

# **Day 6: Deployment Preparation and Staging Environment Setup**

## **Objective:**

The goal of Day 6 is to prepare the marketplace for deployment by setting up a staging environment, configuring the hosting platform, and ensuring the application is ready for production-like testing.

## **Key Activities:**

### **1. Set Up Staging Environment:**

- Chose **Vercel** as the hosting platform.
- Linked GitHub repository to Vercel.
- Configured build and deployment settings on Vercel for a seamless deployment process.

### **2. Environment Variables Configuration:**

- Configured environment variables for sensitive data like API keys and project configurations.
- Used the `.env` file for local development and securely configured the environment variables in the Vercel dashboard for the staging environment.

### **3. Deployment to Staging:**

- Deployed the application to the staging environment.
- Verified deployment success with no errors during the build process.
- Ensured that the website loaded and functioned correctly in the staging environment.

### **4. Testing in Staging Environment:**

- Conducted functional, performance, and security tests to validate the application.

## **Testing Details:**

### *Functional Testing:*

- **Test Case ID:** TC001

**Description:** Validate Product Listing

**Steps:** Open product page > Verify products

**Expected Result:** Products displayed

**Actual Result:** Products displayed

**Status:** Passed

**Remarks:** No issues found

- **Test Case ID:** TC002

**Description:** Test API Error Handling

**Steps:** Disconnect API > Refresh page

**Expected Result:** Show fallback message

**Actual Result:** Fallback message shown

**Status:** Passed

**Remarks:** Handled gracefully

### *Performance Testing:*

- Used **Lighthouse** to analyze the load times, speed, and responsiveness of the staging environment.

### *Security Testing:*

- Validated input fields to ensure protection against SQL injection attacks.
- Ensured HTTPS was enabled for secure communication.
- Verified proper handling of sensitive data like API keys.

## **Deployment Documentation:**

### **1. GitHub Repository Structure:**

- `documents/`: Contains deployment and testing documentation.
- `src/`: Source code files for components and pages.
- `public/`: Static files like images and assets.

### **2. Environment Variable Configuration:**

- Created a `.env` file for local environment variables:

```
NEXT_PUBLIC_SANITY_PROJECT_ID=your_project_id  
NEXT_PUBLIC_SANITY_DATASET=production  
API_KEY=your_api_key
```

- Environment variables were securely configured in Vercel.

### **3. Deployment Steps:**

- **Step 1:** Connect GitHub repository to Vercel.
- **Step 2:** Set up build scripts and deployment settings on Vercel.
- **Step 3:** Deploy to staging.
- **Step 4:** Validate deployment and test functionality.

## **Performance and Test Case Reports:**

### **• Lighthouse Performance Report:**

Performance 97%

Accessibility 85%

Best Practices 96%

SEO 100%

- **Test Case Summary:**

A	B	C	D	E	F	G	H	I	J	K
1	Test Case ID	Test Case Descr	Test Steps	Expected Result	Expected Result Status	Severity Level	Assigned To	Remarks		
2	TC001	Product listing	Open homepage	Products should be listed	Pass	Low	-	All products displayed correctly		
3	TC002	Search function	Enter product name	Relevant products should be listed	Pass	Low	-	Search results accurate		
4	TC003	Cart operations	Add a product to Cart	Product should be added to Cart	Pass	Low	-	Cart functionality smooth		
5	TC004	Checkout process	Select product	Order should be created	Pass	Medium	-	No payment issues detected		
6	TC005	API error handling	Disconnect internet	Show 'No internet'	Pass	Medium	-	Fallback UI works fine		
7	TC006	Cross-browser compatibility	Open site in Chrome	Website should work	Pass	Low	-	Tested on major browsers		
8	TC007	Mobile responsiveness	Open site on mobile	Website should respond well	Pass	Low	-	Design adapts well		
9	TC008	Security testing	Enter invalid data	Validation errors should be shown	Pass	Medium	-	Validation prevents invalid input		
10	TC009	Secure API calls	Check API requests	All API calls should be secure	Pass	High	-	Secure API implementation confirmed		
11	TC010	User authentication	Login with valid credentials	Login should be successful	Pass	Medium	-	Authentication tested with various inputs		
12	TC011	Lighthouse Performance	Run Lighthouse	Performance score	Pass	Medium	-	Further optimization possible		

## README.md File:

- Summarizes all project activities, deployment steps, and test case results.
- Provides clear instructions for running the project locally, deploying to staging, and viewing the performance and test reports.

## Conclusion:

The staging environment has been successfully set up, and the application has passed all essential functional, performance, and security tests. The deployment to a production-like environment allows for further validation before going live. All project files, reports, and documentation have been structured and stored in the GitHub repository.