

PHOTO AND CAMERA RENTALS STORE

EEC 623- SOFTWARE QUALITY ASSURANCE



KARRAR NAWAZ KHAN

CSU ID: 2844629

Email: 2844629@vikes.csuohio.edu

BHUVANESH MANOHARAN

CSU ID: 2861851

Email: 2861851@vikes.csuohio.edu

ANIRUDH GUPTA MYADAM

CSU ID: 2861530

Email: 2861530@vikes.csuohio.edu

TABLE OF CONTENTS

1. INTRODUCTION	1
2. SOFTWARE UNDER TEST	3
3. LIST OF FEATURES THAT ARE TESTED	3
4. TEST DESIGN	5
4.1. COVERAGE CRITERIA	6
4.1.1. Functional Coverage	6
4.1.2. UI Coverage	6
5. TEST CASE SPECIFICATIONS	9
6. TEST RESULTS	23
7. SUMMARY OR CONCLUSION	23
8. APPENDIX	24

1. Introduction

The popularity of e-commerce websites has risen in modern times, with both commercial entities and consumers opting for digital platforms to engage in commercial transactions. Consequently, it is imperative that these websites undergo comprehensive testing and conform to the most stringent standards to ensure a seamless user experience. The present study is centered on the evaluation of an e-commerce platform that provides a diverse array of services, such as camera rental, album design, and cameraman bookings, among other offerings. The principal objective of our study was to conduct comprehensive testing of all elements of the website, encompassing its design, specifications, functionalities, and features, with the aim of ensuring its reliability and mitigating any potential performance-related concerns.

To accomplish this objective, a comprehensive project plan was developed, covering a detailed test plan and test cases to provide direction for our testing activities. The test plan was formulated with the aim of thoroughly assessing and verifying all facets of the website, thereby facilitating the detection of any possible defects or opportunities for enhancement. The utilization of a systematic approach facilitated efficient management and monitoring of our testing activities, thereby guaranteeing the fulfillment of all client-specified requirements for the website.

During the testing phase, a range of testing methodologies were employed, including User Interface Testing and Functionality Testing, to assess the website's overall effectiveness and ease of use. Conducting User Interface Testing was imperative in verifying the website's visual and navigational aspects, thereby facilitating a favorable user experience. The verification of all features and functionalities of a website to ensure that users can effectively engage with the platform and complete their desired transactions is a crucial aspect of Functionality Testing.

After concluding our testing actions, we generated a comprehensive report that expounds on our findings and conclusions. The report presented an overview of the identified issues and areas for enhancement that surfaced during the testing phase. In conclusion, our comprehensive testing activities resulted in an e-commerce website that was functioning, easy to use and fulfilled the specifications.

Finally, in today's competitive digital landscape, the success of an e-commerce website is heavily reliant on its ability to provide a seamless and enjoyable user experience. Our work

demonstrates the necessity of thorough testing in guaranteeing the quality and performance of e-commerce websites and adds to a larger understanding of industry best practices.

2. Software Under Test

Name: camerarentalstore

Website Link: <https://camerarentals.store/>

Version: 1.0.2

Hardware: Personal Computer (Windows or Macintosh Laptop)

Processor - Intel i5

OS: Windows/MacOS

the Xampp Server is software that is used to run PHP files and MySQL databases.

Chrome, Microsoft Edge, or Mozilla Firefox as the System Browser to Access the Front End and Back End Panel (to be executed on the local system)

Languages Used for website development:

Front End language: HTML, CSS, and Java script.

Backend language: PHP

Framework: Laravel

Database: MySQL

3. A list of features that are tested:

1. User Search Category:

- A search bar that allows users to search for services by category.
- List of categories that are easy to navigate and filter to see the website's search function returns accurate and relevant results.

2. Registration:

a) User Registration:

- User-friendly registration form with required fields such as name, email, and password.
- Email verification to confirm the user's email address and ability to edit account information and update profile details.

b) Service Provider Registration:

- User-friendly registration form with required fields such as name, email, and password.

- Ability to upload portfolio and images of the products and services offered.
- Option to add pricing information, availability, and schedule.
- Option to manage payments and manage invoices through the platform.

3. User Login:

- User-friendly login form with fields for email and password.
- Proper validation of email and password fields to ensure that only valid credentials are accepted.
- Error messages should be displayed when invalid credentials are entered.

4. Service Provider Login:

- User-friendly login form with fields for email/username and password.
- Proper validation of email and password fields to ensure that only valid credentials are accepted.
- Error messages should be displayed when invalid credentials are entered.

5. Vendor Details:

- The Vendor Details Page feature provides visitors with access to the details and services offered by a specific vendor by clicking on the vendor's name.
- The expected output of this feature is that the user should be able to access the vendor details page and view the relevant information without encountering any errors.

6. Book Now:

- Ensures successful booking of a specific service with a specific vendor, by providing booking details and confirming the booking.
- The confirmation message verifies the successful completion of the booking process and confirms the service details for the chosen date, time, and location.

7. Review:

- The review functionality on the website allows users to provide feedback on their experience with a specific vendor.
- Users should be able to select a rating and type a message in the review form.
- The system should verify that the review has been successfully submitted and displayed on the vendor's profile page.

8. Feedback:

- Users can fill in all mandatory fields such as Name, Email Address, Phone, Subject, and Message.

9. Product Acceptance:

- The service request processing feature allows vendors to accept or reject service requests and update the booking status accordingly.
- Vendors should be able to update the booking status from "pending" to "completed" after accepting a service request.
- Users should receive a notification when the vendor updates the booking status, providing timely updates on the status of their service request.

10. Payment:

- The payment processing feature allows users to make payments for services booked on manual payment option.

12. Admin:

- Managing manual payments:
To test whether the administrator can accept or reject the manual payments and confirm the bookings.

A list of features that are not tested:

1. Website content:

- This includes all the website content such as text and images that are displayed on the website.

2. Payment gateway (PayPal):

- This includes testing of payment through PayPal and its transaction.

4. Test Design:

- **Requirements Analysis:** The initial stage of testing involves carefully going over both the functional and non-functional requirements for the system. To ensure that all necessary components are covered during the testing process, this aids in understanding the important features and parts of the system that need to be

evaluated. A detailed requirements analysis helps in identifying any gaps or inconsistencies in the requirements, which can be addressed before the testing begins.

- **Test Scenario Identification:** After reviewing the requirements, the next step is to divide them into more manageable chunks. High-level descriptions of the system's behavior under circumstances are found in test scenarios. Testers can make sure that all the system's crucial functionalities and features are assessed during the testing process by identifying test scenarios.
- **Test Case Design:** Designing test cases to verify that the system meets specified requirements, including a unique ID, preconditions, test steps and expected results.
- **Test Case Execution:** After the test cases have been created, the system's behavior will be assessed through their execution. Depending on the system complexity and the resources available, test case execution can be done manually and with the aid of automated testing tool like selenium. Any errors or discrepancies are noted for further investigation and correction.
- **Test Case Reporting:** The final step in the testing process is to document the test case results. This involves creating detailed reports that outline the test cases executed, their outcomes, any defects identified, and the steps taken to address those defects. Test case reporting helps in tracking the progress of the testing efforts, identifying areas for improvement, and ensuring that all critical aspects of the system have been thoroughly tested.

Coverage Criteria:

a) Functional Coverage:

- Comprehensive functional coverage is crucial for reducing risks associated with software deployment.
- Testing all functionalities can help identify potential defects or shortcomings before the product is released.
- The resolution of issues during the testing phase can significantly enhance the quality of the final product and prevent any negative impacts on end-users.
- The provision of comprehensive functional coverage has been found to have a positive impact on user satisfaction and the overall user experience.
- The comprehensive testing of all features and functionalities of a product is crucial in ensuring its proper functioning and alignment with the requirements of its users. This

process is essential in verifying that the product operates as intended and satisfies the expectations of its target audience.

b) UI Coverage:

1. UI coverage refers to the extent to which the graphical user interface of the software or system being tested is covered during testing.
2. The objective of UI coverage is to ensure that the interface is user-friendly and easy to use for end-users.
3. UI coverage includes testing of all elements on the user interface, such as buttons, menus, icons, and other graphical elements, to ensure that they function correctly and provide a positive user experience.

Possible Test scenarios:

1. Search for services by category or keyword:
 - Verify that the search bar is visible and properly positioned.
 - Verify that the search results are displayed accurately based on the user's input.
 - Verify that the user can filter the search results by category.
2. Login/registration page:
 - Verify that the login/registration form fields are properly labeled and positioned.
 - Verify that the error messages are displayed when user enters incorrect credentials or leaves a required field blank.
 - Verify user/Service provider enter valid credentials.
3. Vendor details:
 - Verify that user can click on a vendor's name to access the vendor details page.
 - Verify that the system displays the vendor details page correctly.
 - Verify that the user can view the vendor's details and services without encountering any errors.
 - Verify that the user can click on a service offered by the vendor.
 - Verify that the system displays the service details page correctly.
 - Verify that the user can view the service details without encountering any errors.

4. Booking a service:

- Verify that user can select the desired rental service.
- Verify that user can select the date and time for the rental service.
- Verify that user can add additional services to the rental service.
- Verify that user can view the total cost of the rental service before booking.
- Verify that user can successfully book the rental service and receive. Confirmation

5. Payment for rental services:

- Verify that user can select the desired payment method.
- Verify that user can enter the payment details.
- Verify that user can view the total cost of the rental service before making payment.
- Verify that user can make payment successfully and receive confirmation.

6. Providing feedback or contacting support:

- Verify that user can access the feedback/contact support page.
- Verify that user is able to fill out the feedback/contact support form.
- Verify that user is able to submit the feedback/contact support form successfully.
- Verify that user receives a confirmation message after submitting the form.

7. Reviewing and rating vendors:

- Verify that the user can view the list of vendors.
- Verify that the user can select a vendor and view their details.
- Verify that the user can leave a review and rating for the vendor.
- Verify that the review and rating is displayed on the vendor's profile.

8. Vendor approval of service bookings:

- Verify that the vendor can view the booking requests.
- Verify that the vendor can approve or reject the booking request.
- Verify that the user receives a notification about the vendor's decision.

9. Service booking form with input validation:

- Verify that the form fields are properly labeled and positioned.
- Verify that the user is unable to submit the form if any required field is left blank
- Verify that the user is unable to enter invalid input in the form fields (such as letters in a numeric field)

10. User dashboard with booking history and payment status:

- Verify that the user is able to access their dashboard after logging in
- Verify that the user is able to view their booking history and payment status
- Verify that the dashboard is visually appealing and user-friendly.

11. Admin payment approval:

- Verify that admin can approve the payment request from user.
 - Verify that the admin can accept the payment request.
 - Verify that admin confirms the booking request and update the status to “confirmed”.
 - Verify that the system verifies that the booking status is updated correctly, and that the user receives the appropriate notification.

5. Test Case Specifications

Test case Identifier	TC 01
Test Items/feature	User Registration (User)
Test Steps	<ol style="list-style-type: none">1. Click registration button in the home page.2. Enter First Name: Bhuvanesh3. Enter Last Name: Mano4. Enter Email: bhuvaneshcsu47@gmail.com5. Enter Password: Bhuvaneshqwer@0986. Enter confirm password: Bhuvaneshqwer@0987. Click “Register Now” button.8. Enter the verification code sent to the Gmail id.9. Click “Verify Now” button.
Expected Output	User should be directed to the dashboard page.
Precondition	User should enter valid email id.
Inter-case Dependencies	N/A
Pass/Fail	PASS

Test case Identifier	TC 02
Test Items/feature	User Log-in (User)
Test Steps	<ol style="list-style-type: none"> 1. Navigate to the login page. 2. Enter User Id: bhuvaneshcsu47@gmail.com 3. Enter Password = Bhuvaneshqwer@098 4. Click “Login Now”.
Expected Output	User should log into the user dashboard.
Precondition	User should be registered first.
Inter-case Dependencies	TC001 (User Registration (User))
Pass/Fail	PASS

Test case Identifier	TC 03
Test Items/feature	User Registration (Service Provider)
Test Input	<ol style="list-style-type: none"> 1. Navigate to the service provider registration page. 2. Enter First Name: Karrar 3. Enter Last Name: K 4. Enter Email: bhuv9812@gmail.com 5. Enter Password: Bhuvanesh_123 6. Enter confirm password: Bhuvanesh_123 7. Click on to “Register Now” button. 8. Enter the verification code sent to the Gmail id. 9. Click “Verify Now” button. 10. In the dashboard page, enter the service type and attach the image of the product or service.

	<p>11. Complete the address and contact details.</p> <p>12. Choose the schedule and click save button.</p>
Expected Output	Service provider should be directed to the dashboard page and service and product images should be displayed under the vendor details in the expert's page.
Precondition	Service provider should enter valid email id.
Inter-case Dependencies	N/A
Pass/Fail	PASS

Test Case Identifier	TC 04
Test Items/feature	User Log-in (Service Provider)
Test Steps	<ol style="list-style-type: none"> 1. Navigate to the log-in page. 2. Enter User id: bhuv9812@gmail.com 3. Enter Password=Bhuvanesh_123 4. Click "Login Now".
Expected Output	User (Service Provider) should log into the service provider dashboard.
Pre-Condition	Service provider should be registered first.
Inter-case Dependencies	TC003 (User Registration (Service Provider))
Pass/Fail	PASS

Test Case Identifier	TC 05
Test Items/feature	User Login with invalid email
Test Steps	<ol style="list-style-type: none"> 1. Navigate to the login page. 2. Enter User id: bhuv98122@gmail.com 3. Enter Password= Bhuvanesh_123 4. Click "Login Now".
Expected Output	

	Error message “No user associated with this email address” is displayed.
Pre-Condition	User should be registered on the website first.
Inter-case Dependencies	TC002 (User Login)
Pass/Fail	PASS

Test Case Identifier	TC 06
Test Items/feature	User Login with invalid password
Test Steps	<ol style="list-style-type: none"> 1. Navigate to the login page. 2. Enter User Id: bhuv9812@gmail.com 3. Enter Password= bhuvanesh_ 4. Click “Login Now”.
Expected Output	Error message "Invalid credentials" is displayed.
Pre-Condition	User should be registered first.
Inter-case Dependencies	TC002 (User Login)
Pass/Fail	PASS

Test Case Identifier	TC 07
Test Items/feature	Admin Login.
Test Steps	<ol style="list-style-type: none"> 1. Navigate to the admin login page. 2. Enter User id = admin@gmail.com 3. Enter Password = 123 4. Click “Login Now”.
Expected Output	Admin dashboard should be displayed.
Pre-Condition	Admin account exists.
Inter-case Dependencies	N/A
Pass/Fail	PASS

Test Case Identifier	TC 08
Test Items/feature	Forgot Password
Test Steps	<ol style="list-style-type: none"> 1. Navigate to login page. 2. Click on the option, "Click to reset." 3. Enter your email address to be sent a verification code. 4. Enter the Verification code. 5. Enter new password and confirm the password.
Expected Output	User should see "password reset Successfully"
Pre-Condition	User should be registered
Inter-case Dependencies	N/A
Pass/Fail	PASS

Test Case Identifier	TC 09
Test Items/feature	Vendors (Alex Fate)
Test Steps	<ol style="list-style-type: none"> 1. Navigate to the vendors page by clicking vendors tab. 2. Then, select the Alex Fate among the other vendors.
Expected Output	User should be directed to a page showing the details and services offered by the selected vendor.
Pre-Condition	User should be on the camerarentals.store homepage to see the Vendors details.
Inter-case Dependencies	N/A
Pass/Fail	PASS

Test Case Identifier	TC 10
Test Items/feature	Book Now Exception
Test Steps	<ol style="list-style-type: none"> 1. Navigate to the home page. 2. Click Vendors tab.

	<ol style="list-style-type: none"> 3. Select Alex Fate as the vendor option. 4. Click 4K Video Camera Camcorder, 48MP 60FPS service. 5. Click Book Now button. 6. Fill in the required booking details (start date, hours, start time, and location). 7. Click on the "Book Service" button.
Expected Output	User should see the message as "Already booked service "
Pre-Condition	User should be on the camerarentals.store homepage.
Inter-case Dependencies	N/A
Pass/Fail	PASS

Test Case Identifier	TC 11
Test Items/feature	Book now (Alex Fate)
Test Steps	<ol style="list-style-type: none"> 1. Navigate to the home page. 2. Click Vendors tab. 3. Select Alex Fate as the vendor option. 4. Click 4K Video Camera Camcorder, 48MP 60FPS service. 5. Click Book Now button. 6. Fill in the required booking details (start date, hours, start time, and location). 7. Click on the "Book Service" button.
Expected Output	User should see a message saying "Successfully booked the service."
Pre-Condition	<ul style="list-style-type: none"> • User should be on the camerarentals.store homepage. • The Vendors tab should be visible and functional. • User should be logged in before booking.

Inter-case Dependencies	TC 11(Vendors)
Pass/Fail	PASS

Test Case Identifier	TC 12
Test Items/feature	Book now (paisley Mule)
Test Steps	<ol style="list-style-type: none"> 1. Navigate to the home page. 2. Click Vendors tab. 3. Click Paisley Mule vendor. 4. Click Album Design by Paisley service. 5. Click Book Now button. 6. Fill in the required booking details (start date, hours, start time, and location). 7. Click on the "Book Service" button. 8. Verify that the message "Successfully booked the service" is displayed on the screen.
Expected Output	User should see a message saying, "Successfully booked the service."
Pre-Condition	<ul style="list-style-type: none"> • User should be on the camerarentals.store homepage. • The Vendors tab should be visible and functional. • User should be logged in before booking.
Inter-case Dependencies	TC 11 (Vendors)
Pass/Fail	PASS

Test Case Identifier	TC 13
Test Items/feature	Book now (Ryan Clark)
Test Steps	<ol style="list-style-type: none"> 1. Navigate to home page.

	<ol style="list-style-type: none"> Click Vendors tab. Click Ryan as vendor. Click Ryan as Camera Man service. Click Book Now button. Fill in the required booking details (start date, hours, start time, and location). Click on the "Book Service" button. Verify that the message "Successfully booked the service" is displayed on the screen.
Expected Output	User should see a message saying, "Successfully booked the service."
Pre-Condition	<ul style="list-style-type: none"> User should be on the camerarentals.store homepage. The Vendors tab should be visible and functional. User should be logged in before booking.
Inter-case Dependencies	11(Vendors)
Pass/Fail	PASS

Test Case Identifier	TC 14
Test Items/feature	Book now with empty message field
Test Steps	<ol style="list-style-type: none"> Navigate to home page. Click Vendors tab. Click Ryan as vendor. Click Nikon Digital Camera Z 5 Kit with NIKKOR Z 24-200mm service. Click Book Now button. Fill in the required booking details (start date, hours, start time, and location).

	<ol style="list-style-type: none"> Click on the "Book Now" button. Verify that an error message is displayed indicating that the booking cannot be completed due to missing input
Expected Output	User should see a message saying "The message field is required"
Pre-Condition	<ul style="list-style-type: none"> User should be on the camerarentals.store homepage. The Vendors tab should be visible and functional. User should be logged in before booking.
Inter-case Dependencies	TC 11(Vendors)
Pass/Fail	PASS

Test Case Identifier	TC 15
Test Items/feature	Review
Test Input	<ol style="list-style-type: none"> Click on the "Vendors" tab on the homepage. Select "Ryan" from the list of vendors. Click write a review button. Select ratings from ratings drop down box. Type "Service is Good" Click Submit button. Verify that user ratings and user message is displayed under review section with stars.
Expected Output	User should be able to see their ratings and message "Service is Good" under the reviews section.
Pre-Condition	<ul style="list-style-type: none"> User should be on the camerarentals.store homepage. The Vendors tab should be visible and functional.
Inter-case Dependencies	TC 11(Vendors)
Pass/Fail	PASS

Test Case Identifier	TC 16
Test Items/feature	Feedback
Test Steps	<ol style="list-style-type: none"> 1. Select the Contact Us/ Feedback tab. 2. Then, type Name =" bhuvi". 3. Email Address =" kumar1234@gmail.com". 4. Type phone ="12345678". 5. Type subject=" service". 6. Type message=" good service". 7. Click send message button.
Expected Output	<p>User should be able to see the message as "contact us successfully".</p> <p>User should receive the confirmation mail.</p>
Pre-Condition	The feedback/contact us page should be accessible to the user and all the mandatory fields should be present.
Inter-case Dependencies	N/A
Pass/Fail	PASS

Test Case Identifier	TC 17
Test Items/feature	Product acceptance
Test Steps	<ol style="list-style-type: none"> 1. Vendor login to service provider dashboard. 2. Then, vendor should accept the service which is requested from user. 3. Finally, status change from pending to completed should be verified.
Expected Output	Vendor should be able to see the status changed to completed".
Pre-Condition	A user should book the service before the vendor can approve it.
Inter-case Dependencies	N/A

Pass/Fail	PASS
------------------	------

Test Case Identifier	TC 18
Test Items/feature	Payment
Test Steps	<ol style="list-style-type: none"> 1. Select dashboard. 2. Click bookings> all bookings. 3. Next, user clicks Pay Bill option then user should select pay via bank then type details=" Test_input". 4. Transaction number="1230945274". 5. User should select an attachment and clicks send proof for payment button. 6. Click on Pay Now.
Expected Output	User should be able to see the message as "payment request has been submitted successfully".
Pre-Condition	The bank account details provided by the user must be valid.
Inter-case Dependencies	N/A
Pass/Fail	PASS

Test Case Identifier	TC 19
Test Items/feature	Admin (Managing Manual payment and bookings)
Test steps	<ol style="list-style-type: none"> 1. Admin should click on manual payments button. 2. Admin should be able to see the accept or deny buttons.

	3. Next, admin should click manage booking > all booking tab. 4. Admin should click on complete button to finish the booking.
Expected Output	User should see the booking status as completed.
Pre-Condition	<ul style="list-style-type: none"> • User should do the booking first. • Next User should do the manual payment.
Inter-case Dependencies	TC 18
Pass/Fail	PASS

Test Case Identifier	TC 20
Test Items/feature	Logout
Test Input	Click logout button.
Expected Output	User is redirected to the home page.
Pre-Condition	User must be logged in to the system.
Inter-case Dependencies	N/A
Pass/Fail	PASS

Test Case Identifier	TC 21
Test Items/feature	Search category
Test steps	1. Click on search category in the search field. 2. Click on camera rental option in the drop-down list. 3. Click on search button.
Expected Output	User should be able to see various camera rental options
Pre-Condition	User must be logged in to the system first.
Inter-case Dependencies	TC 08
Pass/Fail	FAIL

Traceability Matrix:

Automation for invalid credentials:

The screenshot displays a Selenium IDE interface on the left and a web browser on the right. The Selenium IDE shows two test cases for user login functionality. The first test case, 'In [1]', is a user log-in functionality that uses Selenium WebDriver to launch Firefox, navigate to the login page, find the email and password fields, send keys, and click the login button. The second test case, 'In [3]', is another user log-in functionality that uses Selenium WebDriver to launch Firefox, navigate to the login page, find the email and password fields, send keys, and click the login button. The web browser shows the login page of 'camerarentalstore@gmail.com'. The page has a header with social media links and navigation links. The main content area has a 'Login' section with a form for email and password. Below the form is a 'Login Now' button and a link to 'Forgot Password? Click Here To Reset'. The footer has sections for 'About Us', 'Categories', 'Quick Links', and 'Contact Info'.

```
In [1]: ## User Log-In functionality
from selenium import webdriver
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.common.by import By
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC

# Launch Firefox
driver = webdriver.Chrome()

# Navigate to the login page
driver.get("https://camerarentals.store/user/login")

# Find the "Email" field by name and type in an email address
email_field = driver.find_element(By.NAME, "email")
email_field.send_keys("bhuvanshmarichar1@gmail.com")

# Find the "Password" field by name and type in a password
password_field = driver.find_element(By.NAME, "password")
password_field.send_keys("Thur123")

wait = WebDriverWait(driver, 10)
ClickNow = wait.until(EC.element_to_be_clickable((By.XPATH, "//*[@id='recaptcha']")))

# click on the recaptcha element
ClickNow.click()

In [3]: ## User Log-In functionality
from selenium import webdriver
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.common.by import By
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC

# Launch Firefox
driver = webdriver.Chrome()

# Navigate to the login page
driver.get("https://camerarentals.store/user/login")

# Find the "Email" field by name and type in an email address
email_field = driver.find_element(By.NAME, "email")
email_field.send_keys("bhuvanshmarichar1@gmail.com")

# Find the "Password" field by name and type in a password
password_field = driver.find_element(By.NAME, "password")
password_field.send_keys("123")

wait = WebDriverWait(driver, 10)
ClickNow = wait.until(EC.element_to_be_clickable((By.XPATH, "//*[@id='recaptcha']")))

# click on the recaptcha element
ClickNow.click()
```

Automation for unregistered user:

The screenshot displays a Selenium IDE interface on the left and a web browser on the right. The Selenium IDE shows two test cases for user login functionality. The first test case, 'In [1]', is a user log-in functionality that uses Selenium WebDriver to launch Firefox, navigate to the login page, find the email and password fields, send keys, and click the login button. The second test case, 'In [4]', is another user log-in functionality that uses Selenium WebDriver to launch Firefox, navigate to the login page, find the email and password fields, send keys, and click the login button. The web browser shows the login page of 'camerarentalstore@gmail.com'. The page has a header with social media links and navigation links. The main content area has a 'Login' section with a form for email and password. Below the form is a 'Login Now' button and a link to 'Forgot Password? Click Here To Reset'. The footer has sections for 'About Us', 'Categories', 'Quick Links', and 'Contact Info'.

```
In [1]: ## User Log-In functionality
from selenium import webdriver
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.common.by import By
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC

# Launch Firefox
driver = webdriver.Chrome()

# Navigate to the login page
driver.get("https://camerarentals.store/user/login")

# Find the "Email" field by name and type in an email address
email_field = driver.find_element(By.NAME, "email")
email_field.send_keys("bhuvanshmarichar1@gmail.com")

# Find the "Password" field by name and type in a password
password_field = driver.find_element(By.NAME, "password")
password_field.send_keys("Thur123")

wait = WebDriverWait(driver, 10)
ClickNow = wait.until(EC.element_to_be_clickable((By.XPATH, "//*[@id='recaptcha']")))

# click on the recaptcha element
ClickNow.click()

In [4]: ## User Log-In functionality
from selenium import webdriver
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.common.by import By
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC

# Launch Firefox
driver = webdriver.Chrome()

# Navigate to the login page
driver.get("https://camerarentals.store/user/login")

# Find the "Email" field by name and type in an email address
email_field = driver.find_element(By.NAME, "email")
email_field.send_keys("bhuvanshmarichar1@gmail.com")

# Find the "Password" field by name and type in a password
password_field = driver.find_element(By.NAME, "password")
password_field.send_keys("123")

wait = WebDriverWait(driver, 10)
ClickNow = wait.until(EC.element_to_be_clickable((By.XPATH, "//*[@id='recaptcha']")))

# click on the recaptcha element
ClickNow.click()
```

Test Results:

Test Case Identifier	Result	Execution
TC01	PASS	100%
TC02	PASS	100%
TC03	PASS	100%
TC04	PASS	100%
TC05	PASS	100%
TC06	PASS	100%
TC07	PASS	100%
TC08	PASS	100%
TC09	PASS	100%
TC10	PASS	100%
TC11	PASS	100%
TC12	PASS	100%
TC13	PASS	100%
TC14	PASS	100%
TC15	PASS	100%
TC16	PASS	100%
TC17	PASS	100%
TC18	PASS	100%
TC19	PASS	100%
TC20	PASS	100%
TC21	FAIL	0%

7. Conclusion

In conclusion, the testing of the camerarentals.store website was a comprehensive effort to ensure that the site's design, functionalities, pages, and features were thoroughly tested to ensure their sanity and prevent any performance issues. The project plan included a detailed test plan and test cases to guide testing efforts, and the team conducted User Interface Testing, Functionality Testing to ensure that the website was functional, user-friendly, and met the requirements of the client.

The tested features included User Search Category, Registration, Login, Vendor Details, Book Now, Review, Feedback, Product Acceptance, Payment, and Admin management of manual payments. These features were tested to ensure that they worked correctly and provided the expected output without encountering any errors.

While the testing was comprehensive, some features were not tested, such as website content and payment gateway (PayPal) transactions. This is because these features fall outside the scope of functional testing and require separate testing efforts.

Overall, the testing efforts for the camerarentals.store website were successful, and the site was found to be functional and user-friendly. The test results and findings were documented and shared with the client to ensure that any issues or bugs were resolved, and the website was launched successfully.

8. Appendix

A. DEFECT REPORT:

Test case ID	TC21
Test Feature	Search Category
Summary	User should be able to see various rental options to rent the camera.
Description	First User should click on the search category dialog box and then click camera rental under the drop-down list and should click on the search button to see various options and pricings to rent the camera, instead vendor details are displayed after clicking search button.
Actual Result	On the website, Expert page with list of all vendors is displayed
Expected Result	User should see various camera rental options and pricings.
Severity	LOW

