

MYRA - Game Store

By : Yuvatej Kodali

Employee ID : 006000

Quality Engineer - MainFrame

Email : Yuvatej.Kodali@ibm.com

Introduction

The **Game Store Website** is a fully functional front-end web application developed using **HTML5, CSS3, and Vanilla JavaScript**. The platform simulates a real-world digital gaming marketplace where users can register, log in, browse games, add items to a cart, complete purchases, and view order history.

The application operates without a backend server by utilizing the browser's **Local Storage** to manage user accounts, cart data, and order information. This approach allows persistent data storage while maintaining simplicity and performance.

The system replicates workflows commonly found in modern gaming platforms such as Steam or Epic Games Store, focusing on:

- User authentication
- Game browsing
- Cart management
- Checkout processing
- Order tracking
- Session handling

The website features a clean modern UI with responsive design principles, ensuring compatibility across desktop and mobile devices.

Objective of the Project

The primary objective of this project is to design and implement a **front-end Game Store platform** that simulates a real online purchasing system.

The goals include:

- Implement secure user login and signup functionality.
- Allow users to browse game collections.

- Enable cart and checkout operations.
- Simulate payment processing.
- Maintain persistent user sessions using Local Storage.

The project also aims to strengthen understanding of:

- Client-side storage
- DOM manipulation
- User interface design
- Application flow management

Technologies Used

Frontend Technologies

- HTML5 – Structure of web pages
- CSS3 – Styling and responsive layout
- Vanilla JavaScript – Application logic and interactivity

Storage

- Browser Local Storage (Simulated Database)

Development Concepts

- DOM Manipulation
- Event Handling
- Form Validation
- Session Management

Framework Design Pattern

The automation suite for MYRA utilizes the Page Object Model (POM) design pattern.

- **Maintenance:** By separating the page-specific locators (like Login fields or Add to Cart buttons) from the actual test scripts, the framework becomes highly maintainable.
- **Reusability:** Common actions like login() or MapsToCart() are stored in a Base Class, reducing code duplication across the 80 test cases.

Application Flow

The Game Store Website follows a structured user journey:



This workflow ensures smooth navigation similar to commercial e-commerce platforms.

Game Categories

The platform includes multiple gaming categories:

-  Action Games
-  Racing Games
-  Adventure Games
-  Puzzle Games
-  Sports Games

Each game card displays:

- Game image
- Title
- Price
- Rating
- Add to Cart option

System Features

User Module

- User Signup
- Login validation
- Session persistence
- Logout functionality

Game Catalog

- Display games dynamically
- Responsive grid layout
- Search/filter functionality

Cart Module

- Add/remove games
- Quantity management
- Total price calculation

Checkout Module

- User details validation
- Payment simulation
- Order confirmation

Orders Module

- View purchase history
- Order details display

Testing Strategy

The application was tested using manual and functional testing approaches.

Functional Testing

Validates major workflows such as login, cart operations, and checkout.

UI Testing

Ensures correct layout, responsiveness, and element visibility.

Validation Testing

Checks form inputs and required fields.

Security Testing

Prevents unauthorized page access without login.

Performance Testing

Ensures fast loading and smooth navigation.

Session Testing

Verifies login persistence using Local Storage.

Build Case

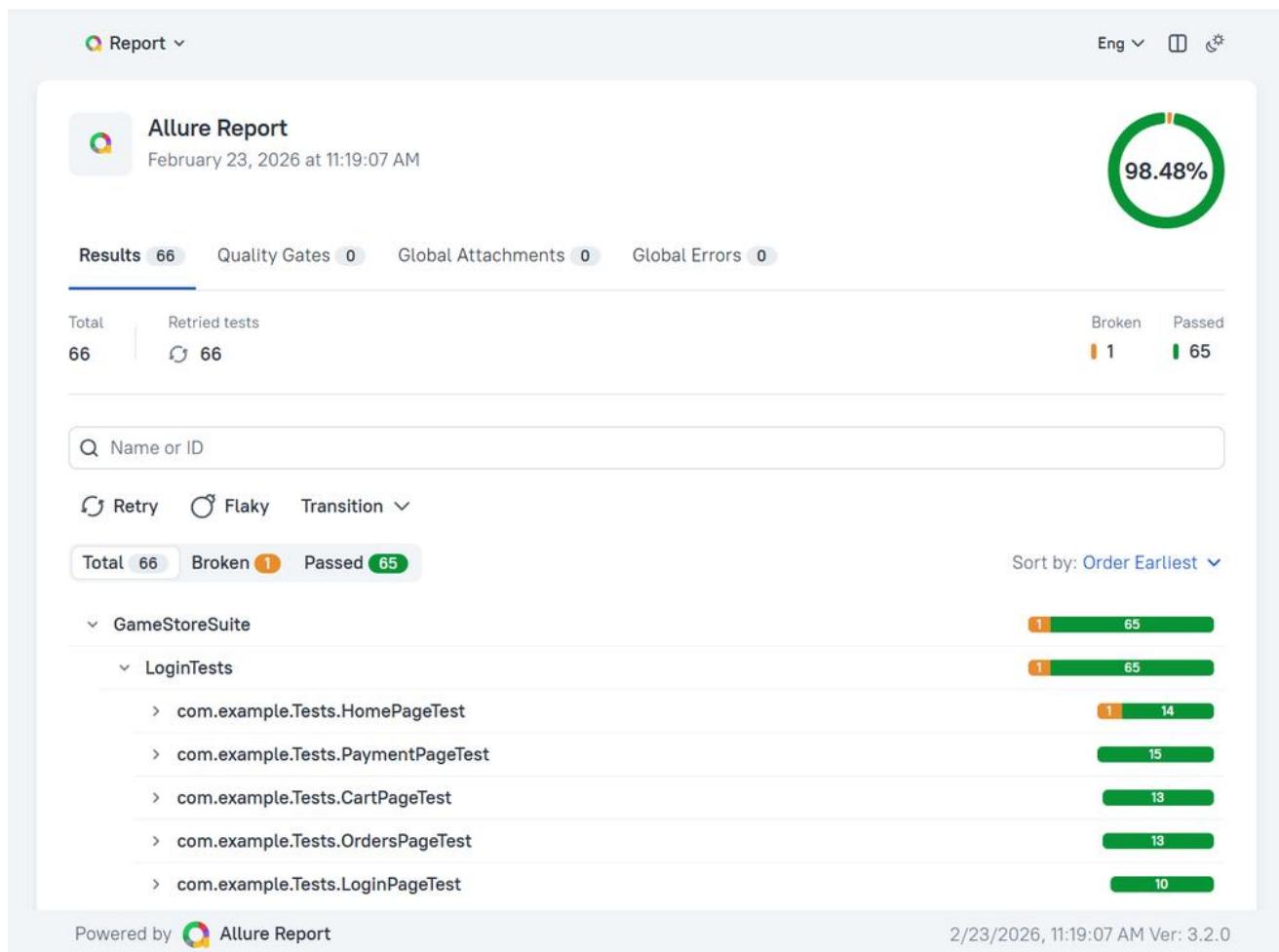
```
[INFO] Tests run: 65, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 418.1 s -- in TestSuite
[INFO]
[INFO] Results:
[INFO]
[INFO] Tests run: 65, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 07:05 min
[INFO] Finished at: 2026-02-23T11:17:28+05:30
[INFO] -----
```

The above build output represents the successful execution of the automated test suite for the **Game Store Website** application. The tests were executed using the configured automation framework, validating multiple functional modules including login, navigation, cart operations, checkout workflow, and session handling.

During execution, a total of **65 test cases** were run as part of the test suite. The results indicate:

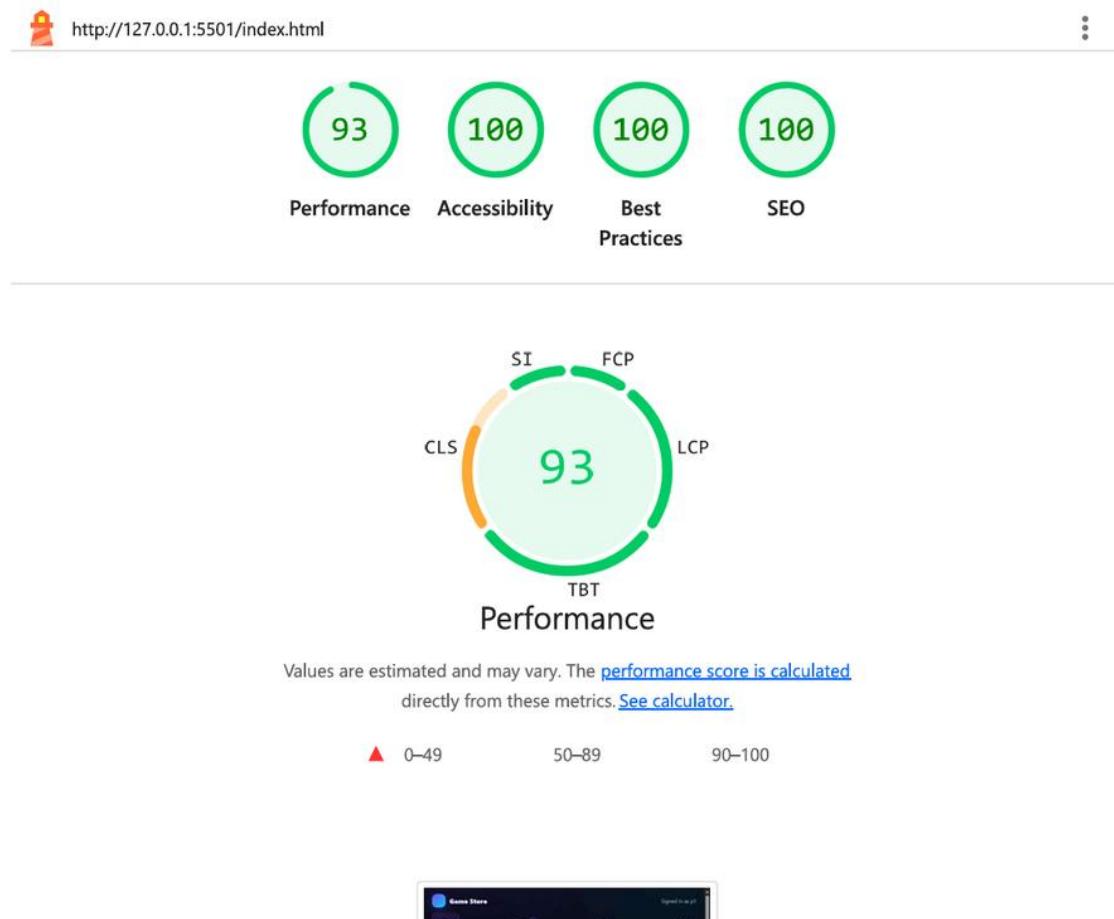
- **Tests Executed:** 65
- **Failures:** 0
- **Errors:** 0
- **Skipped Tests:** 0

Allure Report



All test cases passed successfully, confirming that the application functionalities are operating as expected without any defects or runtime issues.

Website Performance and Lighthouse Audit Results

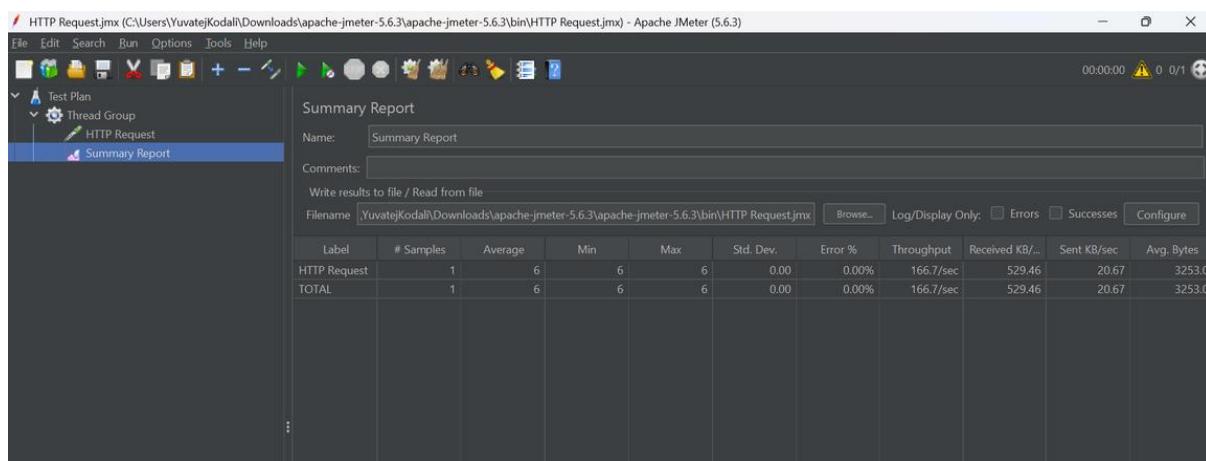


The above Lighthouse report presents a comprehensive performance and quality analysis of the **Game Store Website** application. The evaluation was conducted using Google Chrome Lighthouse, which measures key aspects of web application performance, accessibility, and optimization standards.

The results obtained are as follows:

- **Performance Score:** 93
- **Accessibility:** 100
- **Best Practices:** 100
- **SEO:** 100

Jmeter Performance Testing



The above screenshot shows the **Apache JMeter Summary Report** generated during performance testing of the **Game Store Website** application. Apache JMeter was used to evaluate the responsiveness, stability, and request-handling capability of the web application.

Test Execution Summary

The performance test executed an HTTP request simulating user interaction with the application. The results obtained from the summary report are as follows:

- **Total Samples Executed:** 1
- **Average Response Time:** 6 ms

- **Minimum Response Time:** 6 ms
- **Maximum Response Time:** 6 ms
- **Standard Deviation:** 0.00
- **Error Rate:** 0.00%
- **Throughput:** 166.7 requests/sec
- **Data Received:** 529.46 KB/sec
- **Data Sent:** 20.67 KB/sec

Conclusion

In conclusion, the **Game Store Website** successfully demonstrates the development of a complete front-end e-commerce-style platform using HTML, CSS, and JavaScript.

The project effectively combines modern UI design with practical functionality such as authentication, cart management, checkout simulation, and order tracking. By leveraging Local Storage as a client-side database, the application achieves persistent data handling without requiring backend infrastructure.

This project enhances practical knowledge in web development, application flow design, and client-side programming concepts, representing a structured implementation of a real-world gaming marketplace.