

By Aqsa Saeed

# DAY 6 - DEPLOYMENT PREPARATION AND STAGING ENVIRONMENT SETUP

---

## Deployment Challenges

### Primary Issue

A server-side exception occurred post-deployment, manifesting as:

Copy

```
"a server-side exception has occurred"
```

### Root Cause Analysis

The core issue was identified as hardcoded API endpoints in the application code. Specifically:

- Local development API calls were hardcoded to `http://localhost:3000/api/...`
- These endpoints failed in the production environment as `localhost` doesn't exist on Vercel's servers
- The issue primarily affected the cart page functionality

### Debugging Process

1. Initial Investigation
    - Explored potential database connectivity issues
-

- 
- Checked environment variable configurations
  - Investigated possible Next.js framework bugs
2. Research Methods
    - Utilized AI tools for troubleshooting
    - Consulted GitHub repositories
    - Reviewed Reddit threads and developer forums
    - Analyzed similar deployment issues in the community

## Resolution

### Implementation of Fix

1. Local Development Configuration
  - Created `.env.local` file with development URL:

```
NEXT_PUBLIC_API_URL=http://localhost:3000
```
2. Production Environment Setup
  - Configured Vercel environment variables:

```
NEXT_PUBLIC_API_URL=https://my-production-domain.com
```
3. Code Refactoring
  - Implemented dynamic base URL using environment variables:

```
const API_URL = process.env.NEXT_PUBLIC_API_URL

const response = await fetch(`${API_URL}/api/endpoint`, {

  method: "GET",

  credentials: "include",

});
```

---

# Key Learnings

## Best Practices Identified

1. Environment Configuration
  - Always use environment variables for API URLs
  - Avoid hardcoding environment-specific values
  - Maintain separate configurations for development and production
2. Testing Protocol
  - Implement production-like environment testing before deployment
  - Verify all API endpoints function in a production setting
  - Test critical user flows (especially cart functionality) in staging
3. Debugging Approach
  - Maintain a methodical debugging process
  - Start with simple solutions before exploring complex issues
  - Leverage community resources effectively

## Conclusion

The deployment issue was successfully resolved through the implementation of environment variables and proper API URL configuration. The project is now functioning correctly in production, with all features, including the cart page, working as intended. This experience highlighted the importance of proper environment configuration and thorough testing before deployment.