route('/')

def index():

return render\_template('index.html', products=products)

@app.route('/add\_to\_cart', methods=['POST'])

def add\_to\_cart():

product\_id = int(request.form['product\_id'])

product = next((p for p in products if p['id'] == product\_id), None)

if product:

cart = session.get('cart', [])

cart.append(product)

session['cart'] = cart

return redirect('/')

@app.route('/cart')

def view\_cart():

cart = session.get('cart', [])

total\_price = sum(item['price'] for item in cart)

return render\_template('cart.html', cart=cart, total\_price=total\_price)

@app.route('/checkout', methods=['GET', 'POST'])

def checkout():

if request.method == 'POST':

# Process payment (mocked)

cart = session.get('cart', [])

total\_price = sum(item['price'] for item in cart)

session.pop('cart', None) # Clear the cart after checkout

flash('Order placed successfully! Total: $' + str(total\_price), 'success')

return redirect(url\_for('index'))

return render\_template('checkout.html')

if \_\_name\_\_ == '\_\_main\_\_':

app.run(debug=True)

**create html template**

**(i)index.html**

<!-- ... -->

{% for product in products %}

<li>

{{ product.name }} - ${{ product.price }}

<form action="/add\_to\_cart" method="post">

<input type="hidden" name="product\_id" value="{{ product.id }}">

<input type="submit" value="Add to Cart">

</form>

</li>

{% endfor %}

<!-- ... -->

**(ii)cart.html**

<!-- ... -->

<h2>Shopping Cart</h2>

<ul>

{% for item in cart %}

<li>{{ item.name }} - ${{ item.price }}</li>

{% endfor %}

</ul>

<p>Total Price: ${{ total\_price }}</p>

<form action="/checkout" method="post">

<input type="submit" value="Checkout">

</form>

<!-- ... -->

**(iii)checkout.html**

<!-- ... -->

<h2>Checkout</h2>

<form action="/checkout" method="post">

<label for="name">Name:</label>

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

<label for="address">Address:</label>

<input type="text" id="address" name="address" required><br><br>

<input type="submit" value="Submit Order">

</form>

<!-- ... -->