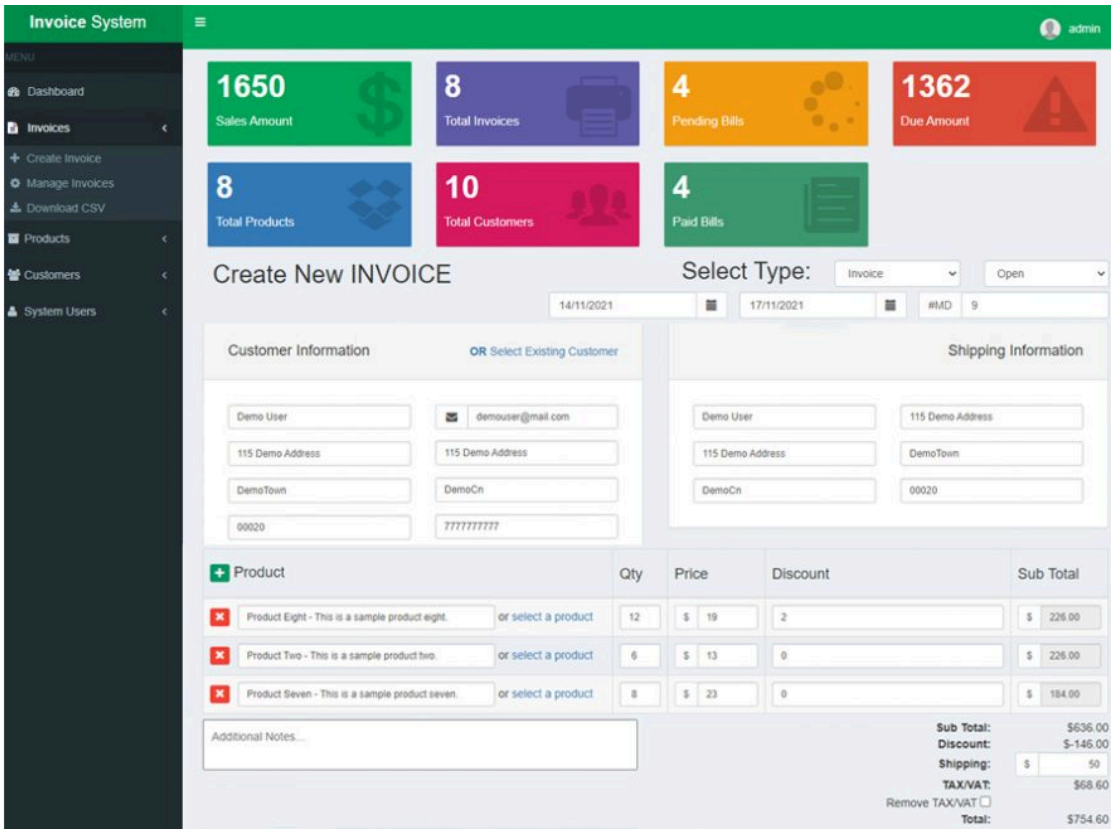
Reseller Panel Web App Automation Testing



Testing Artifact

==========

= **Automated Test Cases Executed**: 6050 =

= **Test Automation Frameworks Utilized**: 1 =

= **Automated Bug Reporting Instances**:5 =

= **Automated Resource Validations**: 50 =

= **Automation Engineers Involved**: 3 =

= **Test Execution Reporting Cycle**: Weekly =

==========

Project Description

About ten percent of people suffer from some kind of challenge and disability. Through proper education and training, they can contribute significantly without being a burden to society.

In developed countries, the government is responsible for their education, training and rehabilitation. Our government has also taken many steps to improve the living standards of these individuals with disabilities.

So to allign with them Bangladesh Government also intends to develop some special mobile games for autistic children that will help bring them back into the mainstream.

A reseller billing panel web app is a comprehensive tool designed to facilitate the management of a reseller's hosting business. It provides an all-in-one platform where resellers can handle various aspects of their operation, including store management, client management, domain services, and billing processes.

Key Features:

- **Easy Store Management**: Resellers can manage multiple stores from a single account, each with unique settings for currency, language, data center preferences, and custom branding options
- **Intuitive Interface**: The panel offers quick navigation, detailed menu labels, and an easy-to-follow design structure, ensuring a zero learning curve for new users
- **Automated Billing**: It automates all billing operations, including invoicing, payment processing, and pro-rata billing for account upgrades or downgrades
- **Client Management**: The panel allows for efficient filtering and management of customers based on account status, and provides the ability to access clients' billing sections or files
- **Service Offerings**: Resellers can configure their offerings, update prices, or create new hosting packages with just a few clicks, selling services like cloud hosting, VPS, dedicated servers, and domain names
- **Coupon Generation**: A built-in coupon generator enables the creation of sale coupons, allowing resellers to offer discounts to their customers
- Branding Options: The control panel comes with multiple branding options to match the look and feel of the reseller's online store

Skills Used:

- **Automation Frameworks**: Selenium WebDriver
- **Design Pattern**: Page Factory, Data Driven Framework
- **Programming Languages**: Java
- **Tools**: Jenkins for CI/CD, Redmine for bug tracking for test case management

Challenges & Solutions:

One of the main challenges was handling dynamic content within the app, which could change based on user preferences and past behavior. To address this, I implemented custom wait strategies in Selenium to dynamically locate and interact with elements.

Results:

The automation suite reduced the regression testing time by 70%, allowing for more frequent releases and a higher level of confidence in the stability of the app's new features.

Code Sample:

```
// Example of a Selenium WebDriver test script for login
functionality

public class LoginTest {

      private WebDriver driver;



      @Before

      public void setUp() {
```

```
        driver = new ChromeDriver();
        driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);
        driver.get("https://example-e-commerce-app.com");
    }

    @Test
    public void testSuccessfulLogin() {
        driver.findElement(By.id("username")).sendKeys("testuser");
        driver.findElement(By.id("password")).sendKeys("securepassword");
        driver.findElement(By.id("loginButton")).click();
        assertTrue("User should be logged in", driver.findElement(By.id("logoutButton")).isDisplayed());
    }

    @After
    public void tearDown() {
        driver.quit();
    }
}
```

```
package com.add_sub_event_code;


import java.io.IOException;


import org.testng.annotations.AfterMethod;

import org.testng.annotations.BeforeMethod;

import org.testng.annotations.Ignore;

import org.testng.annotations.Test;


import com.admin.qa.function.AddSubEventCode;

import com.crm.qa.base.TestBase;

import com.crm.qa.pages.LoginPage;

import com.crm.qa.util.TestUtils;


public class AddEventActionCodeFieldTest extends TestBase{

	LoginPage loginPage;

	TestUtils testUtils;

	AddSubEventCode addSubEventCode;

	//Initializing PageFactory

	public AddEventActionCodeFieldTest() {
```

```
	super();   //Call the Constructor of the Super class - TestBase
}
@BeforeMethod
public void setUp() {
	initialization();
	loginPage = new LoginPage();
	testUtils = new TestUtils();
}



public void addSubEventCodeCommon(String Code) throws IOException, InterruptedException {
	addSubEventCode = loginPage.addSubEventCodeLogin(props.getProperty("username"),pro ps.getProperty("password")); //login to the system
	addSubEventCode.openAddSubEventCodeForm();// open the Add Sub Event Code Page.
	addSubEventCode.typetoPaymentMethodSelectField();//select from the payment method
	addSubEventCode.typetoEventCodeSelectField();
	addSubEventCode.typeCodeFieldParam(Code);// taking input from the Code
```

```
        addSubEventCode.typeShortDescriptionsParam("this code will
    expire soon");// taking inputs from the Short Descriptions

        addSubEventCode.clickTIsActiveCheckboxField(); //  check
    box for active status

        //addSubEventCode.clickONSubmitButton(); //click on submit
    button



    }
}
```

Testing Methodologies

- Functional Testing: Verified that all VPN functionalities met the specified requirements.
- Regression Testing: Ensured new updates did not adversely affect existing features.
- Performance Testing: Assessed the app's speed and stability across different network conditions.
- Security Testing: Conducted vulnerability assessments to safeguard user data.

Reflection:

This project taught me the importance of robust test planning and the need for continuous learning to keep up with the latest updates in automation tools and practices.

Client Feedback

> *"Working with Nazmul Hasan on our app automation testing project was a remarkable experience. Their expertise in Selenium and Java made a significant difference in the efficiency of our testing process. They were not only adept at writing clean, effective test scripts but also demonstrated exceptional problem-solving skills when faced with complex testing scenarios. Communication was seamless, and the project was delivered ahead of schedule, which speaks volumes about their professionalism and dedication. I highly recommend Nazmul Hasan for any automation testing needs and look forward to future collaborations."- **Riaz Hasan, CTO of Reseller Web panel project***

Visual Documentation

- Screenshot.