# WATCH STORE

Matam Santosh Matam.Santosh@ibm.com

## Introduction

Watch Store is a fully functional front-end watch shopping platform developed using HTML, CSS, and JavaScript. The application simulates a real-world online shopping system where users can register, log in, browse products, add items to the cart, complete checkout, make payments, and view order history. All data is managed using the browser's Local Storage, eliminating the need for a backend server while still maintaining persistent user and order information. The platform is structured to replicate the workflow of leading e commerce websites such as Flipkart and Amazon, focusing on user authentication, product management, cart functionality, checkout validation, and order tracking.

To ensure the reliability, functionality, and stability of the application, automated testing was implemented using Java, Maven, and Selenium WebDriver. Apache Maven is used as the build and dependency management tool, allowing smooth execution of test suites and project compilation. Selenium WebDriver is used to automate browser actions such as login validation, cart operations, checkout process, navigation flow, and session management. This ensures that each feature of the Watch Store platform works as expected under different test scenarios.

For advanced reporting and better visualization of test execution results, the Allure Report framework is integrated into the project. Allure generates detailed, interactive test reports that include passed, failed, and skipped test cases, execution steps, logs, and overall test statistics.

This enhances debugging capabilities and provides a professional-level reporting structure suitable for academic submissions and industry standards.

Overall, Watch Store not only demonstrates strong front-end development capabilities but also showcases structured automation testing, build management, and professional reporting practices, making it a complete end-to-end e-commerce and QA automation project.

## Objective of the Project

The objective of this project is to design and develop a fully functional watch shopping website that simulates a real-world online shopping platform. The project aims to ensure that all modules work smoothly and efficiently through structured testing.

Automation testing is implemented using Selenium WebDriver to validate different user flows such as login, adding products to cart, placing orders, and logout functionality. The project uses Apache Maven for build management, dependency handling, and smooth execution of test cases. Additionally, Allure Report is integrated to generate detailed and professional test execution reports.

Overall, the objective is to combine web development with automation testing and reporting tools to demonstrate a complete software testing lifecycle in a practical and structured manner.

## Technologies Used

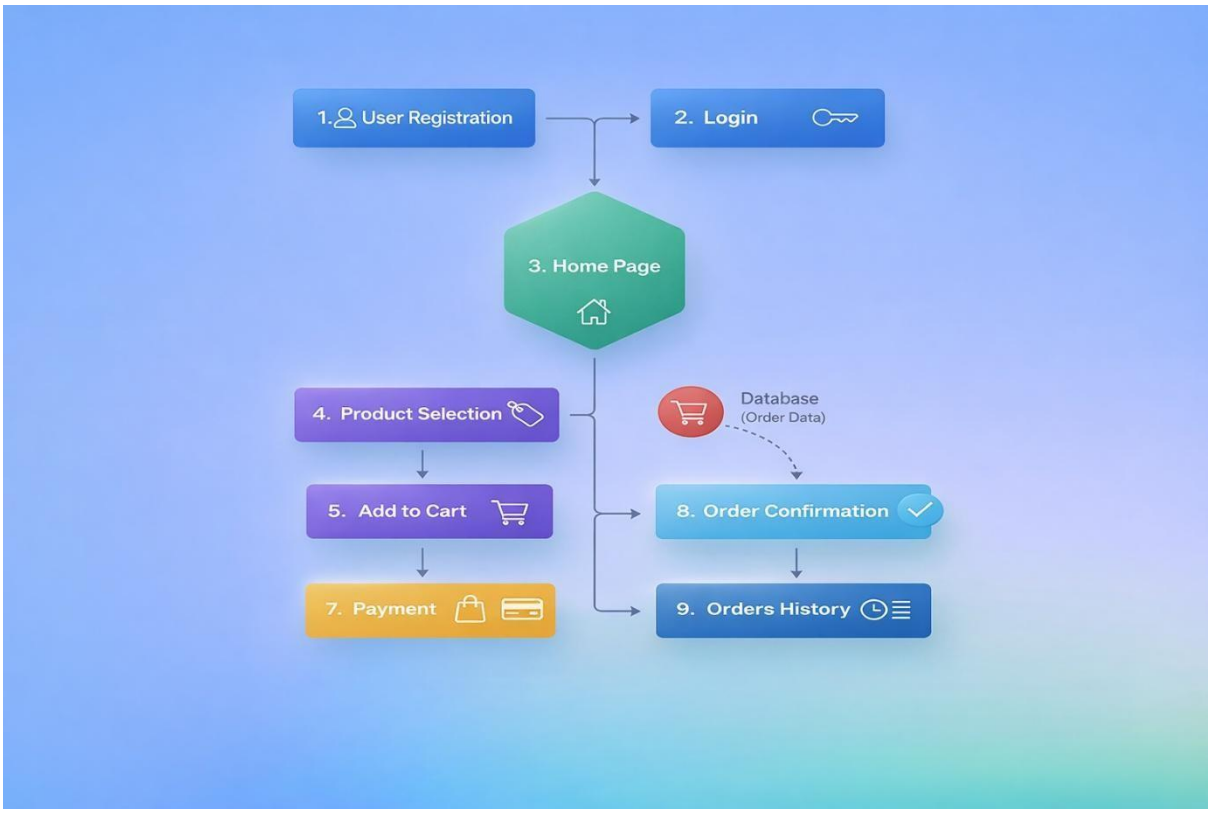
- Frontend: HTML5, CSS3, Vanilla JavaScript
- Storage: Local Storage
- Automation: Java, Selenium WebDriver, Maven
- Reporting: Allure Framework

## Framework Design Pattern

The automation suite for Watch Store Online Shopping Platform Utilizes the Page Object Model (POM) design pattern.

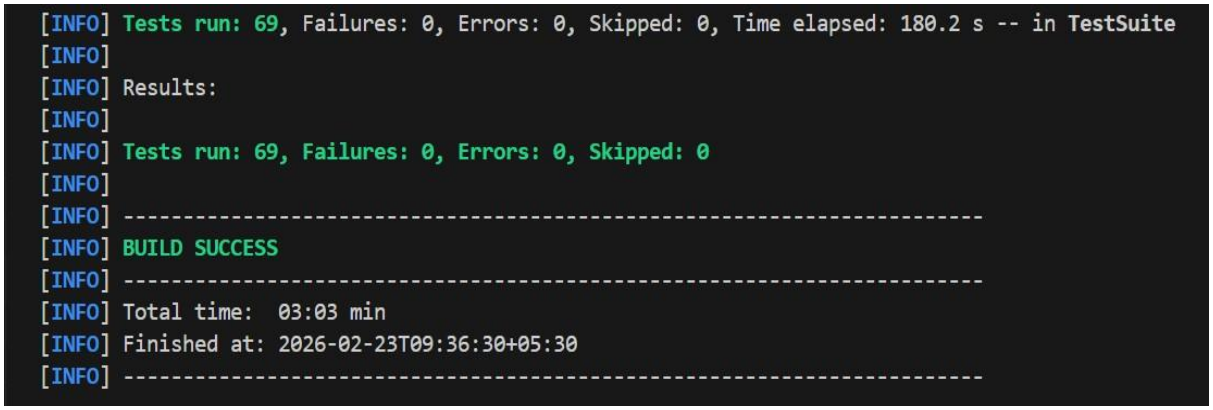
- **Maintenance:** By separating the page-specific locators (like Login fields, Add to Cart buttons, Payment details) from the actual test scripts, the framework becomes highly maintainable.
- **Reusability:** Common actions like login() or checkOut() are stored in a Base Class, reducing code duplication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

## Application Flow



## Testing Strategy

- Functional Testing – Validates core features.
- UI Testing – Ensures proper layout and element visibility.
- Validation Testing – Checks form validations.
- Security Testing – Prevents unauthorized access.
- Performance Testing – Verifies load time.
- Session Testing – Ensures login persistence.




## Test Cases Report

| TC ID | Module | Test Case Description | Preconditions | Test Steps | Expected Result | Priority |
|---|---|---|---|---|---|---|
| TC_01 | Login | Verify login page URL | Browser launched | Open login URL | Login page URL loaded | High |
| TC_02 | Login | Verify login page title | Login page open | Get page title | Title not empty | Medium |
| TC_03 | Login | Verify email field visibility | Login page open | Locate email input | Email field displayed | High |
| TC_04 | Login | Verify password field visibility | Login page open | Locate password input | Password field displayed | High |

| | | | | | | |
|---|---|---|---|---|---|---|
| TC_05 | Login | Verify login button visibility | Login page open | Locate login button | Login button displayed | High |
| TC_06 | Login | Verify login with valid credentials | Valid user exists | Enter valid credentials and submit | User logged in successfully | Critical |
| TC_07 | Login | Verify login with invalid credentials | Login page open | Enter invalid credentials | Error message shown | High |
| TC_08 | Login | Verify empty email validation | Login page open | Submit with empty email | Validation message shown | High |
| TC_09 | Login | Verify empty password validation | Login page open | Submit with empty password | Validation message shown | High |
| TC_10 | Home | Verify home page URL | Logged in | Open home page | Home page URL loaded | High |
| TC_11 | Home | Verify home page title | Home page open | Get page title | Title not empty | Medium |
| TC_12 | Home | Verify product cards displayed | Home loaded | Inspect product section | Products displayed | High |
| TC_13 | Home | Verify product name visibility | Home loaded | Check product name | Product name shown | Medium |
| TC_14 | Home | Verify product price visibility | Home loaded | Check product price | Price shown | High |
| TC_15 | Home | Verify Add to Cart button visibility | Home loaded | Locate add button | Button visible | High |
| TC_16 | Home | Verify Add to Cart functionality | Product available | Click add button | Product added to cart | Critical |
| TC_17 | Home | Verify multiple | Home loaded | Scroll products | Multiple products shown | Medium |

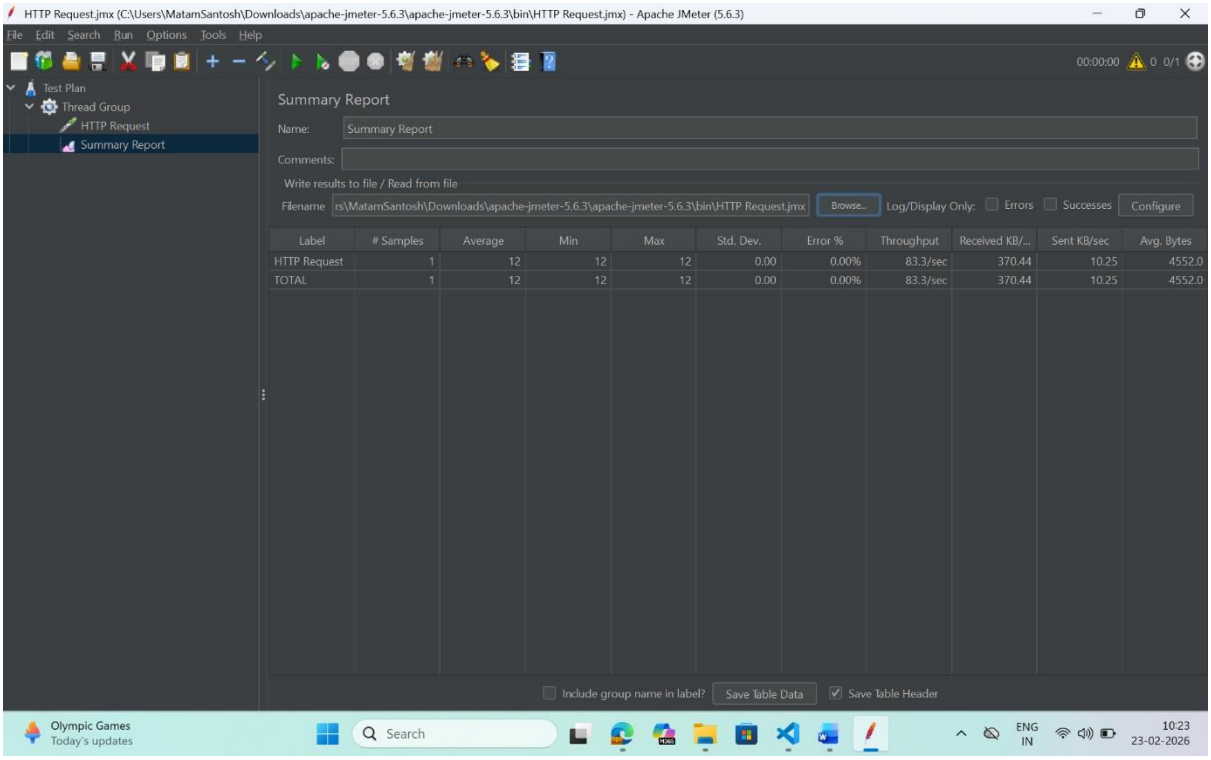
| | | | | | | |
|---|---|---|---|---|---|---|
| | | products displayed | | | | |
| TC_18 | Home | Verify search box visibility | Home loaded | Locate search bar | Search bar visible | High |
| TC_19 | Home | Verify search functionality | Search bar present | Search product | Matching results shown | High |
| TC_20 | Home | Verify scroll down functionality | Home loaded | Scroll down | Page scrolls | Medium |
| TC_21 | Home | Verify scroll up functionality | Home loaded | Scroll up | Page scrolls | Medium |
| TC_22 | Home | Verify page refresh | Home loaded | Refresh page | Page reloads successfully | Medium |
| TC_23 | Home | Verify navigation to cart page | Home loaded | Click cart icon | Cart page opened | High |
| TC_24 | Home | Verify browser back navigation | Navigated away | Click back | Returns to home | Medium |
| TC_25 | Cart | Verify cart page URL | Logged in | Open cart page | Cart URL loaded | High |
| TC_26 | Cart | Verify cart page title | Cart page open | Get title | Title not empty | Medium |
| TC_27 | Cart | Verify added item visible in cart | Item added | Open cart | Item displayed | High |
| TC_28 | Cart | Verify quantity field visibility | Cart page open | Locate quantity | Quantity shown | Medium |
| TC_29 | Cart | Verify quantity increment | Item in cart | Increase quantity | Quantity updated | High |
| TC_30 | Cart | Verify quantity decrement | Item in cart | Decrease quantity | Quantity updated | High |
| TC_31 | Cart | Verify remove item button | Item in cart | Locate remove button | Button visible | High |

| | | | | | | |
|---|---|---|---|---|---|---|
| TC_32 | Cart | Verify remove item functionality | Item in cart | Click remove | Item removed | High |
| TC_33 | Cart | Verify cart total calculation | Multiple items | Check total | Correct total displayed | Critical |
| TC_34 | Cart | Verify empty cart message | Cart empty | Open cart | Empty cart message shown | Medium |
| TC_35 | Cart | Verify continue shopping link | Cart page open | Click continue shopping | Redirects to home | Medium |
| TC_36 | Cart | Verify checkout button visibility | Item in cart | Locate checkout button | Button visible | High |
| TC_37 | Cart | Verify checkout navigation | Item in cart | Click checkout | Navigates to payment | High |
| TC_38 | Cart | Verify cart page refresh | Cart page open | Refresh page | Page reloads | Medium |
| TC_39 | Cart | Verify cart persistence after refresh | Item in cart | Refresh page | Items retained | High |
| TC_40 | Orders | Verify orders page URL | Logged in | Open orders page | Orders URL loaded | High |
| TC_41 | Orders | Verify orders page title | Orders page open | Get title | Title not empty | Medium |
| TC_42 | Orders | Verify orders list visibility | Orders exist | Inspect orders table | Orders displayed | High |
| TC_43 | Orders | Verify order ID displayed | Orders exist | Check order ID | Order ID visible | Medium |
| TC_44 | Orders | Verify order date displayed | Orders exist | Check order date | Date visible | Medium |
| TC_45 | Orders | Verify order status displayed | Orders exist | Check status | Status visible | High |

| | | | | | | |
|---|---|---|---|---|---|---|
| TC_46 | Orders | Verify view order details | Order exists | Click view details | Order details shown | High |
| TC_47 | Orders | Verify multiple orders listed | Multiple orders | Inspect list | All orders shown | Medium |
| TC_48 | Orders | Verify orders page refresh | Orders page open | Refresh page | Page reloads | Medium |
| TC_49 | Orders | Verify navigation from orders to home | Orders page open | Click home | Home page opened | Medium |
| TC_50 | Orders | Verify pagination if applicable | Many orders | Navigate pages | Pagination works | Low |
| TC_51 | Orders | Verify orders page scroll | Orders page open | Scroll page | Scroll works | Low |
| TC_52 | Orders | Verify order summary accuracy | Order exists | Check summary | Correct details shown | High |
| TC_53 | Orders | Verify order amount displayed | Order exists | Check amount | Amount visible | High |
| TC_54 | Orders | Verify no orders message | No orders | Open orders page | Message shown | Medium |
| TC_55 | Payments | Verify payment page URL | Logged in | Open payment page | Payment URL loaded | High |
| TC_56 | Payments | Verify payment page title | Payment page open | Get title | Title not empty | Medium |
| TC_57 | Payments | Verify payment form visibility | Payment page open | Locate form | Form displayed | High |
| TC_58 | Payments | Verify card number field | Payment page open | Locate card field | Field visible | High |
| TC_59 | Payments | Verify expiry date field | Payment page open | Locate expiry field | Field visible | Medium |

| | | | | | | |
|---|---|---|---|---|---|---|
| TC_60 | Payments | Verify CVV field | Payment page open | Locate CVV field | Field visible | High |
| TC_61 | Payments | Verify invalid card number | Payment page open | Enter invalid card | Error shown | High |
| TC_62 | Payments | Verify invalid CVV | Payment page open | Enter invalid CVV | Error shown | High |
| TC_63 | Payments | Verify empty payment submission | Payment page open | Submit empty form | Validation shown | High |
| TC_64 | Payments | Verify successful payment | Valid details | Submit payment | Payment successful | Critical |
| TC_65 | Payments | Verify payment confirmation message | Payment success | Observe confirmation | Message shown | High |
| TC_66 | Payments | Verify redirect after payment | Payment success | Observe redirect | Redirected correctly | High |
| TC_67 | Payments | Verify payment history update | Payment success | Open orders | Order updated | High |
| TC_68 | Payments | Verify payment page refresh | Payment page open | Refresh page | Page reloads | Medium |
| TC_69 | Payments | Verify payment failure handling | Invalid scenario | Submit payment | Failure handled properly | High |

# Performance Testing



Report Eng

# Allure Report

February 23, 2026 at 9:37:15 AM


Results 70 | Quality Gates 0 | Global Attachments 0 | Global Errors 0

| Total | Retried tests | Broken | Passed |
|---|---|---|---|
| 70 | 70 | 1 | 69 |

Name or ID

Retry Flaky Transition

Total 70 Broken 1 Passed 69 Sort by: Order Earliest

| | | |
|---|---|---|
| Selenium Suite | 1 | 69 |
| Website Tests | 1 | 69 |
| com.example.tests.OrdersTest | 1 | 15 |
| com.example.tests.CartTest | | 15 |
| com.example.tests.HomeTest | | 15 |
| com.example.tests.PaymentsTest | | 15 |
| com.example.tests.LoginTest | | 9 |

http://127.0.0.1:5500/home.html

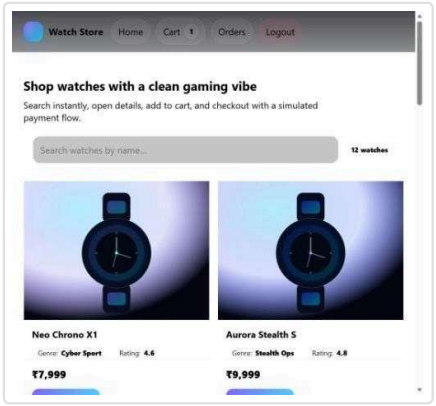
| Performance | Accessibility | Best Practices | SEO |
|---|---|---|---|
| 100 | 98 | 100 | 91 |

100

## Performance

Values are estimated and may vary. The performance score is calculated directly from these metrics. See calculator.

▲ 0–49 50–89 90–100



METRICS Expand view

First Contentful Paint
0.3 s

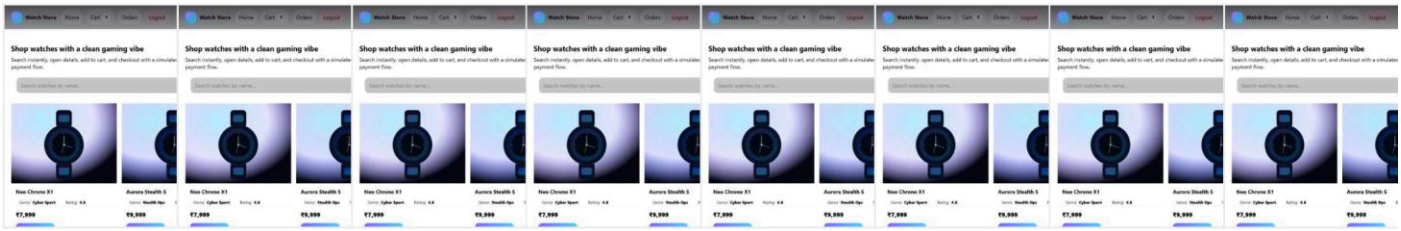
Largest Contentful Paint
0.4 s

Total Blocking Time
0 ms

Cumulative Layout Shift
0.001

Speed Index
0.3 s

1



Show audits relevant to: All FCP LCP CLS

INSIGHTS

- ▲ Render blocking requests — Est savings of 90 ms
- ▲ Network dependency tree
  - Document request latency — Est savings of 3 KiB
- ○ Layout shift culprits
- ○ LCP breakdown

These insights are also available in the Chrome DevTools Performance Panel - record a trace to view more detailed information.

DIAGNOSTICS

- ▲ Page prevented back/forward cache restoration — 1 failure reason
- Minify JavaScript — Est savings of 4 KiB

More information about the performance of your application. These numbers don't directly affect the Performance score.

PASSED AUDITS (22) Show


## Accessibility

These checks highlight opportunities to improve the accessibility of your web app. Automatic detection can only detect a subset of issues and does not guarantee the accessibility of your web app, so manual testing is also encouraged.

NAVIGATION

- ▲ Heading elements are not in a sequentially-descending order

These are opportunities to improve keyboard navigation in your application.

BEST PRACTICES

- Identical links have the same purpose.

These items highlight common accessibility best practices.

ADDITIONAL ITEMS TO MANUALLY CHECK (10) Show

These items address areas which an automated testing tool cannot cover. Learn more in our guide on conducting an accessibility review.

PASSED AUDITS (18) Show

NOT APPLICABLE (40) Show


## Best Practices

TRUST AND SAFETY

- Ensure CSP is effective against XSS attacks
- Use a strong HSTS policy

- Ensure proper origin isolation with COOP
- Mitigate clickjacking with XFO or CSP
- Mitigate DOM-based XSS with Trusted Types

PASSED AUDITS (13) Show

NOT APPLICABLE (2) Show


## SEO

These checks ensure that your page is following basic search engine optimization advice. There are many additional factors Lighthouse does not score here that may affect your search ranking, including performance on Core Web Vitals. Learn more about Google Search Essentials.

### CONTENT BEST PRACTICES

- ▲ Document does not have a meta description

Format your HTML in a way that enables crawlers to better understand your app's content.

### ADDITIONAL ITEMS TO MANUALLY CHECK (1) Show

Run these additional validators on your site to check additional SEO best practices.

Initial page load | Custom throttling | Using Chromium 145.0.0.0 with devtools

PASSED AUDITS (7) Show

NOT APPLICABLE (2) Show

Captured at Feb 23, 2026, 11:53 AM GMT+5:30 | Emulated Desktop with Lighthouse 13.0.1 | Single page session

Generated by **Lighthouse** 13.0.1 | File an issue

## Conclusion

Watch Store successfully demonstrates the development of a complete online shopping system using front-end technologies along with structured automation testing practices. The project effectively integrates core web development concepts with automated validation to ensure functional accuracy and system stability. By using Selenium WebDriver for automation, Apache Maven for build management, and Allure Report for advanced reporting, the project follows industry-standard testing methodologies.

This project not only enhances practical Knowledge of e-commerce workflows but also strengthens understanding automation frameworks, test execution strategies, and professional reporting. Overall, Watch Store represents a complete end-to-end implementation of development and quality assurance processes in a structured and efficient manner.