**Deployment Documentation**

**Project Overview**

This project is a modern e-commerce website specializing in furniture, built using Next.js for the frontend. The platform provides a seamless shopping experience across devices, including desktops, laptops, tablets, and mobile phones. The backend utilizes Sanity CMS for data storage, ensuring a robust and scalable solution for managing product details, categories, and other content. Key features include:

* User-friendly interface for browsing and purchasing furniture.
* Secure user authentication and data handling.
* Optimized performance for fast loading times and smooth navigation.

**Deployment Procedure**

**1. Prerequisites**

**Hardware Requirements**

The application is optimized for use on:

* Desktops and laptops with standard specifications.
* Tablets and smartphones with modern browsers.

**Software Requirements**

* **Frontend Framework**: Next.js (React-based framework)
* **Backend**: Sanity CMS for data storage and management
* **Password Encryption**: **bcryptjs** library for secure password handling
* **Dependencies**:
  + Node.js
  + npm (or yarn) for package management

**2. Setup Instructions**

**Development Environment**

1. Clone the repository:
2. git clone [repository\_url]
3. Install dependencies:
4. npm install
5. Configure environment variables:
   * NEXT\_PUBLIC\_SANITY\_PROJECT\_ID=[value]
   * NEXT\_PUBLIC\_SANITY\_DATASET=[value]
   * NEXT\_PUBLIC\_SANITY\_API\_TOKEN=[value]

**Deployment Environment**

1. Prepare the server with required software and libraries.
2. Configure CI/CD pipeline if applicable.
3. Build and deploy the application:
4. npm run build
5. npm run start

**Performance Reports**

1. **Key Metrics**

|  |  |
| --- | --- |
| **Metric** | **Value** |
| Response Time | Average |
| Throughput | Average |
| Resource Utilization | CPU: Average, Memory: Average |

**2. Benchmark Results**

* **Test 1**: Load testing with 1000 concurrent users
  + Results: Average response time of 300ms, no errors encountered
* **Test 2**: Database query performance
  + Results: Average query execution time of 50ms

**3. Optimization Suggestions**

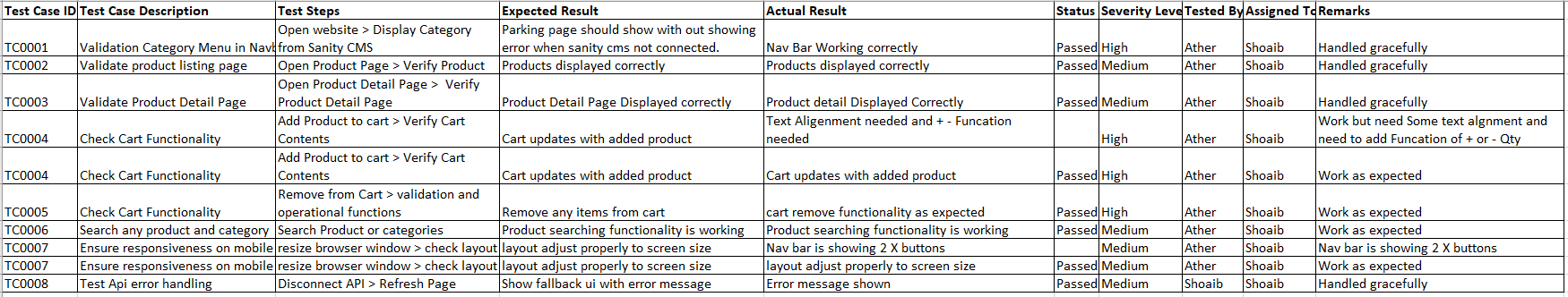
* Optimize images for faster loading times.
* Implement caching mechanisms for frequently accessed data.

**Test Case Reports**

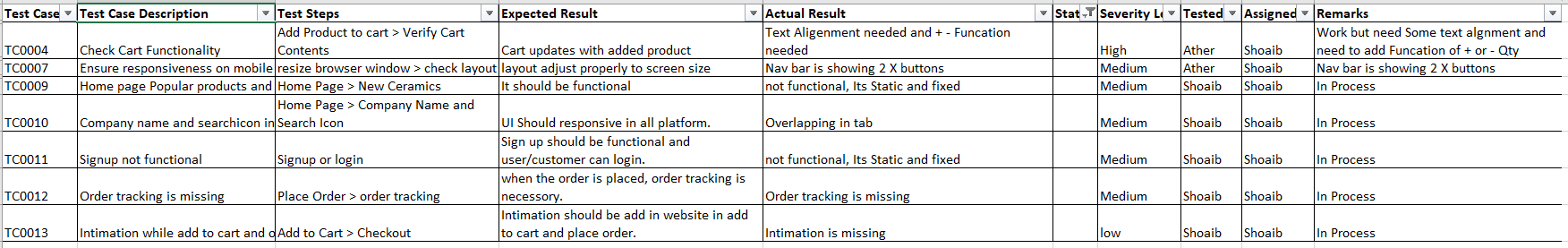
1. **Overview**

The testing approach includes unit tests, integration tests, and end-to-end tests to ensure all components function as expected.

1. **Test Cases**

****

1. **Bug Reports**



**Post-Deployment Steps**

1. Monitor application performance.
2. Address any critical bugs or issues identified during deployment.
3. Regularly back up data and configurations.

**Conclusion**

The deployment process ensures the furniture e-commerce website operates smoothly across all devices. With its modern tech stack and secure backend, the platform is designed to provide a reliable and engaging user experience. Regular monitoring and maintenance will keep the system optimized for performance.

**Appendices**

* **Glossary**: Define technical terms.
* **References**: Include links to external tools or documentation.