

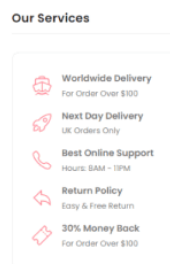
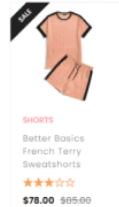
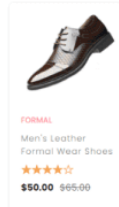
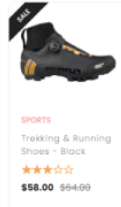
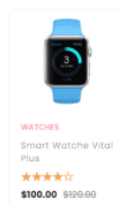
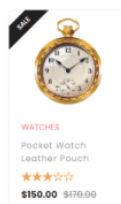
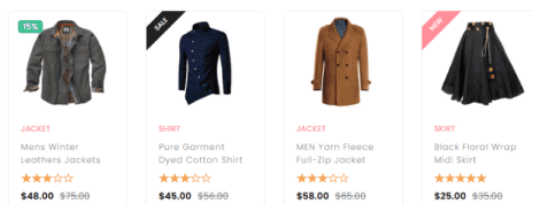
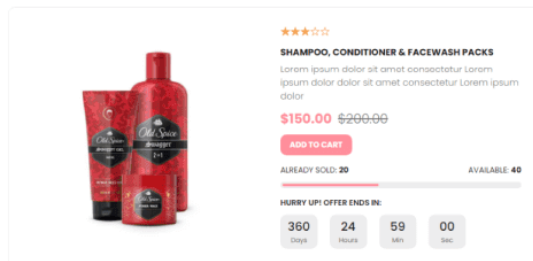
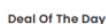
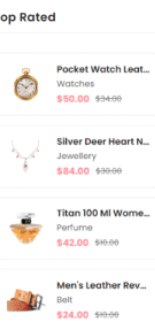
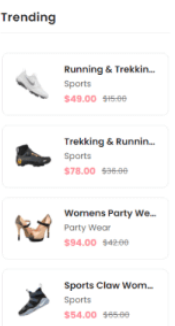
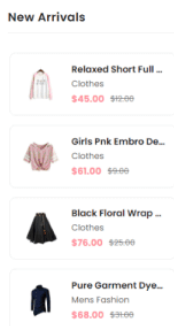
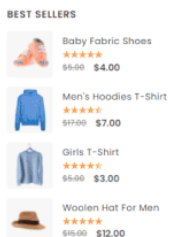
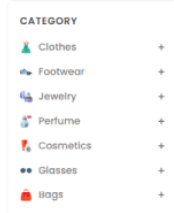
Phase 5 - E-commerce Application Development in IBM Cloud Foundry

Introduction:

Welcome to the comprehensive documentation of our E-commerce application hosted on IBM Cloud Foundry. This document serves as a guide to understanding the core elements of our project and how it harnesses the capabilities of IBM Cloud Foundry to provide a seamless online shopping experience. In this section, we'll provide an overview of the project, highlighting its significance and goals.

Project Overview:

Our E-commerce application, hosted on IBM Cloud Foundry, represents a full-scale solution designed to offer a seamless and secure online shopping experience. This documentation provides an in-depth understanding of the various facets of our E-commerce project, from architecture and design to deployment and security. The application's core architecture leverages IBM Cloud Foundry's robust hosting capabilities, ensuring scalability and high availability. At its heart, our E-commerce system is built upon a microservices architecture, offering flexibility and modularity. The frontend, characterized by a user-friendly interface, caters to customers' needs, while the backend handles critical functions like user authentication, order management, and inventory tracking. Key components include a meticulously designed database schema, a range of API endpoints, and robust security measures. Deployment details, monitoring strategies, and maintenance procedures ensure that our application remains reliable and responsive. The documentation also outlines comprehensive testing processes, scalability strategies, and compliance measures. This project serves as a testament to our commitment to delivering a superior online shopping platform within the IBM Cloud Foundry ecosystem.





★★★★☆

SHAMPOO, CONDITIONER & FACEWASH P...

Lorem ipsum dolor sit amet consectetur
Lorem ipsum dolor dolor sit amet
consectetur Lorem ipsum dolor

\$150.00 ~~\$200.00~~

ADD TO CART

ALREADY SOLD: 20

AVAILABLE: 40

HURRY UP! OFFER ENDS IN:

Testimonial



ALAN DOE

- CEO & Founder Invision



Lorem ipsum dolor sit amet
consectetur Lorem ipsum dolor
dolor sit amet.

Trending



Air Trekking Shoes - White
Sports

Anon

Enter your product name...



Trending Accessories

**MODERN
SUNGLASSES**

SHOP NOW



DRESS & FROCK

(53)

Show All

Our Services



Worldwide Delivery
For Order Over \$100



Next Day Delivery
UK Orders Only

Hair Dye | Sunscreen | Skin Lotion | Liner |
Lipstick

POPULAR CATEGORIES

Fashion
Electronic
Cosmetic
Health
Watches

PRODUCTS

New Products

15%



JACKET

Mens Winter Leathers Jackets

★★★★☆

\$48.00 ~~\$75.00~~

SALE



About Us

Secure Payment

SERVICES

System Architecture:

Our E-commerce application is underpinned by a robust system architecture that enables efficient data flow and ensures high availability. The architecture is based on a microservices model, where various components operate independently while seamlessly communicating with one another. This approach enhances flexibility, scalability, and fault tolerance, which are essential attributes for an E-commerce platform. The microservices architecture also allows us to update and scale individual components without affecting the entire system, ensuring a smoother development and maintenance process.

Technologies Used:

Our tech stack comprises a carefully chosen selection of technologies and frameworks, which play pivotal roles in the application's success. We utilize languages such as Python and JavaScript, while front-end development relies on React.js, providing a dynamic and responsive user interface. On the backend, we leverage Node.js for its efficiency and scalability. Additionally, databases are managed using IBM Db2, ensuring robust data management and storage.

Microservices Architecture:

Microservices are at the core of our system design, enabling the decomposition of complex functionality into smaller, manageable services. Each microservice is responsible for specific functions, such as user authentication, product management, order processing, and more. The decoupled nature of microservices enhances agility, allowing us to make updates and additions without disrupting the entire system.

API Endpoints:

Our E-commerce backend exposes a comprehensive set of API endpoints to facilitate various functionalities within the application. Each endpoint serves a specific purpose, and their documentation is crucial for both internal and external users. Below is an example of an API endpoint used for user registration:

Endpoint: POST /api/users/register

Description: Register a new user.

Request:

```
{  
  "username": "example_user",  
  "email": "user@example.com",  
  "password": "securepassword123"  
}
```

Response:

```
{  
  "message": "User registered successfully."  
}
```

Authentication Mechanisms:

To ensure the security of user data and the integrity of our E-commerce platform, robust authentication mechanisms are implemented. These include OAuth for third-party login integrations and JWT (JSON Web Tokens) for securing user sessions. Additionally, API keys are utilized to protect sensitive endpoints.

Database Schema:

Our database schema is thoughtfully designed to support the various features and functions of our E-commerce application. It includes tables for user data, product information, order history, and more. These tables are structured to store and retrieve data efficiently while ensuring data consistency and integrity.

Property Name	Value
Database name	<database> [e.g. bludb]
Host name	<hostname>
Port	<port>
User Name	<username>
Password	<password>

IBM Cloud Db2

A next generation SQL database. Formerly dashDB For Transactions.

Create About

Type Service

Provider IBM

Last updated 09/20/2023

Category Databases

Compliance EU Supported HIPAA Enabled IAM-enabled

Location Sydney

Select a location

Sydney (au-syd)

Select a pricing plan

Displayed prices do not include tax. Monthly prices shown are for country or location: [India](#)

Plan	Features and capabilities	Pricing
Standard	Instance with flexible scaling of compute and storage Base instance starts at 8 GB RAM x 20 GB Storage	₹11.79736 INR/Instance-Hour ₹0.023429 INR/Gigabyte-Hours ₹8.39108 INR/Virtual Processor Core-Hour ₹0.002575 INR/BACKUP-GIGABYTE-HOURS

Summary

Db2 [Estimate costs](#)

Location: Sydney
Plan: Standard
Service name: Db2-mx
Resource group: Default

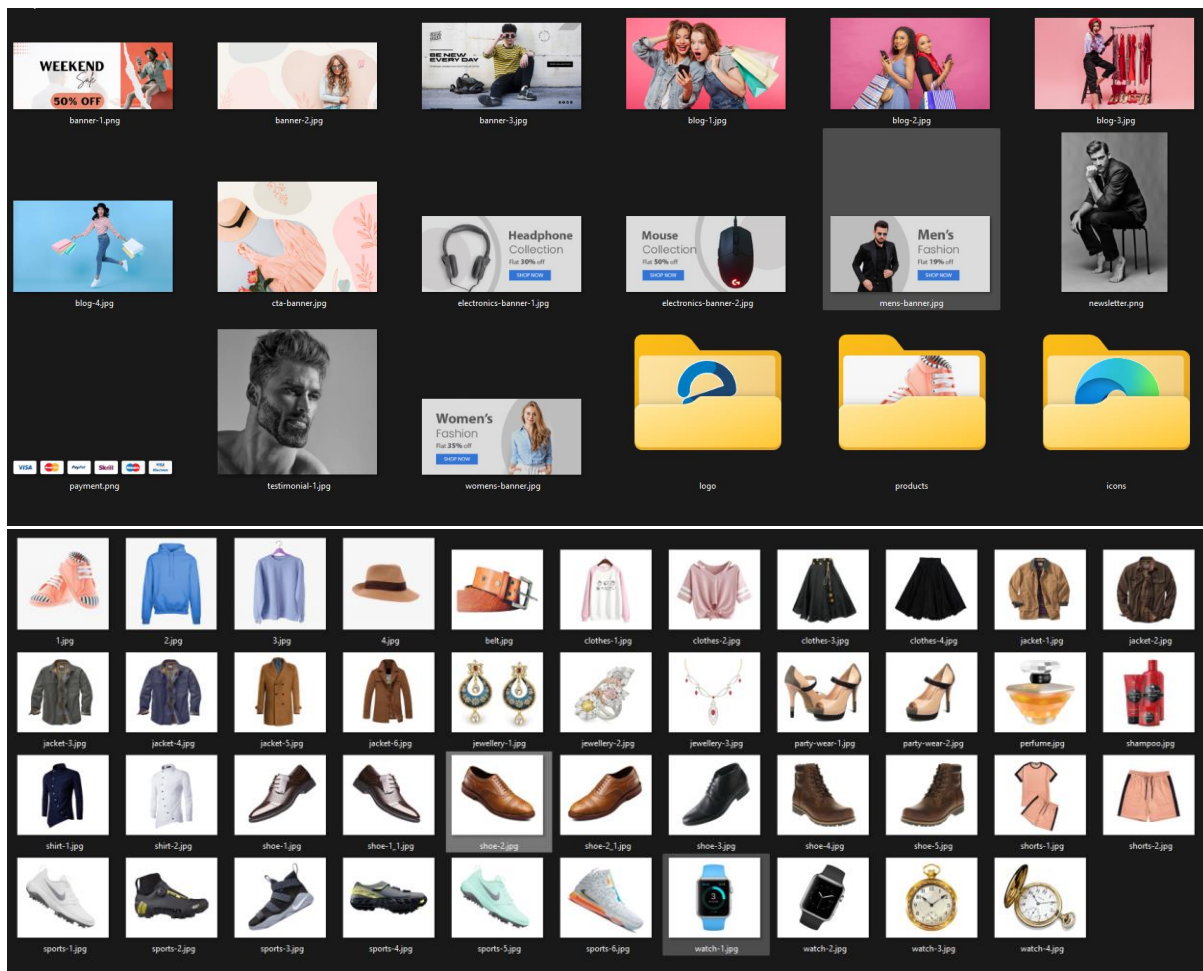
This paid plan cannot be added to an IBM Cloud trial account.
You can add a credit card to create a Pay-As-You-Go account. If a free plan for this service is available, you can choose to add it.

☐ I have read and agree to the following license agreements:
[Terms](#)
[Activate Windows](#)
[Go to Settings to activate Windows.](#)
[Upgrade](#)

UI Design:

Our user interface (UI) is designed with a focus on user-friendliness and a visually appealing layout. The design elements include intuitive navigation menus, responsive product grids, and a user-friendly shopping cart. [Include screenshots of your application's UI to provide visual context.]

Images used:



User Flows:

A vital component of our documentation is the description of user flows. These define the sequences of interactions users will

experience when engaging with our E-commerce application. For instance, we detail the user registration process, product search, checkout, and payment steps.

Frontend Technologies:

Frontend technologies are instrumental in delivering a rich user experience. We employ technologies like React.js, HTML5, CSS3, and JavaScript to create dynamic and responsive web pages. Libraries and frameworks such as Redux and Axios are also used to manage state and handle asynchronous requests.

Developing the frontend

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>eCommerce Shopping</title>
  <!--
    - favicon
  -->
  <link rel="shortcut icon" href="./assets/images/logo/logo.png" type="image/x-
  icon">
  <!--
    - custom css link
  -->
  <link rel="stylesheet" href="./assets/css/style-prefix.css">
  <!--
    - google font link
  -->
  <link rel="preconnect" href="https://fonts.googleapis.com">
```



```

<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
<link
href="https://fonts.googleapis.com/css2?family=Poppins:wght@300;400;500;6
00;700;800;900&display=swap"
rel="stylesheet">
</head>
<body>
<div class="overlay" data-overlay></div>
<!--
- MODAL
-->
<div class="modal" data-modal>
<div class="modal-close-overlay" data-modal-overlay></div>
<div class="modal-content">
<button class="modal-close-btn" data-modal-close>
<ion-icon name="close-outline"></ion-icon>
</button>
<div class="newsletter-img">

</div>
<div class="newsletter">
<form action="#">
<div class="newsletter-header">
<h3 class="newsletter-title">Subscribe Newsletter.</h3>
<p class="newsletter-desc">
Subscribe the <b>eShopping</b> to get latest products and discount
update.
</p>
</div>
<input type="email" name="email" class="email-field"
placeholder="Email Address" required>
<button type="submit" class="btn-newsletter">Subscribe</button>

```

```
        </form>
    </div>
</div>
</div>
<!--
- NOTIFICATION TOAST
-->
<div class="notification-toast" data-toast>
    <button class="toast-close-btn" data-toast-close>
        <ion-icon name="close-outline"></ion-icon>
    </button>
    <div class="toast-banner">
        
    </div>
    <div class="toast-detail">
        <p class="toast-message">
            Someone in new just bought
        </p>
        <p class="toast-title">
            Rose Gold Earrings
        </p>
        <p class="toast-meta">
            <time datetime="PT2M">2 Minutes</time> ago
        </p>
    </div>
</div>
<!--
- HEADER
-->
<header>
    <div class="header-top">
```

```
<div class="container">
  <ul class="header-social-container">
    <li>
      <a href="https://youtube.com/@dank_bro?si=4SMYO-z9o3I_1x40"
target="_blank" class="social-link">
        <ion-icon name="logo-facebook"></ion-icon>
      </a>
    </li>
    <li>
      <a href="https://youtube.com/@dank_bro?si=4SMYO-z9o3I_1x40"
target="_blank" class="social-link">
        <ion-icon name="logo-twitter"></ion-icon>
      </a>
    </li>
    <li>
      <a href="https://youtube.com/@dank_bro?si=4SMYO-z9o3I_1x40"
target="_blank" class="social-link">
        <ion-icon name="logo-instagram"></ion-icon>
      </a>
    </li>
    <li>
      <a href="https://youtube.com/@dank_bro?si=4SMYO-z9o3I_1x40"
target="_blank" class="social-link">
        <ion-icon name="logo-linkedin"></ion-icon>
      </a>
    </li>
  </ul>
  <div class="header-alert-news">
    <p>
      <b>Free Shipping</b>
      This Week Order Over - ₹550
    </p>
  </div>
```

```
</div>
</div>
<div class="header-main">
  <div class="container">
    <a href="#" class="header-logo">
      
    </a>
    <div class="header-search-container">
      <input type="search" name="search" class="search-field"
placeholder="Enter your product name...">
      <button class="search-btn">
        <ion-icon name="search-outline"></ion-icon>
      </button>
    </div>
    <div class="header-user-actions">
      <button class="action-btn">
        <ion-icon name="person-outline"></ion-icon>
      </button>
      <button class="action-btn">
        <ion-icon name="heart-outline"></ion-icon>
        <span class="count">0</span>
      </button>
      <button class="action-btn">
        <ion-icon name="bag-handle-outline"></ion-icon>
        <span class="count">0</span>
      </button>
    </div>
  </div>
</div>
<div>
  <nav class="desktop-navigation-menu">
    <div class="container">
```

```

<ul class="desktop-menu-category-list">
  <li class="menu-category">
    <a href="#" class="menu-title">Home</a>
  </li>
  <li class="menu-category">
    <a href="#" class="menu-title">Categories</a>
    <div class="dropdown-panel">
      <ul class="dropdown-panel-list">
        <li class="menu-title">
          <a href="#">Electronics</a>
        </li>
        <li class="panel-list-item">
          <a href="#">Desktop</a>
        </li>
        <li class="panel-list-item">
          <a href="#">Laptop</a>
        </li>
        <li class="panel-list-item">
          <a href="#">Camera</a>
        </li>
        <li class="panel-list-item">
          <a href="#">Tablet</a>
        </li>
        <li class="panel-list-item">
          <a href="#">Headphone</a>
        </li>
        <li class="panel-list-item">
          <a href="#">
            
          </a>

```

```
</li>
</ul>
<ul class="dropdown-panel-list">
  <li class="menu-title">
    <a href="#">Men's</a>
  </li>
  <li class="panel-list-item">
    <a href="#">Formal</a>
  </li>
  <li class="panel-list-item">
    <a href="#">Casual</a>
  </li>
  <li class="panel-list-item">
    <a href="#">Sports</a>
  </li>
  <li class="panel-list-item">
    <a href="#">Jacket</a>
  </li>
  <li class="panel-list-item">
    <a href="#">Sunglasses</a>
  </li>
  <li class="panel-list-item">
    <a href="#">
      
    </a>
  </li>
</ul>
<ul class="dropdown-panel-list">
  <li class="menu-title">
    <a href="#">Women's</a>
  </li>
```

```
<li class="panel-list-item">
  <a href="#">Formal</a>
</li>
<li class="panel-list-item">
  <a href="#">Casual</a>
</li>
<li class="panel-list-item">
  <a href="#">Perfume</a>
</li>
<li class="panel-list-item">
  <a href="#">Cosmetics</a>
</li>
<li class="panel-list-item">
  <a href="#">Bags</a>
</li>
<li class="panel-list-item">
  <a href="#">
    
  </a>
</li>
</ul>
<ul class="dropdown-panel-list">
  <li class="menu-title">
    <a href="#">Electronics</a>
  </li>
  <li class="panel-list-item">
    <a href="#">Smart Watch</a>
  </li>
  <li class="panel-list-item">
    <a href="#">Smart TV</a>
  </li>
```


Styling:

```
/*-----*\
```

```
#SIDEBAR
```

```
\*-----*/
```

```
.sidebar {  
  background: var(--white);  
  position: fixed;  
  top: 0;  
  left: -100%;  
  bottom: 0;  
  width: 100%;  
  max-width: 320px;  
  padding: 30px;  
  overflow-y: scroll;  
  overscroll-behavior: contain;  
  visibility: hidden;  
  transition: 0.5s ease;  
  z-index: 20;  
}
```

```
.sidebar.active {  
  left: 0;  
  visibility: visible;  
}
```

```
.sidebar-category {
```

```
margin-bottom: 15px;
padding-bottom: 15px;
border-bottom: 1px solid var(--cultured);
}
```

```
.sidebar-top {
  display: flex;
  justify-content: space-between;
  align-items: center;
  margin-bottom: 10px;
}
```

```
.sidebar-title {
  color: var(--onyx);
  font-size: var(--fs-5);
  text-transform: uppercase;
  letter-spacing: 0.8px;
  font-weight: var(--weight-600);
}
```

```
.sidebar-close-btn {
  color: var(--eerie-black);
  font-size: 22px;
  font-weight: var(--weight-600);
}
```

```
.sidebar-close-btn ion-icon { --icon-stroke-width: 50px; }
```

```
.sidebar-accordion-menu {  
  width: 100%;  
  display: flex;  
  justify-content: space-between;  
  align-items: center;  
  padding: 7px 0;  
}
```

```
.sidebar .menu-title-flex {  
  display: flex;  
  align-items: center;  
  gap: 10px;  
}
```

A small part of the css code is only given due to space constraints full source code is available in the GitHub. After the frontend designing we move on to the backend development of the project.

Deployment Strategy:

Our deployment strategy ensures a reliable and efficient application. We utilize the "cf push" command provided by IBM Cloud Foundry to deploy and update our application. The platform takes care of the deployment process, automatically managing resources and maintaining uptime.

Backend Code:

```
'use strict';
```

```
// modal variables
```

```
const modal = document.querySelector('[data-modal]);  
const modalCloseBtn = document.querySelector('[data-modal-close]);  
const modalCloseOverlay = document.querySelector('[data-modal-overlay]);
```

```
// modal function
```

```
const modalCloseFunc = function () { modal.classList.add('closed') }
```

```
// modal eventListener
```

```
modalCloseOverlay.addEventListener('click', modalCloseFunc);
```

```
modalCloseBtn.addEventListener('click', modalCloseFunc);
```

```
// notification toast variables
```

```
const notificationToast = document.querySelector('[data-toast]);
```

```
const toastCloseBtn = document.querySelector('[data-toast-close]);
```

```
// notification toast eventListener
```

```
toastCloseBtn.addEventListener('click', function () {
```

```
    notificationToast.classList.add('closed');
```

```
});
```

```
// mobile menu variables
```

```
const mobileMenuOpenBtn = document.querySelectorAll('[data-mobile-menu-open-btn]);
```

```
const mobileMenu = document.querySelectorAll('[data-mobile-menu]);
```

```
const mobileMenuCloseBtn = document.querySelectorAll('[data-mobile-menu-close-btn]);
```

```
const overlay = document.querySelector('[data-overlay]);
```

```
for (let i = 0; i < mobileMenuOpenBtn.length; i++) {

    // mobile menu function
    const mobileMenuCloseFunc = function () {
        mobileMenu[i].classList.remove('active');
        overlay.classList.remove('active');
    }

    mobileMenuOpenBtn[i].addEventListener('click', function () {
        mobileMenu[i].classList.add('active');
        overlay.classList.add('active');
    });
    mobileMenuCloseBtn[i].addEventListener('click', mobileMenuCloseFunc);
    overlay.addEventListener('click', mobileMenuCloseFunc);
}

// accordion variables
const accordionBtn = document.querySelectorAll('[data-accordion-btn]');
const accordion = document.querySelectorAll('[data-accordion]');

for (let i = 0; i < accordionBtn.length; i++) {

    accordionBtn[i].addEventListener('click', function () {

        const clickedBtn = this.nextElementSibling.classList.contains('active');

        for (let i = 0; i < accordion.length; i++) {
```

```
if (clickedBtn) break;

if (accordion[i].classList.contains('active')) {

    accordion[i].classList.remove('active');
    accordionBtn[i].classList.remove('active');
}
}
this.nextElementSibling.classList.toggle('active');
this.classList.toggle('active');
});
}
```

Scaling Strategies:

To accommodate varying levels of traffic and demand, we implement auto-scaling policies that adjust resource allocation dynamically. This ensures that our E-commerce platform remains responsive during peak usage, preventing any performance bottlenecks.

Security Measures:

Security is paramount in the E-commerce industry. Our application employs a range of security measures, including data encryption, secure socket layer (SSL) certificates, and robust access control mechanisms. Periodic security audits and penetration testing are conducted to identify and rectify vulnerabilities.

Compliance Standards:

Our E-commerce application adheres to industry-specific compliance standards such as the General Data Protection Regulation (GDPR) and Payment Card Industry Data Security Standard (PCI DSS). Compliance is ensured through strict data protection measures, including data anonymization and secure payment processing.

Testing Types:

Our testing process covers a range of critical areas to ensure the reliability and performance of our E-commerce application. This includes:

Unit Testing: Individual components and functions are tested in isolation to identify and resolve bugs.

Integration Testing: We examine how different components interact and ensure that they work seamlessly together.

Performance Testing: Our application is subjected to various stress tests to evaluate its performance under heavy loads.

Security Testing: Rigorous security assessments are conducted to identify vulnerabilities and security flaws.

User Acceptance Testing: Real users participate in testing to validate that the application meets their needs and expectations.

Test Results:

We maintain detailed records of test results and issues encountered during testing, ensuring a transparent and accountable development process. Specific issues and their resolutions are documented in our issue tracking system, providing a valuable resource for troubleshooting. **Monitoring Tools:**

We rely on advanced monitoring tools and systems to track various aspects of our application. Key monitoring areas include:

Server Health: Tools like Nagios and New Relic are used to monitor server health, alerting us to performance issues or outages.

Resource Utilization: We closely track resource utilization, such as CPU, memory, and network bandwidth, to ensure optimal performance.

Application Performance: Tools like Application Performance Management (APM) software help us identify and resolve bottlenecks and slowdowns.

Log Aggregation: Logs from various components are aggregated and analyzed using ELK (Elasticsearch, Logstash, Kibana) to detect and troubleshoot issues.

Maintenance Procedures:

Our maintenance procedures are designed to ensure that the application remains robust and up-to-date. Key elements include:

Patch Management: We regularly apply software patches and updates to address security vulnerabilities and bugs.

Feature Updates: New features and improvements are deployed in accordance with our change management process to minimize disruptions.

Version Control: We maintain a strict version control system, ensuring that code changes are tracked, tested, and documented before being deployed to the live environment. This includes utilizing Git and GitHub for code management.

Conclusion:

In conclusion, our journey to develop a comprehensive E-commerce application hosted on IBM Cloud Foundry has been an invaluable educational endeavor. This extensive documentation serves as a testament to our dedication, innovation, and commitment to learning and honing our development skills.

Throughout this project, we've gained hands-on experience in architecture design, microservices, cloud hosting, frontend and backend development, testing, monitoring, security, and user support. The application may not be intended for real-world customers, but it has provided us with a realistic and complex development environment that mirrors the challenges and complexities of the E-commerce industry.

This documentation not only summarizes our project but also serves as a learning resource for students and educators alike. It showcases the depth and breadth of our technical knowledge and development capabilities, offering insights into the various aspects of building and deploying a modern web application.

As students, we've leveraged the power of IBM Cloud Foundry to deploy and manage our application, applying best practices in development and documenting our progress. This documentation is a valuable resource for future students and developers who seek to understand and apply the principles of software development in a cloud-based environment.

Our journey has been an educational one, and this documentation encapsulates the knowledge and skills we've acquired, providing a roadmap for those embarking on similar educational endeavors. With this documentation, we are well-prepared to share our insights and lessons learned in the dynamic world of web application development.