**Deployment Plan Document**

**Project Title:** E-Commerce Application  
**Version:** 1.0  
**Prepared By:** [Insert Name]  
**Date:** [Insert Date]

**1️⃣ Document Purpose**

This document describes the deployment approach, environments, schedules, roles, and rollback strategies to successfully release the E-Commerce Application to production.

**2️⃣ Scope**

* Deploy customer-facing web application
* Deploy backend APIs and admin portal
* Configure database and third-party integrations (payment gateway, notifications)

**3️⃣ Deployment Environments**

| **Environment** | **Purpose** | **Hosting** |
| --- | --- | --- |
| Development | Feature development | Cloud (AWS/Azure) |
| QA / Staging | Testing & UAT | Cloud (AWS/Azure) |
| Production | Live customer environment | Cloud (AWS/Azure) |

**4️⃣ Pre-Deployment Checklist**

* Successful completion of functional, regression, and UAT testing
* Security audit sign-off
* Database backup completed
* Payment gateway keys configured
* Notifications (SMS/Email) tested
* Rollback plan in place

**5️⃣ Deployment Steps**

**5.1 Front-End (Web UI)**

* Build the production version (React.js/Angular)
* Upload build files to cloud storage (e.g., AWS S3)
* Configure CDN (e.g., AWS CloudFront)
* Enable HTTPS via SSL certificate

**5.2 Backend APIs**

* Pull latest version from Git repository
* Build and deploy using CI/CD pipeline (Jenkins/GitLab CI)
* Deploy backend services on EC2 / App Services / Containers
* Ensure APIs accessible via API Gateway with HTTPS

**5.3 Database**

* Apply final schema changes (if any)
* Seed essential data (admin account, categories)
* Enable automated backups and replication

**5.4 Payment Gateway Setup**

* Configure production API keys
* Conduct end-to-end payment test (with real/production gateway)
* Verify callback URL handling

**5.5 Notifications Setup**

* Integrate production SMS and email service credentials
* Validate test notification delivery

**5.6 Admin Panel Deployment**

* Build admin panel interface
* Deploy as a separate front-end service or under restricted route
* Enable admin authentication (RBAC)

**6️⃣ Rollback Plan**

| **Scenario** | **Action** |
| --- | --- |
| Major production defect | Revert backend APIs to previous stable release |
| Database issue post-deployment | Restore latest database backup |
| Payment gateway failure | Switch to alternative payment method (manual) |
| Complete failure | Disable live site & display maintenance message |

**7️⃣ Post-Deployment Verification**

* Verify homepage and product pages load correctly
* Complete a test order using real payment gateway
* Confirm order receipt email/SMS notifications
* Check admin dashboard access and operations
* Verify application monitoring (logs, performance metrics)

**8️⃣ Deployment Schedule**

| **Activity** | **Responsible** | **Date/Time** |
| --- | --- | --- |
| Final code merge | Development Lead | [Insert Date] |
| Staging deployment | DevOps Engineer | [Insert Date] |
| UAT completion | QA Team | [Insert Date] |
| Production deployment | DevOps Engineer | [Insert Date] |
| Post-deployment testing | QA Team | [Insert Date] |

**9️⃣ Roles & Responsibilities**

| **Role** | **Responsibility** |
| --- | --- |
| Project Manager | Overall coordination |
| Solution Architect | Technical oversight |
| DevOps Engineer | Perform deployment activities |
| QA Lead | Execute post-deployment verification |
| Product Owner | Approve Go-Live |

**🔟 Go-Live Approval**

| **Name** | **Role** | **Signature** | **Date** |
| --- | --- | --- | --- |
| Project Manager | Project Lead |  |  |
| DevOps Lead | Deployment Lead |  |  |
| Product Owner | Business Lead |  |  |