VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"Jnana Sangama", BELAGAVI – 590018

A REPORT ON

"KUSHAL'S WEBSITE TESTING"

Submitted in partial fulfillment of requirements for the course
SOFTWARE TESTING LABORATORY WITH MINI PROJECT
(21ISL66)
of Sixth Semester
of Bachelor of Engineering in Information Science & Engineering
During the academic year 2023-24.

Submitted By

AISHWARYA MN
4MH21IS005

ANAGHA ANANTH
4MH21IS008

Under the Guidance of

Prof. RACHANA S

Assistant Professor,

Dept. of IS&E

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

MAHARAJA INSTITUTE OF TECHNOLOGY MYSORE

Belawadi, S.R. Patna (T), Mandya (D) – 571477.

2023 - 2024

MAHARAJA INSTITUTE OF TECHNOLOGY MYSORE

Belawadi, S.R. Patna (T), Mandya (D) – 571477.

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

CERTIFICATE

This is to certify that the mini project work entitled **"KUSHAL'S"** a bonafide work carried out by **AISHWARYA MN[4MH21IS005]** and **ANAGHA ANANTH[4MH21IS008]** in partial fulfilment for the **Software Testing Laboratory (21ISL66)** prescribed by the Visvesvaraya Technological University, Belagavi during the year 2023-2024 for the sixth semester B.E in Information Science and Engineering. The mini project report has been approved as it satisfies the academic requirements.

Signature of Guide
(Prof. Rachana S)
Assistant Professor, Dept. of IS&E
MIT Mysore

Signature of HOD
(Dr. Sharath Kumar Y H)
Professor & Head, Dept. of IS&E
MIT Mysore

Name of the Examiners	Signature with date
1.	.....
2.	.....

ACKNOWLEDGEMENT

We sincerely owe our gratitude to all the persons who helped and guided us in completing this mini project work.

We are thankful to **Dr. B.G. Naresh Kumar, Principal, Maharaja Institute of Technology Mysore**, for having supported us in our academic endeavors.

We are extremely thankful to **Dr. Sharath Kumar Y H, Professor & Head, Department of Information Science and Engineering,** for his valuable support and his timely inquiries into the progress of the work.

We are greatly indebted to our guide **Prof.Rachana S, Assistant Professor, Department of Information Science and Engineering,** for the consistent co-operation and support.

We are obliged to all **teaching and non-teaching staff members** of **Department of Information Science and Engineering,** for the valuable information provided by them in their respective field's. We are grateful for their co-operation during the period of our mini project.

AISHWARYA MN (4MH21IS005)

ANAGHA ANANTH(4MH21IS008)

ABSTRACT

The automation testing of Kushal's IT website employs Katalon, a versatile framework for web application testing, to ensure the reliability and efficiency of its user interface and functionality. This project's objective is to automate repetitive test cases, ensuring a seamless user experience on Kushal's platform by simulating user interactions across various web browsers through Katalon Studio. The testing suite comprehensively covers essential features such as homepage load, newsletter subscription, social media links, contact us form, site navigation, promotional banners, mobile responsiveness, security, 404 error page, FAQ page, product recommendations, customer support chat, order tracking, return and refund policy, and the loyalty program. This automation aims to streamline the testing process, reduce manual effort, and enhance test coverage, ensuring the website's quality and performance. The use of Katalon's built-in keywords and custom keywords facilitates the creation of robust test scripts, while its integration with CI/CD tools enables continuous testing and faster delivery cycles. Kushal's IT, an innovative digital platform, redefines the user experience by offering a wide range of features designed to engage and support customers. Launched in [Year], Kushal's IT connects users with a seamless and intuitive web interface, enhancing user engagement and satisfaction. The platform's design is responsive and adaptive, catering to users on various devices and ensuring a consistent experience. By integrating advanced automation testing, Kushal's IT not only improves its overall user experience but also ensures a reliable and high-quality digital platform. This abstract explores the impact of automation testing on Kushal's IT website, its business model, and its contribution to providing a superior user experience in the digital marketplace. The automation efforts also focus on security testing to safeguard user data and ensure compliance with industry standards. Through these rigorous testing processes, Kushal's IT is positioned to maintain its competitive edge and continue to innovate in the digital space.

TABLE OF CONTENT

Chapter 1

INTRODUCTION

1.1 Aim of the Project:

The aim of the Kushal's Jewellery App project is to revolutionize the way consumers discover, experience, and purchase premium jewelry through an innovative mobile application. Centered on categories such as traditional, contemporary, and bespoke jewellery, Kushal's Jewellery App strives to provide users with a curated selection of high-quality pieces from established and emerging designers. The project aims to offer a personalized and engaging shopping experience by integrating advanced technology that allows users to virtually try on jewellery or receive detailed physical samples before committing to a purchase.

By bridging e-commerce with experiential marketing, Kushal's Jewellery App aims to enhance brand visibility, consumer trust, and satisfaction. This approach not only ensures a seamless and enjoyable shopping experience but also fosters a deeper connection between consumers and the jewellery brands they love. Through this transformation, Kushal's Jewellery App seeks to reshape the digital marketplace for luxury jewellery, providing users with an unparalleled experience in discovering and purchasing premium jewellery products.

1.2 Overview of the Project:

Kushal's Jewellery App is an innovative mobile application designed to revolutionize the way consumers discover, interact with, and purchase premium jewellery. Launched with a primary focus on categories such as traditional, contemporary, and bespoke jewellery, Kushal's Jewellery App offers users a curated selection of high-quality pieces from a diverse range of established and emerging designers. By integrating advanced technology and personalized shopping experiences, the app aims to provide an engaging and seamless journey for jewellery enthusiasts and buyers.

1.3 Outcome of the Project:

The outcome of implementing the proposed system for automating the testing of Kushal's Jewellery App using Katalon is a significant improvement in the reliability and efficiency of the app's user interface and functionality. The automated tests streamline the testing process, reducing the need for manual intervention and minimizing human errors. This leads to faster identification and resolution of defects, ensuring that new features and updates can be deployed with confidence. By maintaining high-quality standards through automated testing, Kushal's Jewellery App can provide a seamless and enjoyable experience for its users, ultimately enhancing customer satisfaction and loyalty.

1.4 Proposed System:

The proposed system for automating the testing of Kushal's Jewellery App using Katalon encompasses several key components to ensure comprehensive and efficient test coverage. Katalon Studio will be utilized for its integrated test automation capabilities, supporting multiple scripting languages, with Groovy being the chosen language for its ease of use and flexibility. Katalon's built-in features for test case creation, execution, and reporting will streamline the process.

The tests will be conducted across various mobile devices and operating systems, including iOS and Android, to ensure compatibility and responsiveness. Key user interactions such as account registration, product browsing, virtual try-on features, cart addition, and checkout processes will be identified and documented as test scenarios. Data-driven testing will be employed to validate multiple test cases with different inputs and scenarios.

Regular maintenance and updates of test scripts will be conducted to accommodate changes in the app's UI or functionality, with continuous improvements to the automation framework. This approach ensures that the Kushal's Jewellery App remains reliable and provides a high-quality user experience with every update.

1.5 Advantages:

- Increased Efficiency
- Enhanced Test Coverage
- Consistency and Reliability
- Cost Effectiveness
- Faster Feedback
- Improved Test Management
- Better Collaboration
- Scalability, Reusability and Maintainability

Chapter 2

SYSTEM REQUIREMENTS

2.1 Software Used:

- **Operating System:** Windows 10 or 11
- **Front End:** HTML, CSS, JavaScript, Katalon Studio
- **Back End:** Katalon Studio, Groovy (scripting language.

2.2 Software Description:

2.2.1 Katalon Studio:

Katalon Studio is an integrated test automation tool designed for web, mobile, and API testing. Developed by Katalon, Inc., it provides a comprehensive set of features for creating, managing, and executing automated tests with an intuitive interface.

- **Test Creation and Management:** Simplifies test case creation and management with record-and-playback functionality, script editing, and test design features.
- **Cross-Platform Testing:** Supports automation for web, mobile, and API applications, ensuring broad test coverage.
- **Integration Capabilities:** Seamlessly integrates with popular CI/CD tools (e.g., Jenkins, GitLab CI) and version control systems (e.g., Git, SVN) for streamlined automation workflows.
- **Reporting and Analysis:** Offers built-in reporting tools that generate detailed test reports and logs for easy analysis and debugging.
- **Data-Driven Testing:** Facilitates data-driven testing by allowing tests to be run with various data sets to validate different scenarios.
- **Script Editing:** Features a built-in scripting environment using Groovy, enabling advanced test script customization.
- **Object Repository:** Utilizes a centralized object repository for managing and reusing UI elements, enhancing maintainability and reducing script duplication.

2.2.2 Groovy:

Groovy is a dynamic scripting language used in Katalon Studio for advanced test automation. It provides:

- **Dynamic Scripting:** Enables flexible and dynamic scripting, which simplifies writing and maintaining test scripts.
- **Java Integration:** Allows integration with Java libraries and frameworks, benefiting from the Java ecosystem.
- **Simplified Syntax:** Offers a more concise syntax compared to Java, which can speed up development and improve script readability.

Chapter 03

SYSTEM ANALYSIS AND DESIGN

3.1 System Analysis:

System analysis is a detailed of the various operations performed by a system and their relationship within and outside the system. It is a systematic technique that defines goals and objectives. The goal of system development is to develop a system in line with the user requirement, and analysis of the system plays important role. One of the main aspects of analysis is the defining the boundaries of the system.

The various tools of structured analysis are:

- ➢ Stakeholder Analysis
- ➢ System Architecture
- ➢ Use-case diagram
- ➢ Test Planning

The structured analysis has the following attributes:

- ➢ Comprehensiveness**:** Ensures all functional and non-functional requirements are covered.
- ➢ Clarity**:** Provides clear documentation and diagrams for understanding system interactions.
- ➢ Stakeholder Involvement**:**Engages all relevant stakeholders to gather comprehensive requirements and feedback.
- ➢ Scalability**:** Considers the system's ability to handle growth and increased user demand.
- ➢ Maintenance**:** Simplifies system maintenance and updates through modular and documented approaches.
- ➢ Communication**:** Enhances communication among stakeholders by using standardized models and documentation.

3.2 Stake Holder Analysis:

- Developers**:** Write and maintain the codebase; ensure functionality and maintainability.
- Testers**:** Create and execute test cases; ensure features work as intended and are bug-free.
- Project Managers**:** Oversee project progress; ensure timely completion and quality standards.
- End-Users**:** Use the platform; seek a seamless, intuitive, and secure shopping experience.
- Business Analysts**:** Gather and analyse requirements; ensure features align with business goals.

3.3 System Architecture:

- Frontend: Built with HTML, CSS, and JavaScript to provide a responsive and interactive user interface for browsing products and managing shopping carts.
- Backend**:** Utilizes Java and Spring Boot to handle business logic, process user requests, and manage data transactions between the frontend and database.
- Database**:** Employs MySQL or PostgreSQL to securely store user data, product information, order details, and transaction histories.
- External Integrations**:** Integrates with payment gateways like Stripe and PayPal to securely process financial transactions and payments.
- Katalon Testing Framework: Utilizes Katalon Studio's built-in features for automating testing across web, mobile, and API applications. The framework supports the creation and execution of test cases using Katalon's user-friendly interface and scripting capabilities. It integrates data-driven testing and provides detailed reporting to ensure the application performs as expected across different platforms.

System architecture is a critical aspect of software testing and system engineering, defining the structure, behavior, and deployment of complex systems to meet functional and non-functional requirements effectively.

3.4 Use Case Diagram:

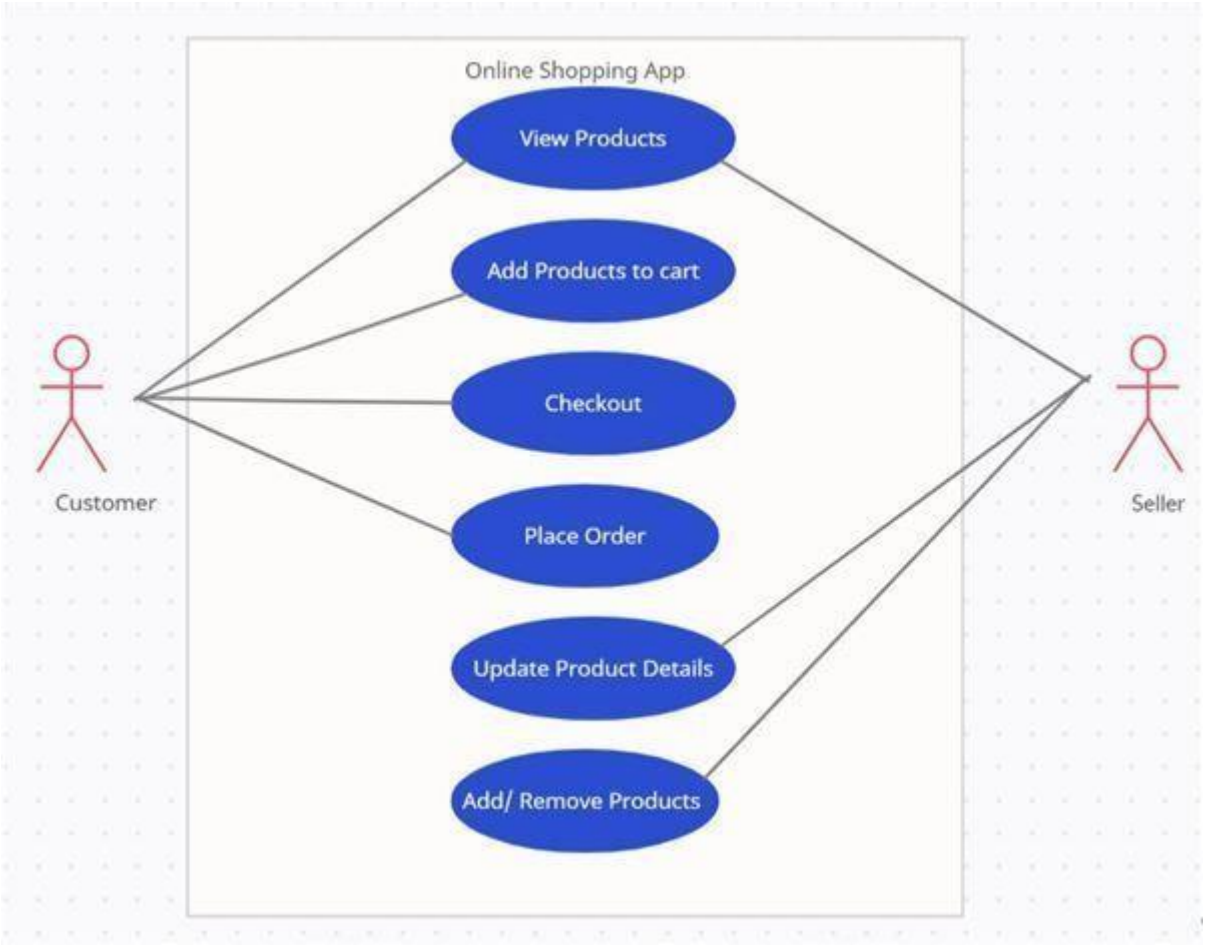

Fig.1 Use case diagram of Customer and Seller

- **View Product Details:** Enables users to see detailed information about a specific product, including images, descriptions, reviews, etc.
- **Add to Cart:** Allows users to add products to their shopping cart for future purchase.
- **Place Order:** Initiates the checkout process where users enter shipping details, select payment methods, and confirm their purchase.
- **Manage Products:** For administrators, includes tasks such as adding new products, updating product information, managing inventory, etc.

3.5 Test Planning:

Test planning for Kushal's website using Katalon Studio involves a structured approach to ensure the quality, reliability, and functionality of the application. Here's a detailed approach to planning the test strategy:

1. **Understanding Requirements and Use Cases:**

   - **Gather Requirements:** Collaborate with stakeholders to comprehend the functionalities and user interactions of Kushal's website.
   - **Use Case Analysis:** Identify key use cases and user workflows that need to be tested, such as product browsing, account management, and checkout processes.

2. **Test Environment Setup:**

   - **Infrastructure:** Set up necessary test environments (e.g., development, testing, staging) that mirror production environments as closely as possible.
   - **Tools:** Install and configure Katalon Studio, including any additional plugins or integrations needed for mobile, web, or API testing.

3. **Test Design:**

   - **Test Scenarios:** Design comprehensive test scenarios based on the requirements and use cases to cover all critical functionalities of Kushal's website.
   - **Test Data:** Prepare and manage test data required for executing the test scenarios, leveraging Katalon's data-driven testing features if applicable.

4. **Test Case Creation:**

   - **Create Test Cases:** Develop detailed test cases within Katalon Studio for each identified test scenario, using its built-in recording and scripting capabilities.
   - **Prioritization:** Prioritize test cases based on risk, criticality, and frequency of use to ensure efficient test execution.

5. Test Execution:

- **Automation Scripts:** Implement and execute automated test scripts using Katalon Studio's built-in features and Groovy scripting.
- **Execution Plan:** Develop a test execution plan detailing the schedule and procedures for running tests, including any parallel execution strategies if required.

6. Defect Management:

- **Reporting:** Define procedures for reporting and documenting defects discovered during testing, utilizing Katalon's built-in defect reporting features or integrating with external defect tracking tools.
- **Tracking:** Use defect tracking tools to monitor the status of reported issues and their resolutions, ensuring timely fixes and retesting.

7. Test Reporting:

- **Metrics:** Define metrics to evaluate test coverage, effectiveness of test cases, and overall application quality.
- **Reports:** Generate and distribute test summary reports using Katalon Studio's reporting features to stakeholders at regular intervals.

8. Test Maintenance:

- **Updates:** Plan for the ongoing maintenance of test cases and scripts to accommodate changes in Kushal's website, including updates to the UI or functionality.
- **Regression Testing:** Schedule periodic regression testing to ensure that new updates or bug fixes do not introduce new issues.

Tools and Technologies:

- **Katalon Studio:** For test automation, including web, mobile, and API testing.
- **Groovy:** For scripting and advanced test case customization within Katalon Studio.
- **CI/CD Tools (e.g., Jenkins, GitLab CI):** For continuous integration and automated test execution.
- **Version Control Systems (e.g., Git):** For managing test scripts and version control.

Chapter 04

IMPLEMENTATION

4.1 Test Case Development:

Test case development is essential for verifying the functionality of Kushal's website and ensuring its quality. Here is a structured approach to creating and managing test cases using Katalon Studio:

1. **Identify Test Scenarios:**

   - **Review Requirements:** Understand both functional and non-functional requirements of Kushal's website by reviewing project documentation and stakeholder inputs.
   - **Identify User Stories:** Use user stories, use cases, or business requirements to identify key scenarios that need to be tested, such as user registration, product search, and checkout.

2. **Define Test Case Structure:**

   - **Test Case ID:** Assign a unique identifier for each test case (e.g., TC_KW_001).
   - **Test Case Name:** Provide a clear and descriptive title that indicates the purpose of the test case (e.g., "Verify User Registration Process").
   - **Objective:** Outline what the test case aims to verify (e.g., "To verify that a user can successfully register on Kushal's website").
   - **Preconditions:** List any necessary conditions that must be met before executing the test case (e.g., "User is on the registration page").
   - **Inputs:** Specify data inputs required to execute the test case (e.g., "Username, Password, Email address").
   - **Steps:** Detail step-by-step instructions for executing the test case using Katalon Studio's test case editor (e.g., "1. Navigate to the registration page. 2. Enter valid username, password, and email address. 3. Click the 'Register' button").
   - **Expected Results:** Describe the expected outcome or behavior after executing the steps (e.g., "User should be registered successfully and redirected to the homepage with a welcome message displayed").

- **Actual Results:** Document the actual outcome observed during test execution, to be filled in after running the test case.

3. **Types of Test Cases:**

   - **Functional Test Cases:** Verify specific functionalities of Kushal's website, such as login, product search, and checkout.
   - **Integration Test Cases:** Validate interactions between different components or systems, such as payment gateways and user accounts.
   - **Regression Test Cases:** Ensure that new changes or updates do not negatively impact existing functionality.
   - **User Interface Test Cases:** Check the appearance and usability of the website, including layout and design elements.
   - **Performance Test Cases:** Assess the website's performance under various load conditions to ensure it handles traffic effectively.
   - **Security Test Cases:** Test for vulnerabilities and verify that security measures, such as data encryption and user authentication, are effective.

4. **Test Case Development Example:** Let's consider a test case for user registration on Kushal's website:

   - **Test Case ID:** TC_KW_001
   - **Test Case Name:** Verify User Registration.
   - **Objective:** To ensure that a user can successfully register on Kushal's website.
   - **Preconditions:** The registration page of the website is accessible.
   - **Inputs:** Username, Password, Email address
   - **Steps:**
     - Open Kushal's website.
     - Navigate to the registration page.
     - Enter valid username, password, and email address.
     - Click the "Register" button.
   - **Expected Results:** The user should be successfully registered and redirected to the homepage with a welcome message displayed.
   - **Actual Results:** To be filled in after execution.

5. **Documentation and Management:**

   - **Test Case Repository:** Maintain a centralized repository within Katalon Studio for storing and managing all test cases.
   - **Version Control:** Utilize version control features in Katalon Studio to track changes and updates to test cases over time.
   - **Traceability:** Ensure each test case is linked to its corresponding requirement or user story for traceability and coverage.

By following these steps, you can systematically develop effective test cases in Katalon Studio that ensure thorough testing coverage and contribute to delivering a high-quality website for Kushal's.

4.2 Test Data Preparation:

Preparing test data for Kushal's website involves creating and managing data that will be used during automated testing with Katalon Studio. Here's a structured approach to preparing test data for Katalon Studio testing:

1. **Identify Test Scenarios and Data Requirements:**

   - **Understand Functionalities:** Identify specific test scenarios you will automate with Katalon Studio, such as user registration, product search, checkout process, etc.
   - **Data Needs:** Determine the types of data required for each scenario to ensure comprehensive testing.

2. **Types of Test Data:**

   - **User Data:**

     - **Valid Users:** Create test accounts with valid usernames, passwords, and email addresses for testing user authentication and profile management.
     - **Invalid Users:** Include test accounts with invalid credentials (e.g., incorrect passwords) to validate error handling mechanisms.

- **Product Data:**

  - **Product Catalog:** Prepare test data for products with different attributes (e.g., name, price, availability) to test product listing and detail views.
  - **Special Cases:** Include products with special attributes (e.g., discounts, out-of-stock status) to test edge cases and special scenarios.

- **Order Data:**

  - **Shopping Cart:** Create scenarios with different items in the shopping cart to test checkout and order management functionalities.
  - **Order History:** Set up test orders with various statuses (e.g., pending, completed) to validate order tracking and history features.

- **User Profile Data:**

  - **Profile Information:** Populate profiles with diverse data (e.g., name, email, address) to test user profile management functionalities.
  - **Preferences:** Include user preferences such as saved addresses or favorite products to test personalized features.

3. Data Preparation Techniques for Katalon Studio:

- **Direct Input:** Use Katalon Studio's built-in features to directly input data into web forms or other interfaces during test execution.
- **CSV Files:** Prepare CSV files containing test data (e.g., user accounts, product details) that can be imported into Katalon Studio for data-driven testing.
- **Excel Files:** Utilize Excel files to manage and organize test data, leveraging Katalon Studio's support for Excel-based data sources.
- **Database Interaction:** Employ Katalon Studio's database testing capabilities or custom scripts to interact with and manage test data in the application's database.
- **API Calls:** Use Katalon Studio's API testing features to create or manipulate test data programmatically, useful for complex data setups.

4. **Managing Test Data in Katalon Studio Scripts:**

   - **Data Initialization:** Initialize the necessary test data before test execution by reading from CSV or Excel files, querying databases, or making API calls.
   - **Data Usage:** During test execution, Katalon Studio scripts can interact with the prepared test data to simulate user actions and validate expected behaviors.
   - **Data Cleanup:** Implement cleanup procedures in your test scripts to reset or remove test data modified during testing, ensuring test isolation and repeatability.

By following these steps, you can effectively prepare and manage test data in Katalon Studio for Kushal's website, ensuring thorough and reliable automated testing.

4.3 Test Environment Setup:

Setting up a test environment for Kushal's website using Katalon Studio involves configuring the necessary infrastructure, tools, and frameworks to automate testing efficiently. Here's a structured approach to setting up the test environment:

1. **Define Test Environment Requirements:**

   - **Hardware Requirements:** Assess and ensure that the hardware meets the needs for running automated tests. This includes having sufficient CPU power, RAM, and disk space to handle testing tasks.
   - **Software Requirements:**

     - **Operating System:** Select an OS supported by Katalon Studio (e.g., Windows, macOS, Linux).
     - **Browser Drivers:** Obtain and set up browser drivers (e.g., ChromeDriver, GeckoDriver) for browsers used by Kushal's website.
     - **Katalon Studio:** Download and install the latest version of Katalon Studio, which includes built-in support for various browsers and testing frameworks.

2. **Setup Katalon Studio:**

   - **Installation:** Install Katalon Studio by downloading it from the official website and following the installation instructions for your operating system.

- **License Configuration:** Activate the license for Katalon Studio, if required, by entering the license key or signing into your Katalon account.
- **Browser Drivers:** Ensure Katalon Studio is configured with the necessary browser drivers. Katalon Studio includes built-in support for popular browsers and manages browser drivers automatically, but you can manually update or add drivers if needed.

3. **Project Setup:**

- **Create a Katalon Project:** Open Katalon Studio and create a new project for testing Kushal's website. You can select from various project templates or start from scratch.
- **Import Dependencies:** Katalon Studio handles dependencies internally, so you do not need to manually configure them as you would with other frameworks. However, ensure that all necessary plugins and add-ons are installed for specific functionalities.

4. **Configure Test Environments:**

- **Test Environments Setup:** Define different test environments within Katalon Studio if needed (e.g., development, staging, production) to simulate various scenarios.
- **Browser Configuration:** Configure the browsers to be used for testing within Katalon Studio. You can manage and set browser preferences through the Test Suite settings.

5. **Write Test Scripts:**

- **Identify Test Scenarios:** Based on the requirements of Kushal's website, determine the test scenarios to be automated (e.g., user registration, product search, checkout process).
- **Create Test Cases:** Develop test cases using Katalon Studio's built-in features and tools. Use the Test Case editor to design and script test scenarios for various functionalities.

- **Page Object Model (POM):** Implement the Page Object Model (POM) design pattern within Katalon Studio to create reusable and maintainable code for interacting with web elements. This can be done by creating Page Object classes that represent different pages and their elements.

6. **Run and Manage Tests:**

- **Test Execution:** Execute test cases using Katalon Studio's Test Suite feature to run multiple test cases or entire test suites. Katalon Studio supports parallel test execution and scheduling.
- **Test Reporting:** Utilize Katalon Studio's built-in reporting features to generate and view test results, including detailed logs, screenshots, and reports.

By following these steps, you can effectively set up a test environment in Katalon Studio for Kushal's website, enabling comprehensive and efficient automated testing.

4.4 Test Execution:

Executing tests for Kushal's website with Katalon Studio involves automating various scenarios to ensure the website's functionality, usability, and performance. Here's a detailed guide on how to approach test execution using Katalon Studio:

1. **Preparation Before Test Execution:**

- **Test Plan Review:**

  - Ensure that the test plan is thorough and covers all critical functionalities of Kushal's website, such as user registration, product search, shopping cart management, checkout, and payment processing.

- **Test Data Setup:**

  - Prepare relevant test data, including user accounts (both valid and invalid), product details, orders, and payment information. Ensure that test data includes typical scenarios as well as edge cases.

- **Environment Readiness:**

  - Verify that the test environment (e.g., staging or test environment) closely mirrors the production environment, including the latest build, configurations, and integrations.

- **Test Execution Schedule:**

  - Plan the execution of test cases based on priority, dependencies, and risk factors. Organize test cases into suites or categories to manage execution effectively.

2. **Types of Tests to Execute:**

- **Functional Testing:**

  - **UI Testing:** Use Katalon Studio to automate tests for user interface elements, such as navigation, form submissions, product searches, product details, adding items to the cart, and completing the checkout process.
  - **Cross-Browser Testing:** Validate compatibility across different browsers (e.g., Chrome, Firefox, Edge) to ensure a consistent user experience.

- **Integration Testing:**

  - Verify the integration between different modules, such as user authentication, payment gateways, order management, and inventory management.

- **User Experience Testing:**

  - Evaluate usability aspects, including responsiveness (for mobile and desktop), accessibility, and consistency in user interface design.

3. **Execution Process:**

   - **Automated Testing with Katalon Studio:**

     - **Implement Test Scripts:** Use Katalon Studio to develop and implement test scripts. Katalon provides a user-friendly interface for creating test cases and test suites. Write scripts to automate various scenarios based on your test plan.
     - **Organize Test Suites:** Group test cases into test suites within Katalon Studio. You can create separate suites for different functional areas such as user management, product catalog, and checkout.
     - **Execute Automated Tests:** Run automated tests using Katalon Studio's built-in execution options. Katalon supports parallel execution, scheduling, and continuous integration, allowing you to efficiently manage and execute tests.

   - **Manual Testing:**

     - **Conduct Manual Tests:** Perform manual testing for scenarios that may require human judgment or are complex to automate. Manual testing helps validate edge cases, complex interactions, and usability aspects.

4. **Reporting and Analysis:**

   - **Test Reports:** Utilize Katalon Studio's reporting features to generate detailed test reports. These reports include test results, execution logs, screenshots, and summary information.
   - **Analyze Results:** Review test results to identify issues, track defects, and analyze test coverage. Use the information to improve test cases and address any issues found during testing.

By following these steps, you can effectively execute and manage tests for Kushal's website using Katalon Studio, ensuring thorough validation and high-quality performance.

4.5 Defect Reporting and Tracking:

Defect reporting and tracking for Kushal's website using Katalon Studio involves a structured process to identify, document, and manage issues encountered during automated testing. When defects are discovered during test execution, testers meticulously document each issue in a standardized format. This includes providing a clear title or summary of the defect, a detailed description of the problem including steps to reproduce it, and relevant attachments such as screenshots or logs. Katalon Studio facilitates capturing and attaching these details directly within the tool, ensuring comprehensive documentation.

This structured approach ensures that defects are systematically assigned to the appropriate developers, providing them with all necessary information to understand and address the issue effectively. Throughout the defect lifecycle—from initial reporting through resolution and verification—testers closely monitor the status of each defect. Effective communication among team members is crucial, and ongoing updates and discussions within the defect tracking tool help clarify requirements, provide additional context, and explore potential solutions.

Once developers implement fixes, testers re-execute the relevant test cases to verify that the defect has been resolved. This verification process ensures that the issue is addressed and does not reoccur. Comprehensive documentation of each defect and its resolution not only supports immediate defect management but also contributes to knowledge sharing and continuous improvement in the testing processes.

Chapter 05

RESULT ANALYSIS

5.1 Test Cases:

Test Case ID	Description	Expected Output	Actual Output	Status	Comments
TC01	User login with valid details.	login successful.	login successful.	Pass	Login verified.
TC02	User login with invalid email	Login fails with appropriate error message.	Login fails with appropriate error message.	Pass	Login failed due to incorrect email-id.
TC03	User login with invalid password	Login fails with appropriate error message.	Login fails with appropriate error message.	Pass	Login failed due to incorrect password.
TC04	Home page	Home page is displayed.	Home page is displayed.	Pass	Home page opens.
TC05	Valid search.	Search item is displayed.	Search item is displayed.	Pass	Searched item appears.
TC06	Invalid search.	Search item is not displayed.	Search item is not displayed.	Pass	Searched item doesn't show.
TC07	Filter the product.	Filtered product is displayed.	Filtered product is displayed.	Pass	Products are filtered.
TC08	Multiple filter for products.	Filtered products are displayed.	Filtered products are displayed.	Pass	Products are filtered.
TC09	Adding product to wishlist.	Product added to wishlist.	Product added to wishlist.	Pass	Product are added to wishlist.

TC10	Adding multiple products to wishlist.	Multiple products added to wishlist.	Multiple products added to wishlist.	Pass	Products are added to wishlist.
TC11	Add product to cart.	Product is successfully added to cart.	Product is successfully added to cart.	Pass	Product are added to cart.
TC12	Remove product from cart.	Product removed from cart.	Product removed from cart.	Pass	Product are removed to wishlist.
TC13	Check for Delivery date.	Successfully checked.	Successfully checked.	Pass	Delivery date is shown.
TC14	My orders history.	Orders displayed.	Orders displayed.	Pass	Order history is shown.
TC15	Reviews for a product.	Reviews displayed.	Reviews displayed.	Pass	Reviews are shown.
TC16	Wish list to cart.	Added successfully.	Added successfully.	Pass	Product added to cart from wishlist.
TC17	Selecting an item.	Item selected.	Item selected.	Pass	Item is selected.
TC18	Price low to high.	Successful.	Successful.	Pass	Products sorted from low to high.
TC19	Price high to low.	Successful.	Successful.	Pass	Products sorted from high to low.
TC20	Selecting images.	Images displayed.	Images not found.	Fail	Invalid result,image not selected.

5.2 Result and discussion:

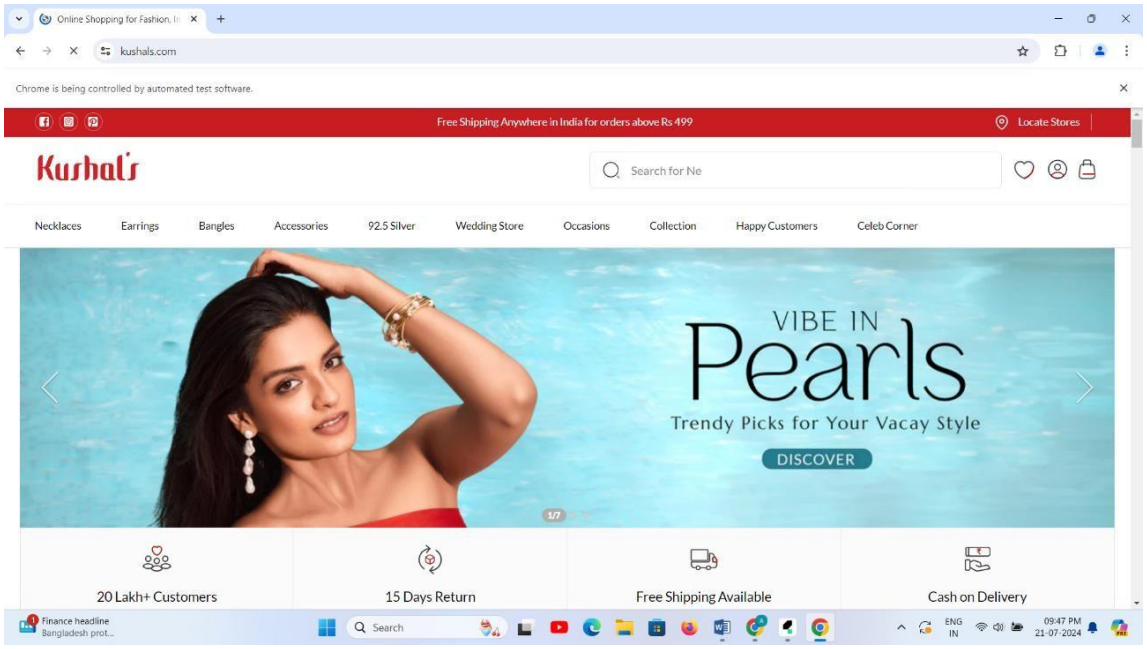

Fig 1: Home page

This figure shows the homepage of Kushal's website, displaying the main navigation menu and promotional banner. It's used to verify the layout, visual elements.

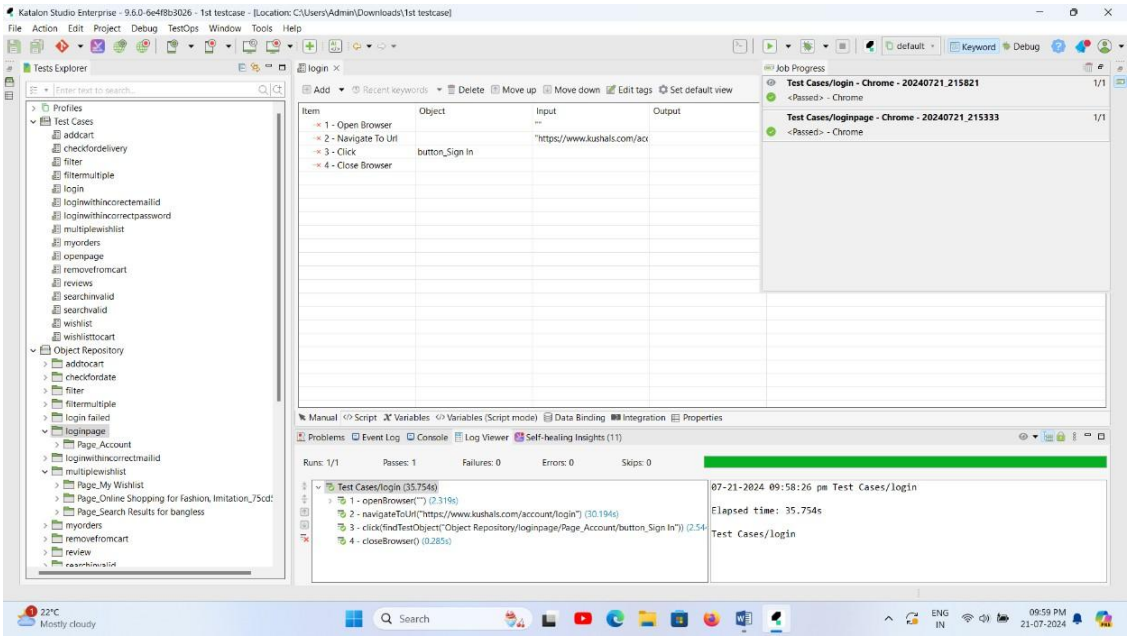

Fig 2: Login page test case

This figure captures Katalon Studio where test cases for Kushal's website are developed and executed. It shows the test steps for a login scenario, including successful execution results.

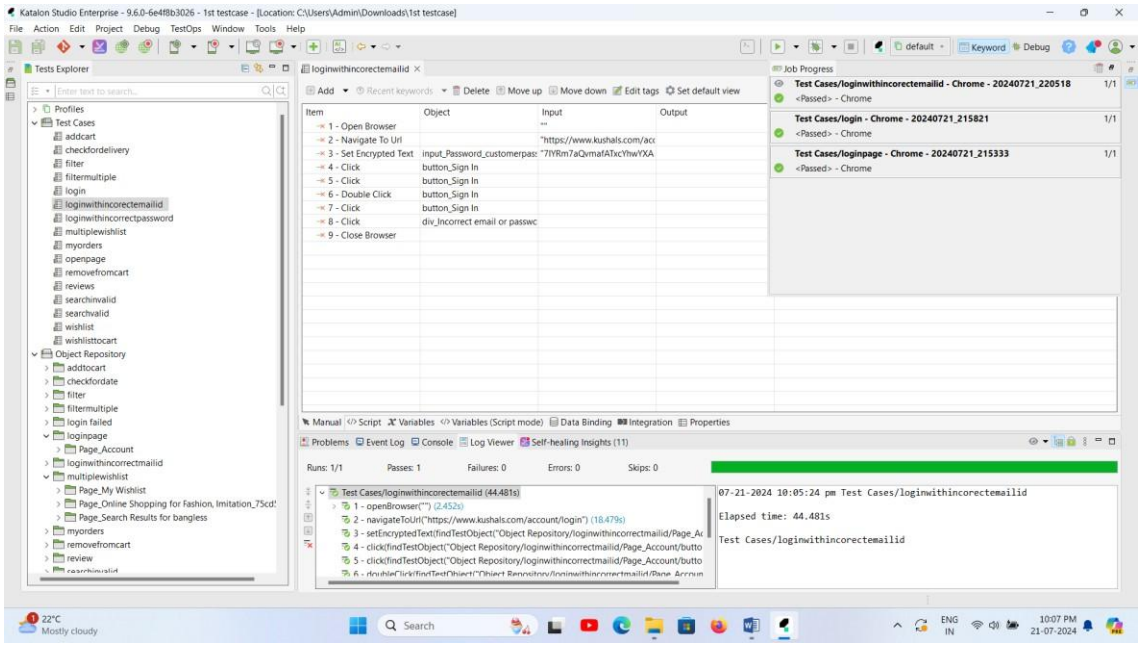

Fig 3: Login Page with incorrect Email id test case

This figure captures Katalon Studio where test cases for Kushal's website are developed and executed. It shows the test steps for an incorrect email-id login scenario, including successful execution results.

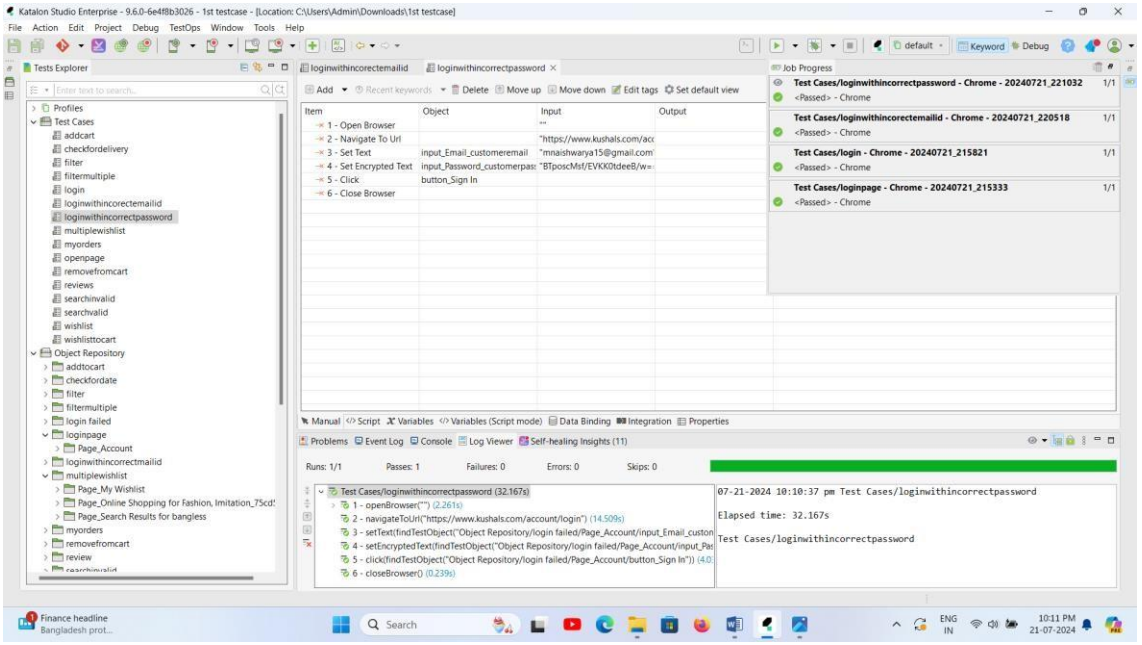

Fig 4: Login Page with incorrect password test case

This figure captures Katalon Studio where test cases for Kushal's website are developed and executed. It shows the test steps for an incorrect password login scenario, including successful execution results.

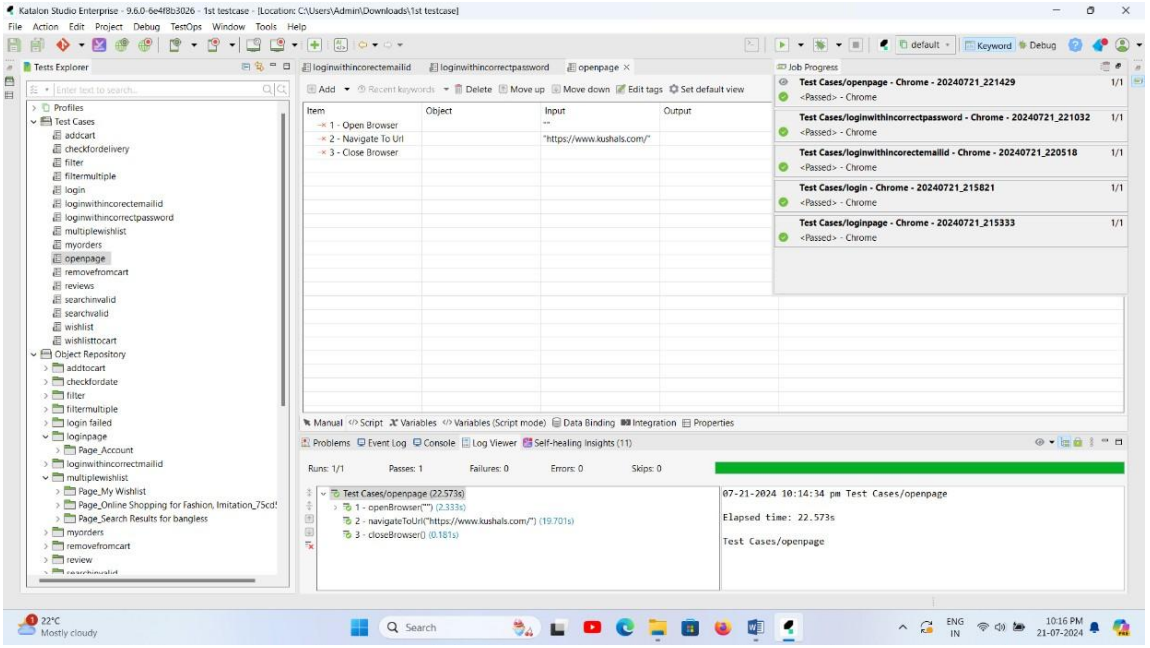

Fig 5: Open home page test case

This figure captures Katalon Studio where test cases for Kushal's website are developed and executed. It shows the test steps to open home page, including successful execution results.

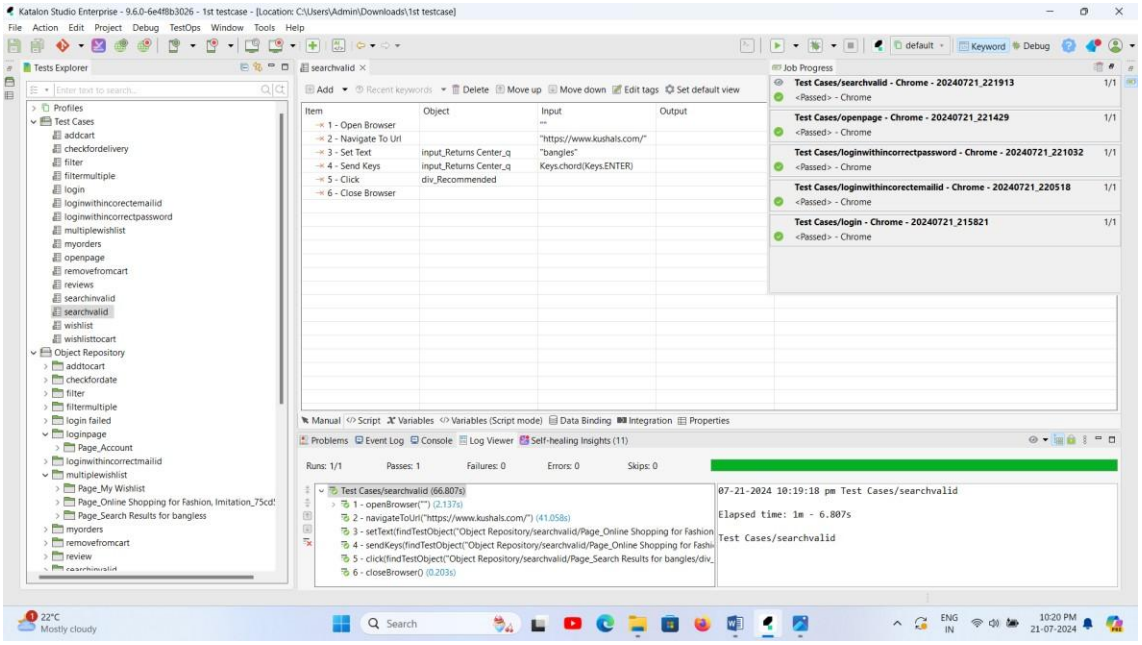

Fig 6: Search valid test case

This figure captures Katalon Studio where test cases for Kushal's website are developed and executed. It shows the test steps to search is valid, including successful execution results.

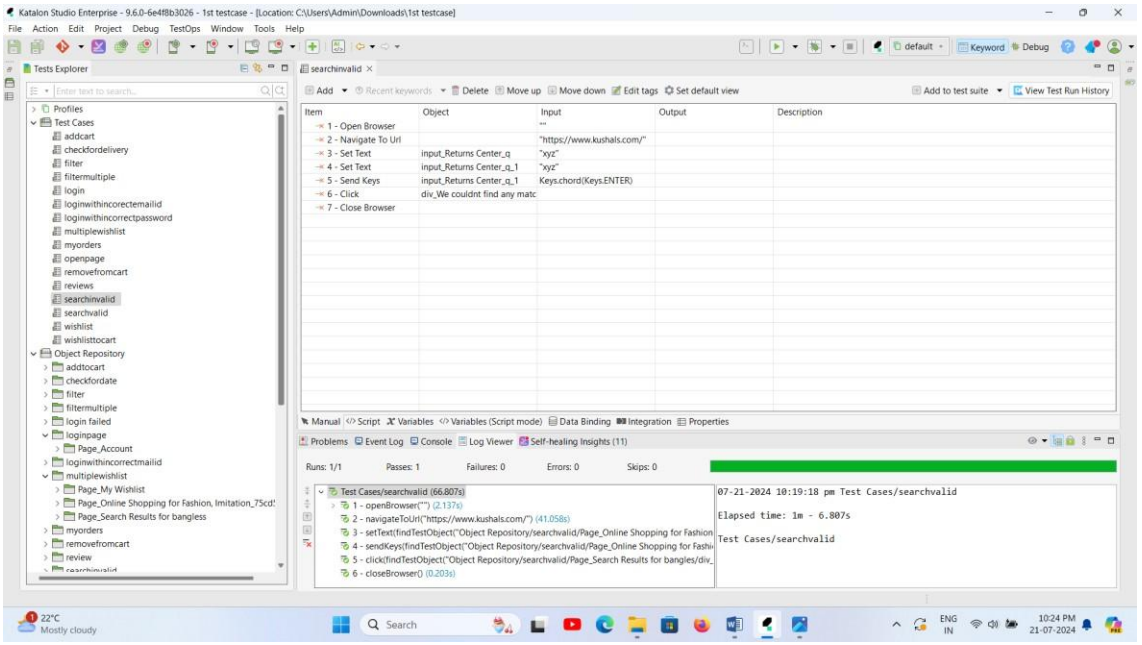

Fig 7: Login Page with incorrect password

This figure captures Katalon Studio where test cases for Kushal's website are developed and executed. It shows the test steps to search is invalid, including successful execution results.

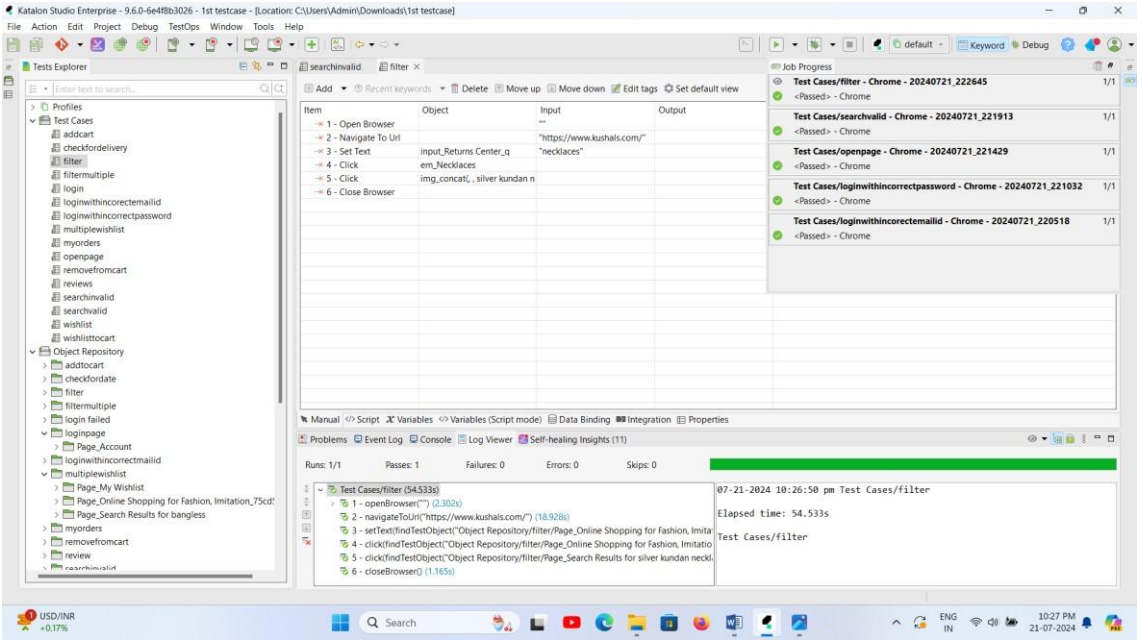

Fig 8: Single Filter Option test case

This figure captures Katalon Studio where test cases for Kushal's website are developed and executed. It shows the test steps to show single item filter scenario, including successful execution results.

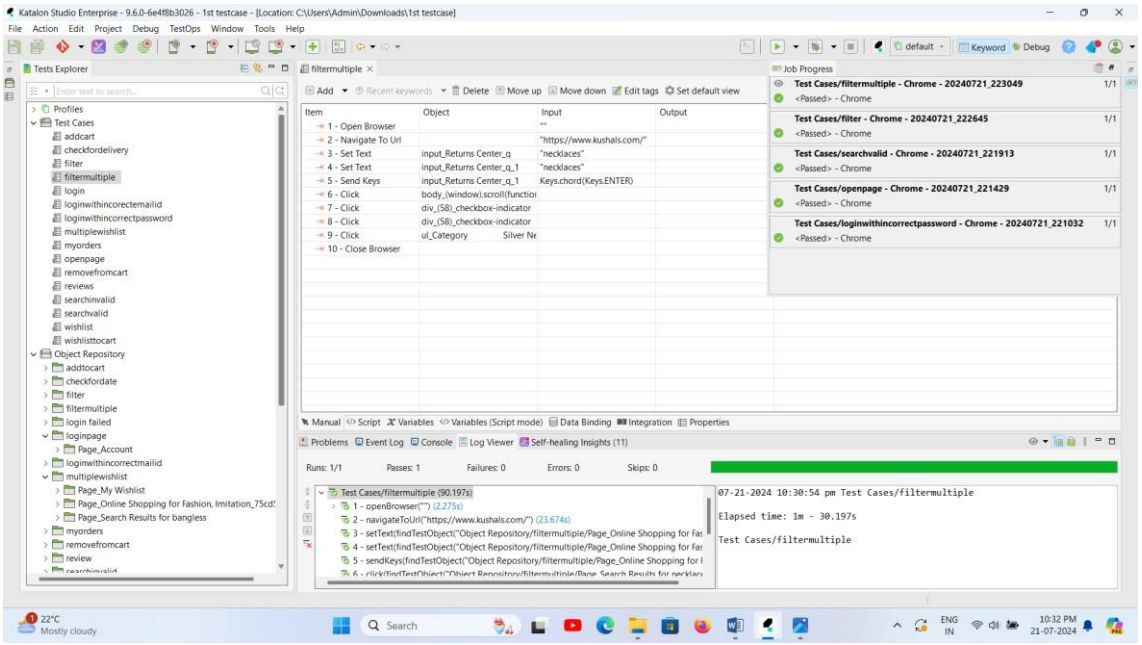

Fig 9: Multiple Filter Option test case

This figure captures Katalon Studio where test cases for Kushal's website are developed and executed. It shows the test steps to show multiple item filter scenario, including successful execution results.

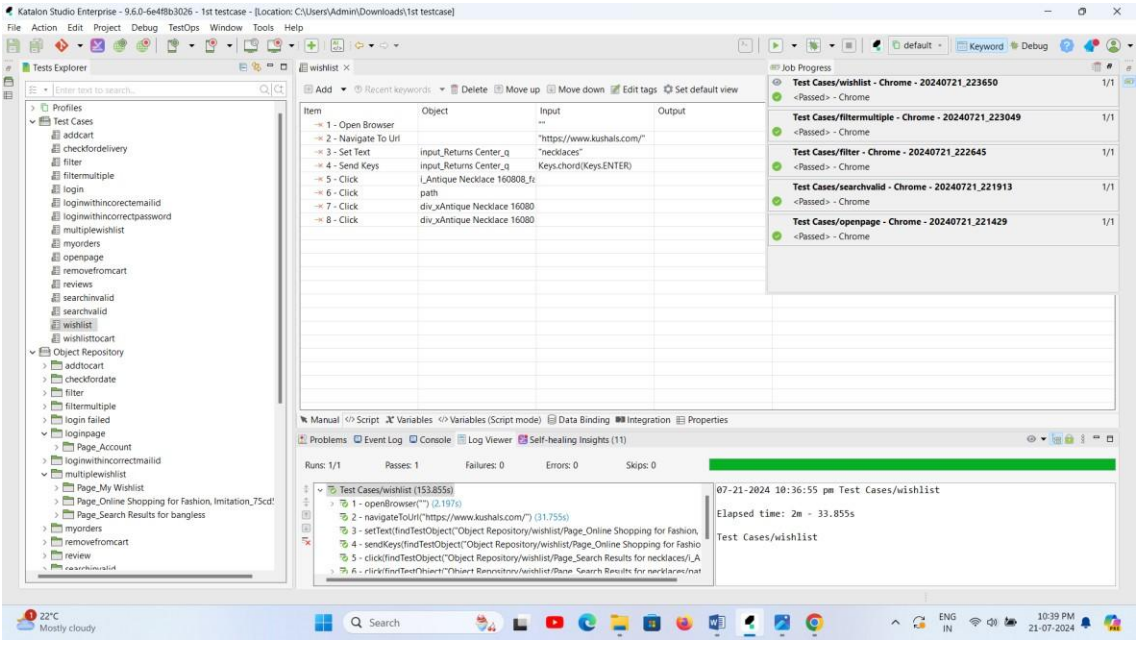

Fig 10: Single item to wish list test case.

This figure captures Katalon Studio where test cases for Kushal's website are developed and executed. It shows the test steps to add single item to wish list, including successful execution results.

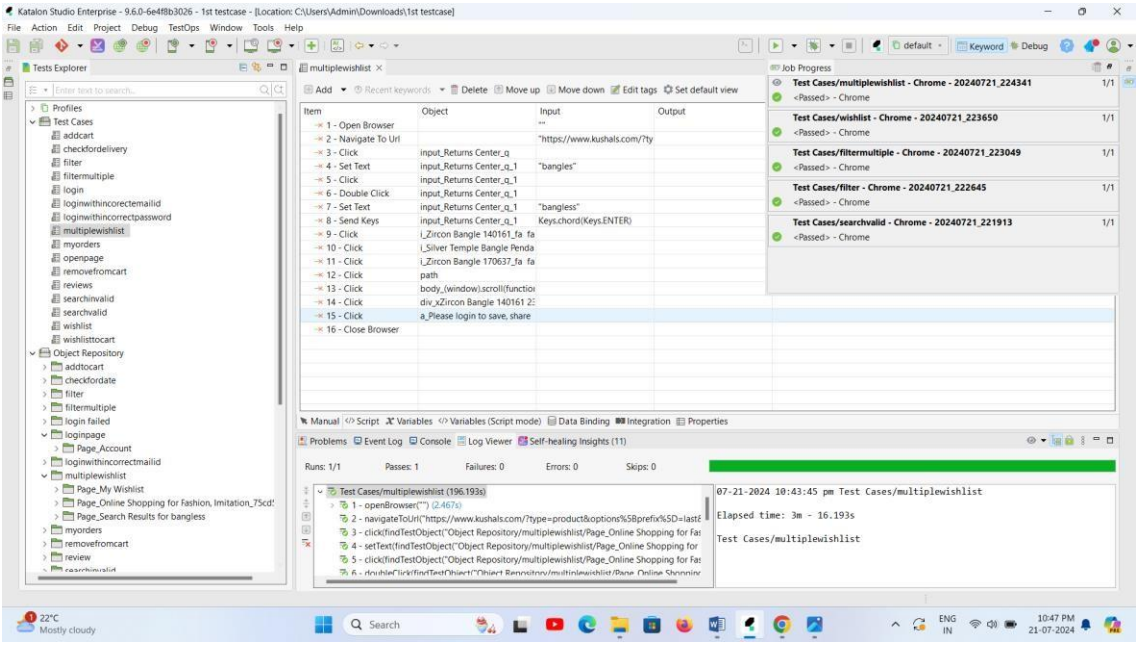

Fig 11: Multiple item to wish list test case.

This figure captures Katalon Studio where test cases for Kushal's website are developed and executed. It shows the test steps to add multiple item to wish list, including successful execution results.

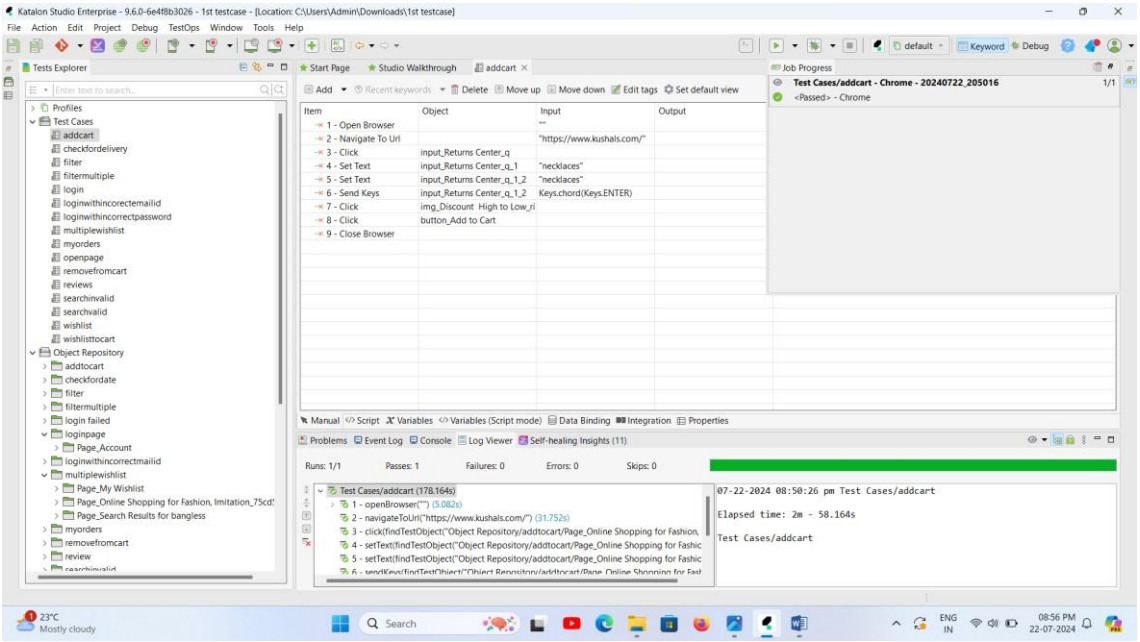

Fig 12: Add item to cart test case.

This figure captures Katalon Studio where test cases for Kushal's website are developed and executed. It shows the test steps to add item to cart, including successful execution results.

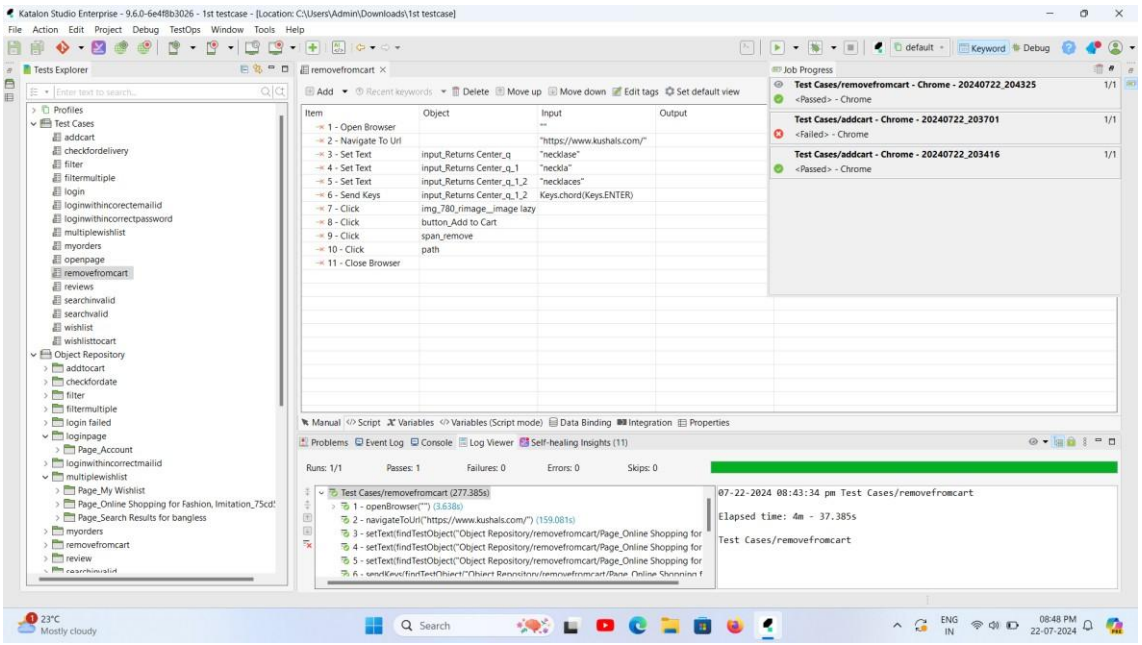

Fig 13: Remove item to cart test case.

This figure captures Katalon Studio where test cases for Kushal's website are developed and executed. It shows the test steps to remove item from cart, including successful execution results.

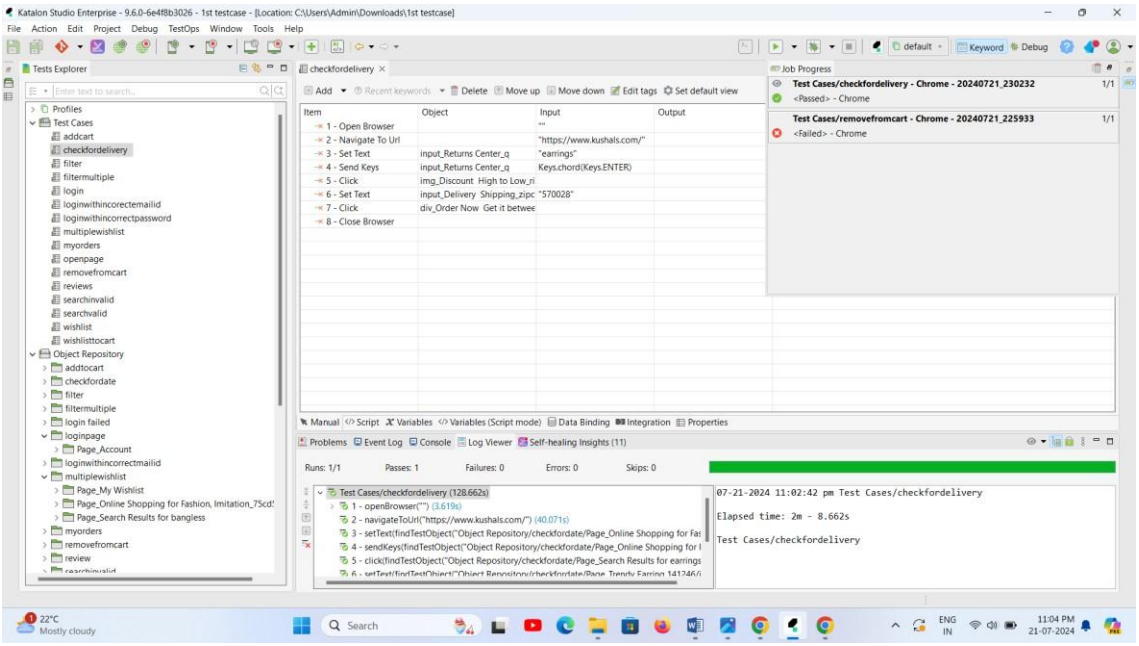

Fig 14: Check for delivery date test case.

This figure captures Katalon Studio where test cases for Kushal's website are developed and executed. It shows the test steps to check for delivery date scenario, including successful execution results.

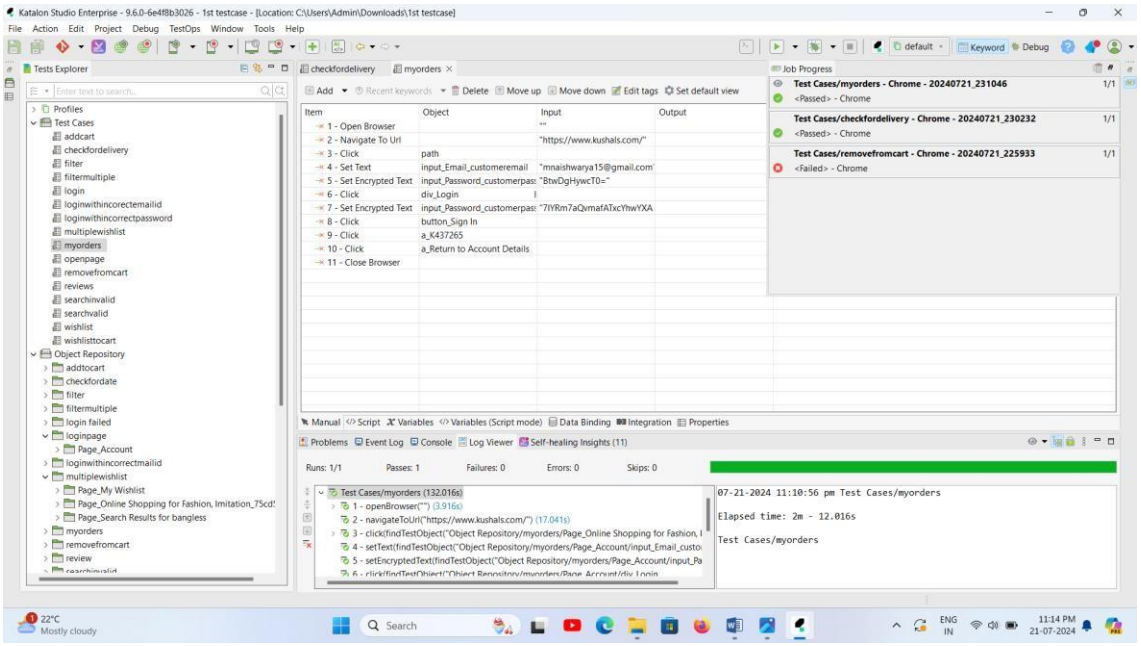

Fig 15: Check for previous orders test case.

This figure captures Katalon Studio where test cases for Kushal's website are developed and executed. It shows the test steps to see previous orders, including successful execution results.

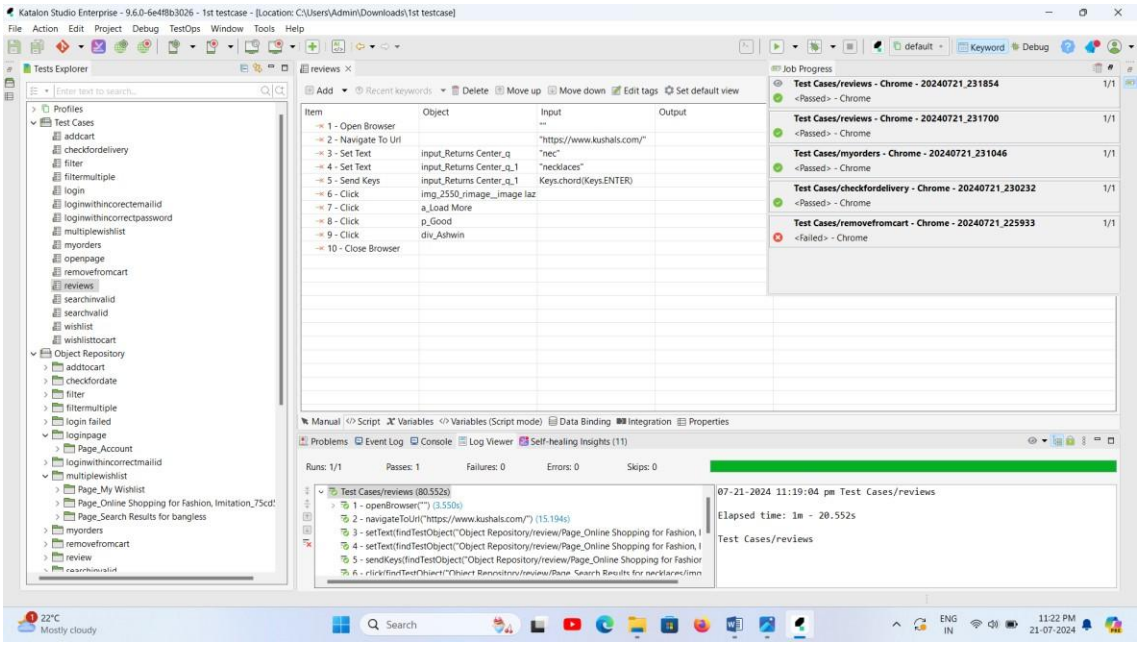

Fig 16: To check order reviews test case.

This figure captures Katalon Studio where test cases for Kushal's website are developed and executed. It shows the test steps to check check order reviews, including successful execution results.

CONCLUSION

The Kushal's Jewelry project represents a significant achievement in e-commerce, aiming to redefine the online shopping experience for jewelry enthusiasts through innovation and user-centric design. Throughout its development lifecycle, Kushal's has prioritized functionality, usability, and performance to ensure a seamless journey for its users. The project's success is underpinned by rigorous testing methodologies, including automated testing with tools like Katalon, which have been instrumental in identifying and rectifying potential issues before deployment. This approach not only ensures the reliability and stability of the platform but also enhances overall user satisfaction by delivering a bug-free experience. By promptly addressing reported issues and implementing enhancements based on user feedback, Kushal's strives to maintain its competitive edge in the dynamic e-commerce landscape.

Kushal's Jewelry project has established itself as a benchmark in the e-commerce industry by consistently delivering exceptional user experiences through continuous innovation and meticulous quality assurance. The integration of cutting-edge technologies and robust testing frameworks has enabled Kushal's to anticipate and adapt to the evolving needs of its customers, ensuring that the platform remains both reliable and user-friendly. By fostering a culture of continuous improvement and leveraging user feedback, Kushal's is able to swiftly implement enhancements that not only address existing challenges but also introduce new features that elevate the shopping experience. This commitment to excellence and responsiveness has solidified Kushal's position as a leader in the online jewelry market, setting new standards for performance, usability, and customer satisfaction.

FUTURE WORK

Future work for Kushal's Jewelry in the e-commerce website domain encompasses several strategic initiatives aimed at enhancing user experience, expanding market reach, and leveraging emerging technologies. Firstly, Kushal's could focus on further personalization by implementing advanced recommendation engines based on user behavior and preferences, thereby tailoring product offerings more accurately. Secondly, integrating augmented reality (AR) and virtual reality (VR) technologies could revolutionize the shopping experience, allowing customers to visualize jewelry pieces in real-world settings before purchase. Additionally, Kushal's could explore integrating social media functionalities to facilitate social shopping experiences, enabling users to share product recommendations and reviews easily. Finally, expanding into new markets and optimizing the mobile shopping experience would further strengthen Kushal's position in the competitive e-commerce landscape.

Future work for Kushal's Jewelry also involves enhancing the platform's sustainability and ethical sourcing initiatives, which are increasingly important to modern consumers. By incorporating blockchain technology, Kushal's could provide transparent supply chain information, assuring customers of the ethical origins of their jewelry. Additionally, implementing AI-driven chatbots and virtual assistants can further improve customer service, offering personalized support and instant responses to queries. Another strategic initiative could be the development of a loyalty program that rewards repeat customers with exclusive offers and discounts, fostering customer retention and brand loyalty. Furthermore, investing in cybersecurity measures will be crucial to protect user data and maintain trust in the platform. By embracing these forward-thinking strategies, Kushal's Jewelry can continue to innovate and thrive in the ever-evolving e-commerce landscape.

REFERENCES

[1] "Katalon Studio Cookbook" by Nanddeep Nachan.

[2] "Automated Testing with Katalon Studio" by Raghav Pal.

[3] "Katalon Studio Tutorial: A Complete Guide" by Raghav Pal.

[4] Online Documentation and Tutorials.

[5] Katalon Studio Documentation.

[6] Katalon Academy.

[7] Katalon Studio YouTube Channel.

[8] HTML Tutorial - https://www.w3schools.com/html

[9] CSS Tutorial - https://www.w3schools.com/Css

[10] JS Tutorial - https://www.w3schools.com/Js

[11] Katalon Tutorial- https://youtube.com/playlist?list=PLhW3qG5bs-L_D4ZePNNjvmIULuu6mBHbu&feature=shared