

## Selenium Automation Script & regression plane

### Why Selenium?

- It aligns with previous automation scripts.
- Best for web-based applications like Tanami Capital's payment module.
- Supports CI/CD integration for continuous testing.

### Regression Testing Plan

Regression testing ensures that existing functionality is not broken by new changes.

Test Type	Scope	Frequency	Tools
<b>UI Tests</b>	Login, Payment, Refund	Before every release	Selenium
<b>API Tests</b>	Payment API, Authentication API	Daily	Postman/Newman
<b>Security Checks</b>	KYC/AML, Unauthorized Access	Monthly	Manual & Automated

## Automated Test Script (python)

```
1  from selenium import webdriver
2  from selenium.webdriver.common.by import By
3  from selenium.webdriver.common.keys import Keys
4  from selenium.webdriver.support.ui import WebDriverWait
5  from selenium.webdriver.support import expected_conditions as EC
6  import json
7  import logging
8  import time
9
10 # Configure logging
11 logging.basicConfig(filename="test_logs.log", level=logging.INFO, format="%(asctime)s - %(levelname)s - %(message)s")
12
13 # Load test data from a JSON file
14 with open("test_data.json", "r") as file:
15     test_data = json.load(file)
16
17 username = test_data["username"]
18 password = test_data["password"]
19 card_number = test_data["card_number"]
20 expiry_date = test_data["expiry_date"]
21 cvv = test_data["cvv"]
22
23 # Initialize WebDriver
24 driver = webdriver.Chrome()
25 wait = WebDriverWait(driver, 10)
26
27 def test_successful_payment():
28     """Automates a successful payment transaction with improved checks and logging."""
29     try:
30         # Step 1: Open the Payment Page
31         driver.get("https://tanamicapital.com/payment")
32         logging.info("Opened Payment Page")
33
34         # Step 2: Log in
35         wait.until(EC.presence_of_element_located((By.ID, "username"))).send_keys(username)
36         driver.find_element(By.ID, "password").send_keys(password, Keys.RETURN)
37         logging.info("User logged in successfully")
38         time.sleep(2)
39
40         # Step 3: Enter Payment Details
41         driver.find_element(By.ID, "card_number").send_keys(card_number)
42         driver.find_element(By.ID, "expiry_date").send_keys(expiry_date)
43         driver.find_element(By.ID, "cvv").send_keys(cvv)
44         logging.info("Entered payment details")
45
46         # Step 4: Submit Payment
47         driver.find_element(By.ID, "submit_payment").click()
48         time.sleep(3)
49         logging.info("Submitted payment")
50
51         # Step 5: Validate Payment Success Message
52         success_message = driver.find_element(By.ID, "success_message").text
53         assert "Payment Successful" in success_message, "Test Failed: Payment did not go through."
54         logging.info("Payment transaction was successful.")
55
```

```
55
56     # Step 6: Validate Transaction ID
57     transaction_id = driver.find_element(By.ID, "transaction_id").text
58     assert transaction_id != "", "Test Failed: No transaction ID generated."
59     logging.info(f"Transaction ID: {transaction_id}")
60
61     except Exception as e:
62         logging.error(f"Test Failed: {e}")
63     finally:
64         # Close Browser
65         driver.quit()
66         logging.info("Browser closed.")
67
68     # Run the test
69     test_successful_payment()
```