

## Day 7 - Test case solution document

### Code Used:

#### BaseTest.java

```
package com.ibm.test;

import java.io.File;
import java.io.IOException;
import java.util.Date;
import org.apache.commons.io.FileUtils;
import org.openqa.selenium.OutputType;
import org.openqa.selenium.TakesScreenshot;
import org.testng.Assert;
import org.testng.Reporter;
import org.testng.annotations.DataProvider;
import org.testng.annotations.Optional;
import org.testng.annotations.Parameters;
import org.testng.annotations.Test;
import com.ibm.pages.CategoryPage;
import com.ibm.pages.CouponPage;
import com.ibm.pages.DashboardPage;
import com.ibm.pages.DriverLaunch;
import com.ibm.pages.LoginPage;
import com.ibm.pages.ProductsPage;
import com.ibm.pages>ReturnsPage;
import com.ibm.pages.SettingsPage;
import com.ibm.pages.ShippingPage;
import com.ibm.pages.TabsPage;
import com.ibm.pages.UserPage;
import com.ibm.utilities.ExcelReader;

public class BaseTest extends DriverLaunch
{
    @Test(testName="EditProductValidation",dataProvider="EditProduct")
    public void Day8TestCase(String old_name,String ProNewName,String
    ProMetaTitle,String ProModel,String ProGST,String ProPrice,String
    ProSpDisc,String ProSort,String ProStatus,String ProTabs,String
    ProCategories,String ProImgPath) throws InterruptedException

    {
        LoginPage login=new LoginPage(driver,wait);
        DashboardPage dash = new DashboardPage(driver,wait);
        ProductsPage prod=new ProductsPage(driver,wait);
        UserPage user=new UserPage(driver,wait);
        String adminurl=hashData.get("adminurl");
        String username=hashData.get("username");
        String password=hashData.get("password");
        String userurl=hashData.get("userurl");
```

```

//Navigating to Admin page of atozgroceries.com
driver.get(adminurl);
//logging into the admin page
login.EnterEmail(username);
login.EnterPassword(password);
login.clickLogin();

dash.OpenCatalog();

prod.OpenProduct();

prod.EditProduct(old_name,ProNewName,ProMetaTitle,ProModel,ProGST,ProPrice,Pro
SpDisc,ProSort,ProStatus,ProTabs,ProCategories,ProImgPath);
prod.ProductValidation(ProNewName);
login.ClickLogout();
driver.get(userurl);
user.ProductsValidation(ProNewName);
}

@DataProvider(name="EditProduct")
public Object[][] Product() throws IOException
{
    return ExcelReader.DataTable("./TestData/TestData.xlsx", "Products");
}
}

```

## LoginPage.java

```

package com.ibm.pages;

import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.support.FindBy;
import org.openqa.selenium.support.PageFactory;
import org.openqa.selenium.support.ui.WebDriverWait;

public class LoginPage {
    @FindBy(name="email")
    WebElement emailEle;

    @FindBy(name="pword")
    WebElement passwordEle;

    @FindBy(xpath="/html/body/div/div/div/div[2]/form/button")
    WebElement loginEle;

    @FindBy(xpath="//a[@title='Logout']")
    WebElement logoutEle;

    WebDriverWait wait;
    WebDriver driver;

    public LoginPage(WebDriver driver, WebDriverWait wait)
    {
        PageFactory.initElements(driver, this);
    }
}

```

```

        public void EnterEmail(String username)
        {
            emailEle.sendKeys(username);
        }
        public void EnterPassword(String password)
        {
            passwordEle.sendKeys(password);
        }
        public void clickLogin()
        {
            loginEle.click();
        }
        public void ClickLogout()
        {
            logoutEle.click();
        }
    }
}

```

## **DashboardPage.java**

```

package com.ibm.pages;

import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.support.FindBy;
import org.openqa.selenium.support.PageFactory;
import org.openqa.selenium.support.ui.ExpectedConditions;
import org.openqa.selenium.support.ui.WebDriverWait;

public class DashboardPage {

    //Finding the WebElements
    @FindBy(partialLinkText="Catalog")
    WebElement catalogEle;

    @FindBy(xpath="/html/body/div[1]/div[1]/div/ul/li[5]/a")
    WebElement marketEle;

    @FindBy(xpath="//*[contains(text(),'System')]")
    WebElement systemEle;

    @FindBy(xpath="//*[contains(text(),'Shipping')]")
    WebElement shipEle;

    WebDriver driver;
    WebDriverWait wait;

    //catalog page constructor
    public DashboardPage(WebDriver driver, WebDriverWait wait)
    {
        this.driver=driver;
        this.wait=wait;
        PageFactory.initElements(driver, this);
    }
}

```

```

//Opencatalog method to click on catalog
public void OpenCatalog()
{
    wait.until(ExpectedConditions.visibilityOf(catalogEle));
    catalogEle.click();
}

//method for opening marketing
public void OpenMarketing()
{
    marketEle.click();
}
//method for opening system
public void OpenSystem()
{
    systemEle.click();
}
public void OpenShipping()
{
    shipEle.click();
}
}

```

## **ProductsPage.java**

```

package com.ibm.pages;

import org.openqa.selenium.By;
import org.openqa.selenium.JavascriptExecutor;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.support.FindBy;
import org.openqa.selenium.support.PageFactory;
import org.openqa.selenium.support.ui.ExpectedConditions;
import org.openqa.selenium.support.ui.Select;
import org.openqa.selenium.support.ui.WebDriverWait;
import org.testng.Assert;
import org.testng.Reporter;

public class ProductsPage {

    @FindBy(xpath="//*[contains(text(),'Products')]")
    WebElement prodEle;

    @FindBy(xpath="//*[contains(text(),'Product List')]")
    WebElement prodListEle;

    @FindBy(name="dataTableExample2_length")
    WebElement showEle;

    @FindBy(id="pro_name")
    WebElement proNameEle;
    @FindBy(id="meta_title")
    WebElement metaTitleEle;
    @FindBy(id="toTop")
    WebElement toTopEle;
}

```

```

//@FindBy(xpath="//a[contains(text()='Data')]")
@FindBy(xpath="/html/body/div[1]/div[3]/div/form/div[2]/div/div/div[2]/div/div
/ul/li[2]/a")
WebElement dataEle;
@FindBy(id="model")
WebElement modelEle;
@FindBy(id="gst")
WebElement gstEle;
@FindBy(id="price2")
WebElement priceEle;
@FindBy(id="special_dis")
WebElement spDisEle;
@FindBy(id="price_after_special_dis2")
WebElement SpPriceEle;
@FindBy(id="sort_order")
WebElement sortEle;
@FindBy(id="status")
WebElement statusEle;
//@FindBy(xpath="//a[contains(text()='Link')]")
@FindBy(xpath="/html/body/div[1]/div[3]/div/form/div[2]/div/div/div[2]/div/div
/ul/li[3]/a")
WebElement linkEle;
@FindBy(id="tabs")
WebElement tabsEle;
@FindBy(id="categories")
WebElement categoryEle;
//@FindBy(xpath="//a[contains(text()='Image')]")
@FindBy(xpath="/html/body/div[1]/div[3]/div/form/div[2]/div/div/div[2]/div/div
/ul/li[4]/a")
WebElement imgEle;
@FindBy(xpath="//input[@id='pro_image']")
WebElement proImgEle;

@FindBy(xpath="//button[@title='Save']")
WebElement saveEle;

WebDriver driver;
WebDriverWait wait;
JavascriptExecutor js;

public ProductsPage(WebDriver driver, WebDriverWait wait)
{
    this.driver=driver;
    this.wait=wait;
    PageFactory.initElements(driver, this);
    js=(JavascriptExecutor) driver;
}
public void OpenProduct()
{
    prodEle.click();
}

public void EditProduct(String old_name,String new_name,String
meta_Title,String model,String gst,String price,String SplDiscount,String sort,String
Status,String tabs,String categories,String imgPath) throws InterruptedException

```

```

{
    wait.until(ExpectedConditions.visibilityOf(prodListEle));
    /*Select showSel=new Select(showEle);
    showSel.selectByVisibleText("25");*/
    wait.until(ExpectedConditions.visibilityOf(prodListEle));
    int
rowLen=driver.findElements(By.xpath("//table[@id='dataTableExample2']/tbody/tr")).size();
    for(int i=1;i<=rowLen;i++)
    {
        String
prod_name=driver.findElement(By.xpath("//table[@id='dataTableExample2']/tbody/tr["+i+"]/


```

```

        js.executeScript("arguments[0].scrollIntoView(true);",modelEle);
        modelEle.sendKeys(model);
        gstEle.sendKeys(gst);
        priceEle.sendKeys(price);
        spDisEle.sendKeys(SplDiscount);

        js.executeScript("arguments[0].scrollIntoView(true);",statusEle);
        sortEle.sendKeys(sort);
        Select statusSel=new Select(statusEle);
        statusSel.selectByVisibleText(Status);
        driver.findElement(By.id("toTop")).click();
        //opening Link Tab
        js.executeScript("arguments[0].click()",linkEle);
        Thread.sleep(5000);//due to my browser prb, added this
        Select TabsSel=new Select(tabsEle);
        TabsSel.selectByVisibleText(tabs);
        Select CatSel=new Select(categoryEle);
        CatSel.selectByVisibleText("");
        CatSel.selectByVisibleText(categories);
        //opening Image tab
        js.executeScript("arguments[0].click()",imgEle);
        //wait.until(ExpectedConditions.visibilityOf(proImgEle));
        Thread.sleep(5000);//due to my browser prb, added this
        proImgEle.sendKeys(imgPath);
        //clicking on Save to save the product
        saveEle.click();
        break;
    }
}
}

public void ProductValidation(String pro_newname)
{
    wait.until(ExpectedConditions.presenceOfElementLocated(By.xpath("//div[contains(text(),'successfully updated product')]")));
    /*Select showSel=new Select(showEle);
    showSel.selectByVisibleText("25");*/
    String actualMsg=driver.getPageSource();
    Assert.assertTrue(actualMsg.contains(pro_newname), "Assertion on Edited Product in Admin Page!!!");
    Reporter.log("Assertion on Edited Product in Admin Page!!!");
}
}

```

## UserPage.java

```

package com.ibm.pages;

import org.openqa.selenium.By;
import org.openqa.selenium.JavascriptExecutor;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.support.FindBy;
import org.openqa.selenium.support.PageFactory;
import org.openqa.selenium.support.ui.ExpectedConditions;

```

```

import org.openqa.selenium.support.ui.WebDriverWait;
import org.testng.Assert;
import org.testng.Reporter;

public class UserPage
{
    //Finding the elements of the User Page
    @FindBy(xpath="(/span[@class='click-categories flaticon-bars'])[2]")
    WebElement catBarEle;
    @FindBy(xpath="(*[@class='category-drop-list-inner'])[2]")
    WebElement CateList;
    @FindBy(xpath="/html/body/div[1]/div[5]/div/div[1]/div/nav/ul/li[2]/span")
    WebElement HdrEle;
    @FindBy(xpath="//input[@id='search-box']")
    WebElement mainSrchEle;

    WebDriver driver;
    WebDriverWait wait;
    JavascriptExecutor js;
    //Constructor
    public UserPage(WebDriver driver,WebDriverWait wait)
    {
        this.driver=driver;
        this.wait=wait;
        js=(JavascriptExecutor) driver;
        PageFactory.initElements(driver, this);
    }
    //this method validates the edited category presence in User Page under "Shop
    By Category"
    public void CatergoryValidation(String catename)
    {
        catBarEle.click();
        wait.until(ExpectedConditions.visibilityOf(CateList));
        WebElement newcateEle=driver.findElement(By.partialLinkText(catename));
        newcateEle.click();
        String validCate=HdrEle.getText().trim();
        Assert.assertTrue(validCate.contains(catename), "Assertion on Edited
        Category in User Page!!!");
        Reporter.log("Assertion on Edited Category in User Page!!!");
    }
    //This method validates the edited Phone number present in the footer
    information in User Page
    public void footerPhnoValidation(String Phno) throws InterruptedException
    {
        WebElement
footPhno=driver.findElement(By.xpath("//p[contains(text(), '"+Phno+"')]"));
        js.executeScript("arguments[0].scrollIntoView(true);", footPhno);
        String footerPh=footPhno.getText().trim();
        Assert.assertTrue(footerPh.contains(Phno), "Assertion on Phone number in
        footer section!!!");
        Reporter.log("Assertion on Phone number in footer section!!!");
    }
    //This method validates the edited Phone number present in the header
    information in User Page
    public void headerPhnoValidation(String Phno) throws InterruptedException

```

```

{
    WebElement hdrPhno=driver.findElement(By.xpath("//div[@class='header-mid-right-content']")[1]));
    js.executeScript("arguments[0].scrollIntoView(true);", hdrPhno);
    String headerPh=hdrPhno.getText().trim();
    Assert.assertTrue(headerPh.contains(Phno), "Assertion on Phone number in header section!!");
    Reporter.log("Assertion on Phone number in header section!!");
}
//this method validates the edited Product in User page
public void ProductsValidation(String ProNewName) throws InterruptedException
{
    mainSrchEle.click();
    //sending keys in the search

    driver.findElement(By.xpath("//*[@id=\"searchModal\"]//div/div/div/div[1]/input")).sendKeys(ProNewName);
    //Thread.sleep(5000);
    //selecting the prooduct from the dropdown

    driver.findElement(By.xpath("//span[text()='"+ProNewName+"']/ancestor::div[@id ='searchproducts-div']")).click();
    String validProduct=HdrEle.getText().trim();
    Assert.assertTrue(validProduct.contains(ProNewName), "Assertion on Edited Product in User Page!!!");
    Reporter.log("Assertion on Edited Product in User Page!!!");
}
}

```

## **DriverLaunch.java**

```

package com.ibm.pages;

import java.io.IOException;
import java.util.HashMap;
import java.util.concurrent.TimeUnit;

import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.firefox.FirefoxDriver;
import org.openqa.selenium.ie.InternetExplorerDriver;
import org.openqa.selenium.support.ui.WebDriverWait;
import org.testng.annotations.AfterMethod;
import org.testng.annotations.BeforeMethod;
import org.testng.annotations.BeforeSuite;
import org.testng.annotations.Optional;
import org.testng.annotations.Parameters;

import com.ibm.utilities.PropertiesFileHandler;

public class DriverLaunch {

    public WebDriver driver;
    public WebDriverWait wait;
    public PropertiesFileHandler propFileHandle;
}

```

```

public HashMap<String, String> hashData;

@BeforeSuite
public void preSetForTest() throws IOException
{
    String file=".~/TestData/data.properties";
    propFileHandle =new PropertiesFileHandler();
    hashData=propFileHandle.getPropertiesAsMap(file);
}
@BeforeMethod
@Parameters({"browser"})
public void initialisation(@Optional("ch")String browser)
{
    browserInitialization(browser);
    driver.manage().window().maximize();
    driver.manage().timeouts().implicitlyWait(30, TimeUnit.SECONDS);
    wait=new WebDriverWait(driver, 60);
}
@AfterMethod
public void closingBrowser()
{
    driver.quit();
}

public void browserInitialization(String browser)
{
    switch(browser.toLowerCase())
    {
        case "ff":
            System.setProperty("webdriver.gecko.driver",
".~/drivers/geckodriver.exe");
            driver=new FirefoxDriver();
            break;
        case "ch":
            System.setProperty("webdriver.chrome.driver",
".~/drivers/chromedriver.exe");
            driver=new ChromeDriver();
            break;
        case "ie":
            System.setProperty("webdriver.ie.driver",
".~/drivers/IEDriverServer.exe");
            driver=new InternetExplorerDriver();
            break;
    }
}
}

```

### **PropertiesFileHandler.java**

```

package com.ibm.utilities;

import java.io.FileInputStream;
import java.io.IOException;
import java.util.HashMap;
import java.util.Properties;

```

```

import java.util.Set;

public class PropertiesFileHandler {

    public HashMap<String, String> getPropertiesAsMap(String file) throws
IOException {
        HashMap<String, String> magentoMap = new HashMap<String, String>();

        FileInputStream fileIn = new FileInputStream(file);
        Properties prop = new Properties();
        prop.load(fileIn);

        Set<Object> keysProp = prop.keySet();
        for (Object key : keysProp) {
            magentoMap.put(key.toString(), prop.getProperty(key.toString()));
        }
        prop.clear();
        return magentoMap;
    }
}

```

### **data.properties**

```

adminurl=https://atozgroceries.com/admin
userurl=https://atozgroceries.com
username=demo@atozgroceries.com
password=456789

```

### **testng.xml**

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE suite SYSTEM "http://testng.org/testng-1.0.dtd">
<suite name="Suite">
    <test thread-count="5" name="Test">
        <parameter name="browser" value="ff"></parameter>
        <classes>
            <class name="com.ibm.test.BaseTest">
                <methods><include name="Day8TestCase"></include></methods>
            </class>
        </classes>
    </test> <!-- Test -->
</suite> <!-- Suite -->

```

### **ExcelReader.java**

```

package com.ibm.utilities;

import java.io.FileInputStream;
import java.io.IOException;

import org.apache.poi.ss.usermodel.DataFormatter;
import org.apache.poi.xssf.usermodel.XSSFSheet;
import org.apache.poi.xssf.usermodel.XSSFWorkbook;

public class ExcelReader {
    public static Object[][] DataTable(String WBLoc, String sheetName) throws
IOException

```

```

}

XSSFWorkbook TestDataWB=new XSSFWorkbook(new FileInputStream(WBLoc));
XSSFSheet ProductsSheet=TestDataWB.getSheet(sheetName);
Object[][] ProductInfo=new Object[ProductsSheet.getPhysicalNumberOfRows()-1][ProductsSheet.getRow(0).getPhysicalNumberOfCells()];
int rowCount=ProductsSheet.getPhysicalNumberOfRows();
int cellCount=ProductsSheet.getRow(0).getPhysicalNumberOfCells();
for(int i=1;i<rowCount;i++)
{
    for(int j=0;j<cellCount;j++)
    {
        DataFormatter format=new DataFormatter();
        String cellValue=format.formatCellValue(ProductsSheet.getRow(i).getCell(j));
        ProductInfo[i-1][j]=cellValue;
    }
}

return ProductInfo;
}
}

```

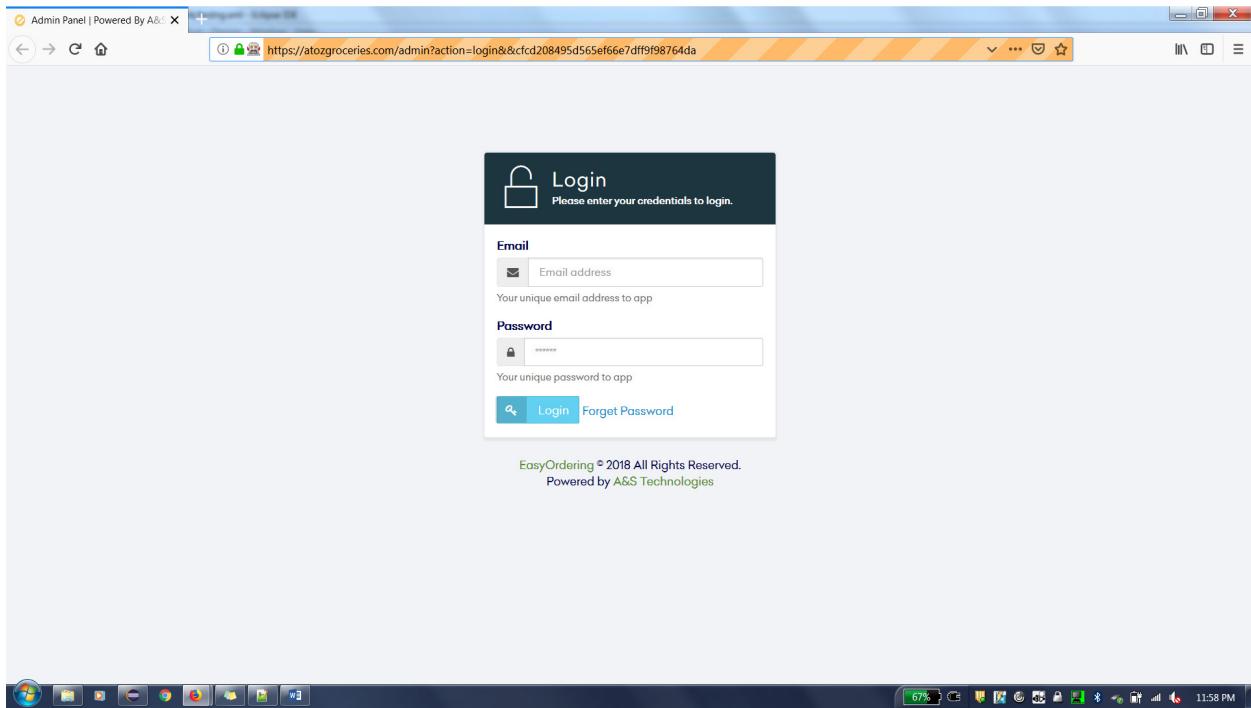
## TestData.xlsx

The screenshot shows the Eclipse IDE interface with the TestData.xlsx file open. The left pane displays the project structure, and the right pane shows the Excel spreadsheet with the following data:

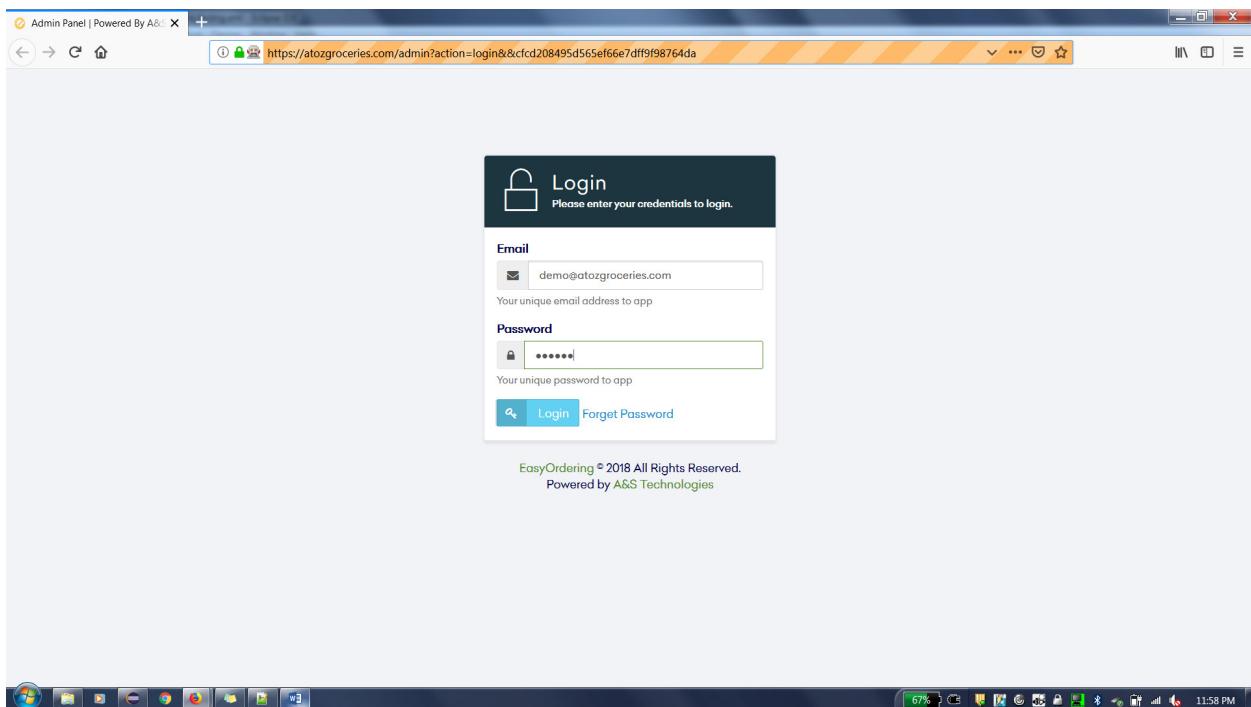
	A	B	C	D	E	F	G	H	I	J	K	
1	old_name	ProNewName	PrometaTitle	ProModel	ProGST	ProPrice	ProSpDisc	ProSort	ProStatus	ProTabs	ProCategories	ProImgPath
2	Chocolate	Cashew-Harisha	Cashew	casew	5	450	10	2	Enabled	Test02	Dry Fruits	C:\Harisha\Selenium_Training\EclipseWorkSpace\SeleniumFramework\TestData\almonds.jpg
3												
4												
5												
6												
7												
8												
9												
10												
11												
12												
13												
14												
15												
16												
17												
18												
19												
20												
21												
22												
23												
24												
25												
26												

## Step by step output:

### 1. Navigating to the Admin Page of “atozgroceries”



### 2. Providing login credentials



### 3. Dashboard after logging into the site

The screenshot shows the Admin Panel Dashboard for 'atozgroceries.com'. The left sidebar includes links for Dashboard, Catalog, Sales, Customers, Marketing, System, and Reports. The main area displays four key metrics: Total Orders (1), Total Sales (₹5,537.07), Total Customers (3), and Total Products (44). Below these are sections for 'Latest Orders' and 'Recent Activity'.

#	Order ID	Customer	Status	Total	Date Added	Action
1	#10	Rajendra Sahoo	Pending	₹21,750.00	2018-12-21	
2	#6	TestNew	Pending	₹3,950.00	2018-12-20	
3	#5		Pending	₹8,700.00	2018-12-20	
4	#3		Pending	₹3,570.00	2018-12-18	
5	#2		Pending	₹3,570.00	2018-12-18	

### 4. Opening Catalog

The screenshot shows the Admin Panel Catalog section for 'atozgroceries.com'. The left sidebar includes links for Catalog (Tabs, Banners, Categories, Products, Product Image, Information), Sales, Customers, Marketing, System, and Reports. The main area displays four key metrics: Total Orders (6), Total Sales (₹27,965.00), Total Customers (19), and Total Products (227). Below these are sections for 'Latest Orders' and 'Recent Activity'.

#	Order ID	Customer	Status	Total	Date Added	Action
1	#10	Rajendra Sahoo	Pending	₹21,750.00	2018-12-21	
2	#6	TestNew	Pending	₹3,950.00	2018-12-20	
3	#5		Pending	₹8,700.00	2018-12-20	
4	#3		Pending	₹3,570.00	2018-12-18	
5	#2		Pending	₹3,570.00	2018-12-18	

5. Opening Products and clicking on the action of desired product to edit

The screenshot shows the 'Products' section of the Admin Panel. On the left is a sidebar with categories like Catalog, Sales, Customers, Marketing, System, and Reports. The 'Catalog' section is expanded, showing sub-options like Tabs, Banners, Categories, Products, Product Image, and Information. The main area displays a 'Product List' table with columns: #, Product Name, Image, Model, Price, Total Quantity, Sort Order, Status, and Action. There are 10 entries shown. The second entry, 'Chocolate', has its 'Action' button highlighted.

#	Product Name	Image	Model	Price	Total Quantity	Sort Order	Status	Action
1	New Prod Req		1098	1,890.00 1		1	Enabled	Action
2	Chocolate		Chocolates	203.3 24		2	Disabled	Action
3	Basmathi Rice-harisha		IndiaGate Rice	174.96 194.4	1	1	Enabled	Action
4	NewTest(working on this)		NewTest	425.25 472.5	5	1	Enabled	Action
5	Bisly Water		Model-W	19.95 24	25	0	Enabled	Action
6	NewBadhamProduct		New	1,320.00 124		1	Enabled	Action
7	DoNotDelete		excellent	1,025.00 1026		1	Enabled	Action

This screenshot is identical to the one above, showing the 'Products' list. However, the 'Edit' button for the second product, 'Chocolate', is now highlighted, indicating it has been selected for modification.

#	Product Name	Image	Model	Price	Total Quantity	Sort Order	Status	Action
1	New Prod Req		1098	1,890.00 1		1	Enabled	Action
2	Chocolate		Chocolates	203.3 24		2	Disabled	Edit
3	Basmathi Rice-harisha		IndiaGate Rice	174.96 194.4	1	1	Enabled	Delete
4	NewTest(working on this)		NewTest	425.25 472.5	5	1	Enabled	Action
5	Bisly Water		Model-W	19.95 24	25	0	Enabled	Action
6	NewBadhamProduct		New	1,320.00 124		1	Enabled	Action
7	DoNotDelete		excellent	1,025.00 1026		1	Enabled	Action

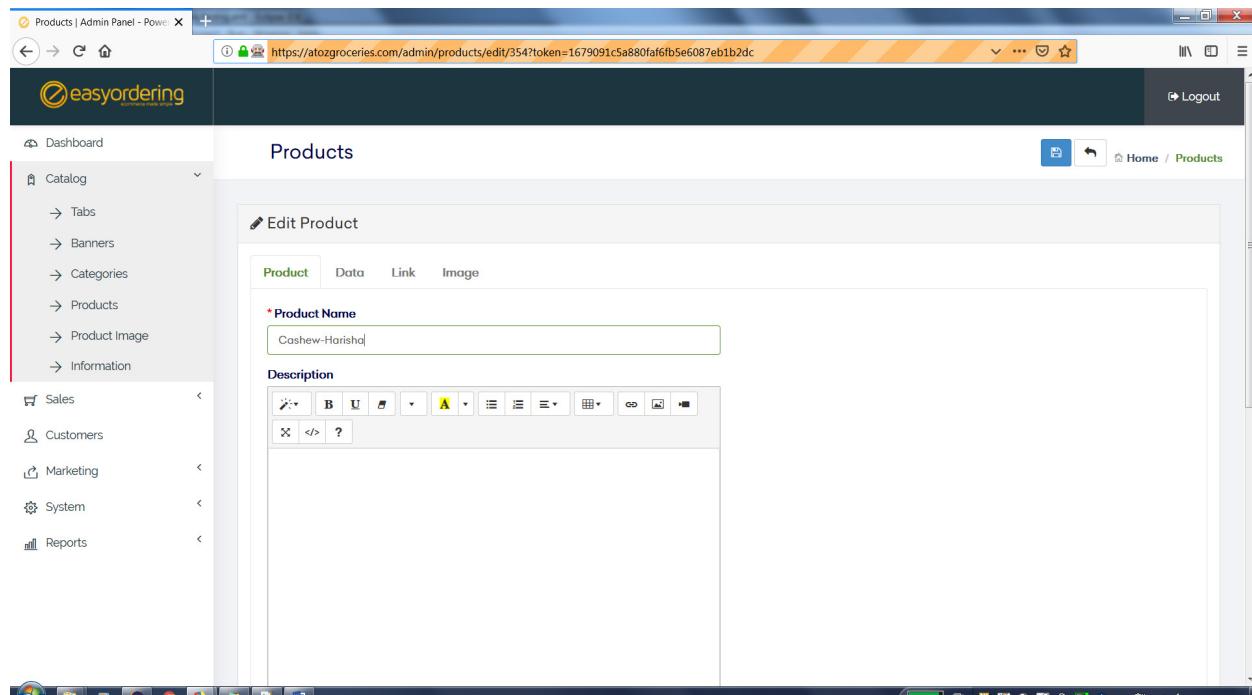
## 6. After clicking on Edit

The screenshot shows the 'Edit Product' form in the easyordering Admin Panel. The left sidebar has a 'Catalog' section with 'Products' selected. The main form has tabs for 'Product', 'Data', 'Link', and 'Image'. The 'Product' tab is active. It shows a 'Product Name' field containing 'Chocolate' and a rich text 'Description' editor which is currently empty.

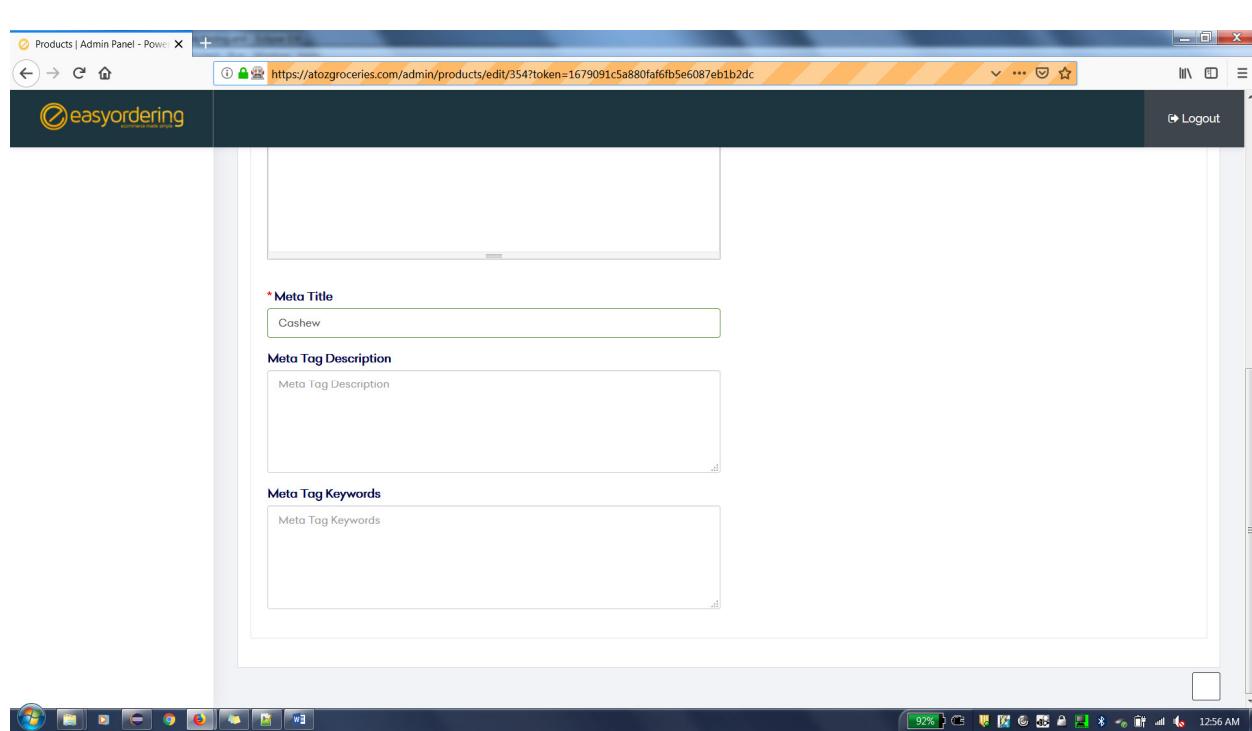
## 7. Clearing the existing values in Product tab

The screenshot shows the 'Edit Product' form after clearing the existing values. The 'Product' tab is still active. The 'Product Name' field now contains 'Product Name' and the 'Description' field is completely empty.

## 8. Sending the new values in product tab



The screenshot shows the 'Edit Product' screen in the easyordering Admin Panel. The left sidebar is collapsed, and the main area displays the 'Edit Product' form. The 'Product' tab is selected, showing the 'Product Name' field containing 'Casew-Harish'. Below it is a rich text editor for 'Description'. The status bar at the bottom indicates the system is at 92% battery and the time is 12:55 AM.

The screenshot shows the 'Edit Product' screen with the 'Meta Tag' section expanded. It includes fields for 'Meta Title' (containing 'Casew'), 'Meta Tag Description' (containing 'Meta Tag Description'), and 'Meta Tag Keywords' (containing 'Meta Tag Keywords'). The status bar at the bottom indicates the system is at 92% battery and the time is 12:56 AM.

## 9. Opening Data tab

The screenshot shows the 'Products' section of the easyordering Admin Panel. On the left, there's a sidebar with links like Dashboard, Catalog, Sales, Customers, Marketing, System, and Reports. The main area is titled 'Edit Product' and has tabs for Product, Data, Link, and Image. The 'Data' tab is selected. It contains fields for Model ('chocolates'), HSN ('HSN-4'), GST (%)' (8), Price (200), Special Discount (2), and Price After Special Discount (190). Below these are sections for 'Discount Quantity' and 'Discount Price'. The status bar at the bottom shows system information like battery level (92%) and time (12:56 AM).

## 10. Clearing the existing values in Data tab

This screenshot shows the same 'Edit Product' page after clearing the values. The fields now contain placeholder text: 'Price' (Price), 'Special Discount' (Special Discount), 'Price After Special Discount' (190), 'Discount Quantity' (0), 'Discount Price' (0), 'Quantity' (1), 'Weight' (test1), 'Total Quantity' (Total Quantity), 'Out Of Stock Status' (In Stock), 'Sort Order' (2), and 'Status' (Disabled). The status bar at the bottom remains the same.

## 11. Sending new values to data tab

The screenshot shows the 'Edit Product' page in the easyordering Admin Panel. The left sidebar is collapsed, and the main area is titled 'Products'. The 'Data' tab is selected. The form contains the following fields:

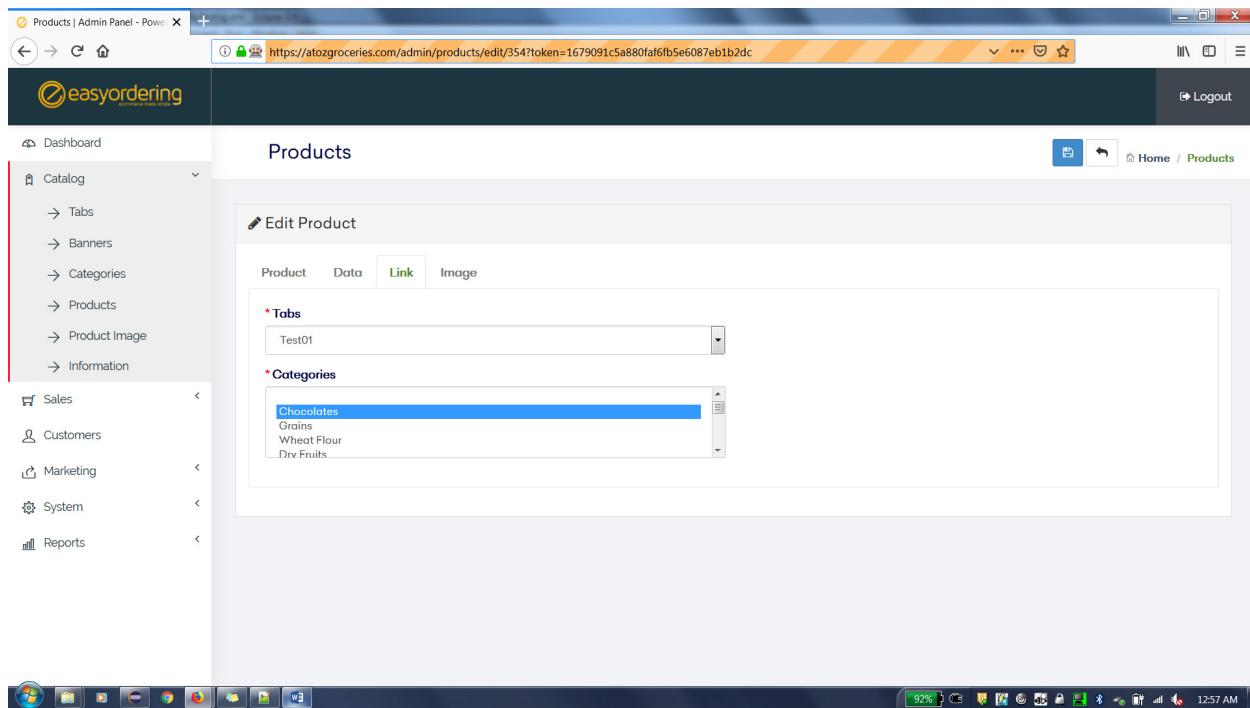
<b>* Model</b>	cashew
<b>HSN</b>	HSN-4
<b>* GST (%)</b>	5
<b>* Price</b>	450
<b>* Special Discount</b>	10
<b>* Price After Special Discount</b>	405

Below the form, there are two buttons: 'Discount Quantity' and 'Discount Price'.

The screenshot shows the same 'Edit Product' page after new values have been entered. The 'Data' tab is still selected. The form now displays the following data:

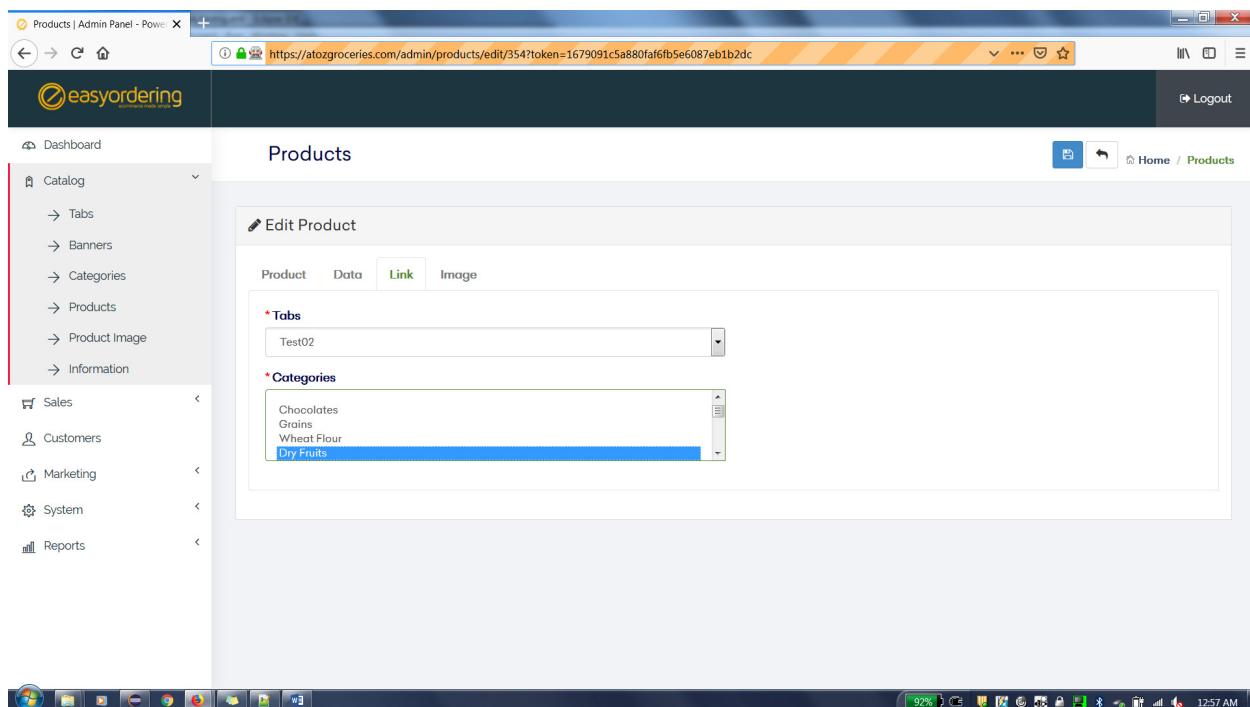
<b>* Model</b>	450
<b>* Special Discount</b>	10
<b>* Price After Special Discount</b>	405
<b>Discount Quantity</b>	0
<b>Discount Price</b>	0
<b>Quantity</b>	1
<b>Weight</b>	test1
<b>Total Quantity</b>	Total Quantity
<b>Out Of Stock Status</b>	In Stock
<b>Sort Order</b>	2
<b>Status</b>	Enabled

## 12. Opening Link Tab



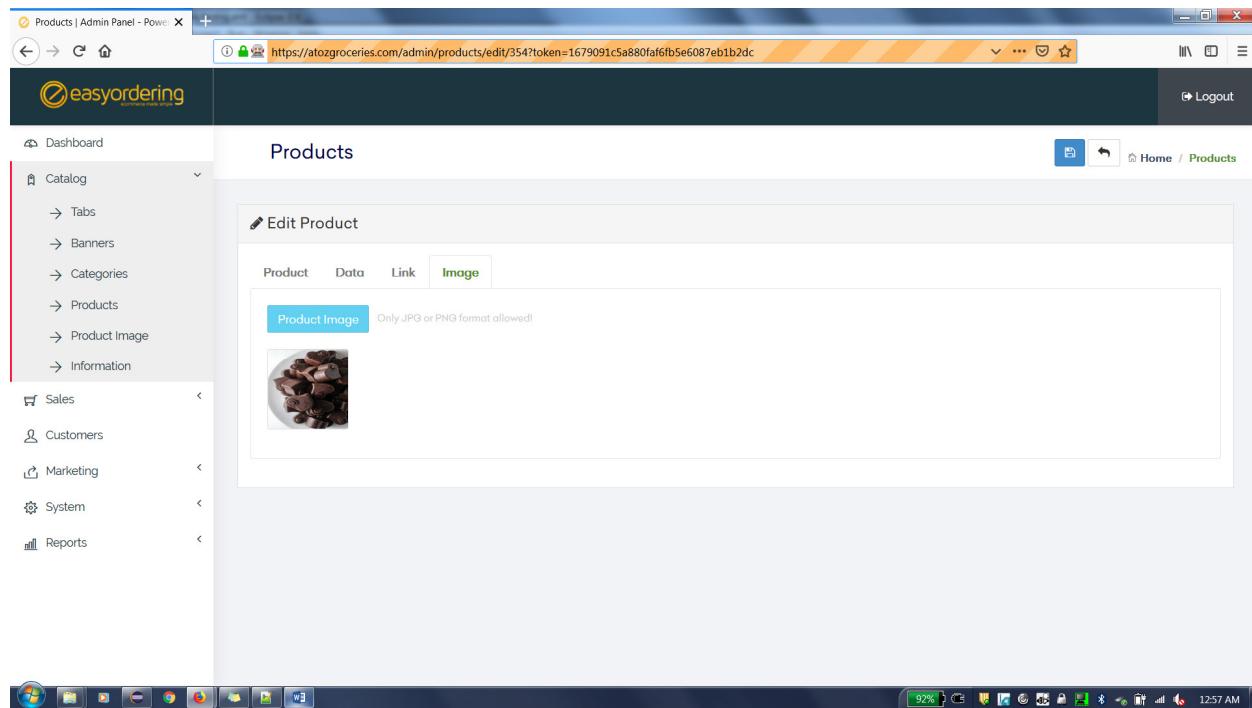
The screenshot shows the 'Edit Product' interface on the 'Link' tab. The left sidebar has 'Catalog' expanded, with 'Tabs' selected. The main area displays two dropdown menus: 'Tabs' containing 'Test01' and 'Categories' containing 'Chocolates', 'Grains', 'Wheat Flour', and 'Dry Fruits'. The status bar at the bottom shows system icons and the time '12:57 AM'.

## 13. Selecting new values in Link Tab

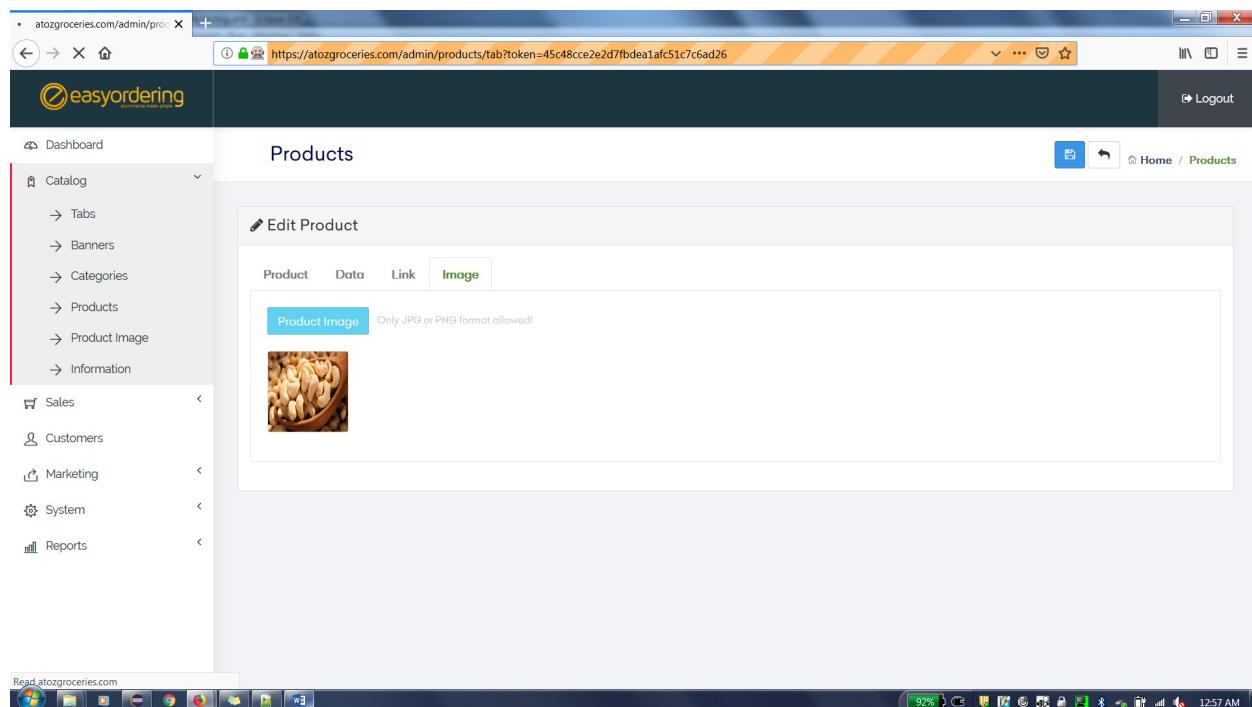


The screenshot shows the 'Edit Product' interface on the 'Link' tab. The left sidebar has 'Catalog' expanded, with 'Tabs' selected. The main area displays two dropdown menus: 'Tabs' containing 'Test02' and 'Categories' containing 'Chocolates', 'Grains', 'Wheat Flour', and 'Dry Fruits'. The 'Dry Fruits' option is highlighted with a blue selection bar. The status bar at the bottom shows system icons and the time '12:57 AM'.

## 14. Opening Image Tab



## 15. Selecting new image and Clicking on Save



## 16. Validation of edited product in Admin Page

The screenshot shows a Windows desktop environment with a browser window open to the 'Products' section of the Admin Panel at https://atozgroceries.com/admin/products/tab?token=45c48cce2e2d7fbdea1afc51c7c6ad26. A green success message box displays 'Success: You have successfully updated product!'. Below it is a 'Product List' table with the following data:

#	Product Name	Image	Model	Price	Total Quantity	Sort Order	Status	Action
1	New Prod Req		1098	1,890.00 1		1	Enabled	Action
2	Cashew-Harisha		cashew	425.25 472.5		2	Enabled	Action
3	Basmathi Rice-harisha		IndiaGate Rice	174.96 194.4	1	1	Enabled	Action
4	NewTest(working on this)		NewTest	425.25 472.5	5	1	Enabled	Action
5	Bisly Water		Model-W	19.95 21	25	0	Enabled	Action
6	NewBadhamProduct		New	1,320.00 123		1	Enabled	Action

## 17. Logging out from Admin Page

The screenshot shows a Windows desktop environment with a browser window open to the Admin Panel at https://atozgroceries.com/admin?action=login&&1679091c5a880faf6fb5e6087eb1b2dc. A green success message box displays 'Success: You have successfully logged out!'. Below it is a 'Login' form with the following fields:

**Login**  
Please enter your credentials to login.

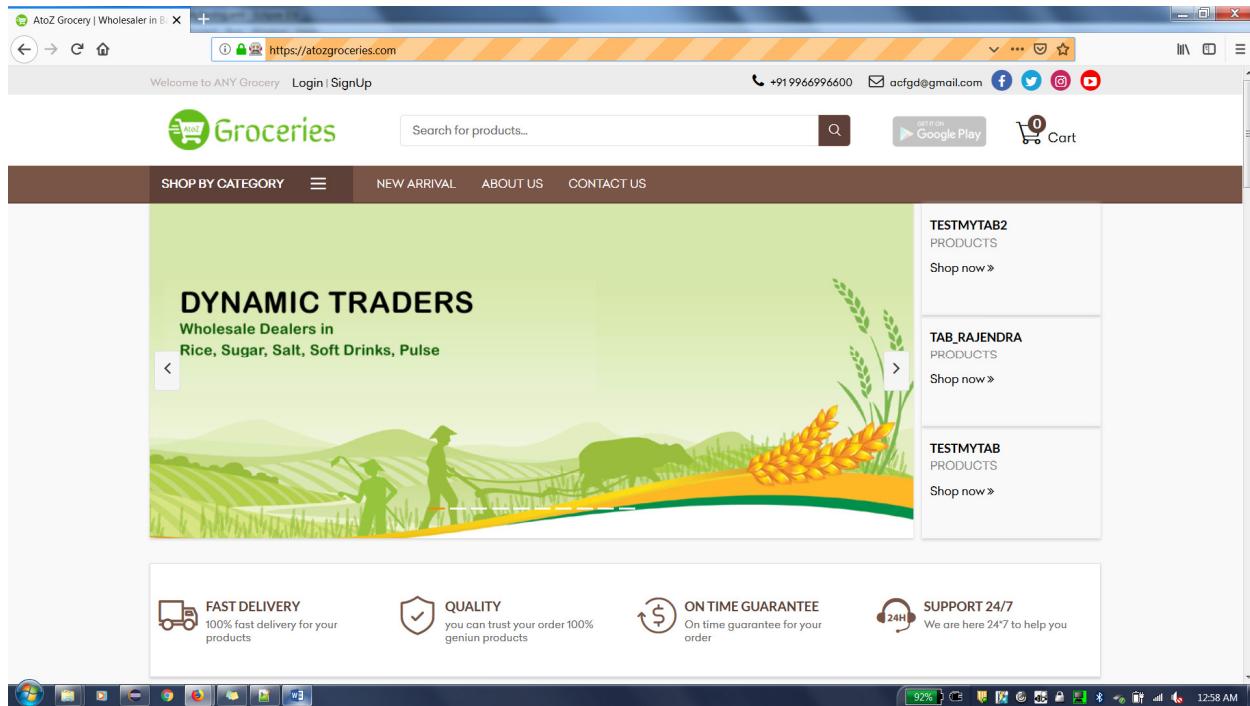
**Email**  
 Email address  
Your unique email address to app

**Password**  
 Password  
Your unique password to app

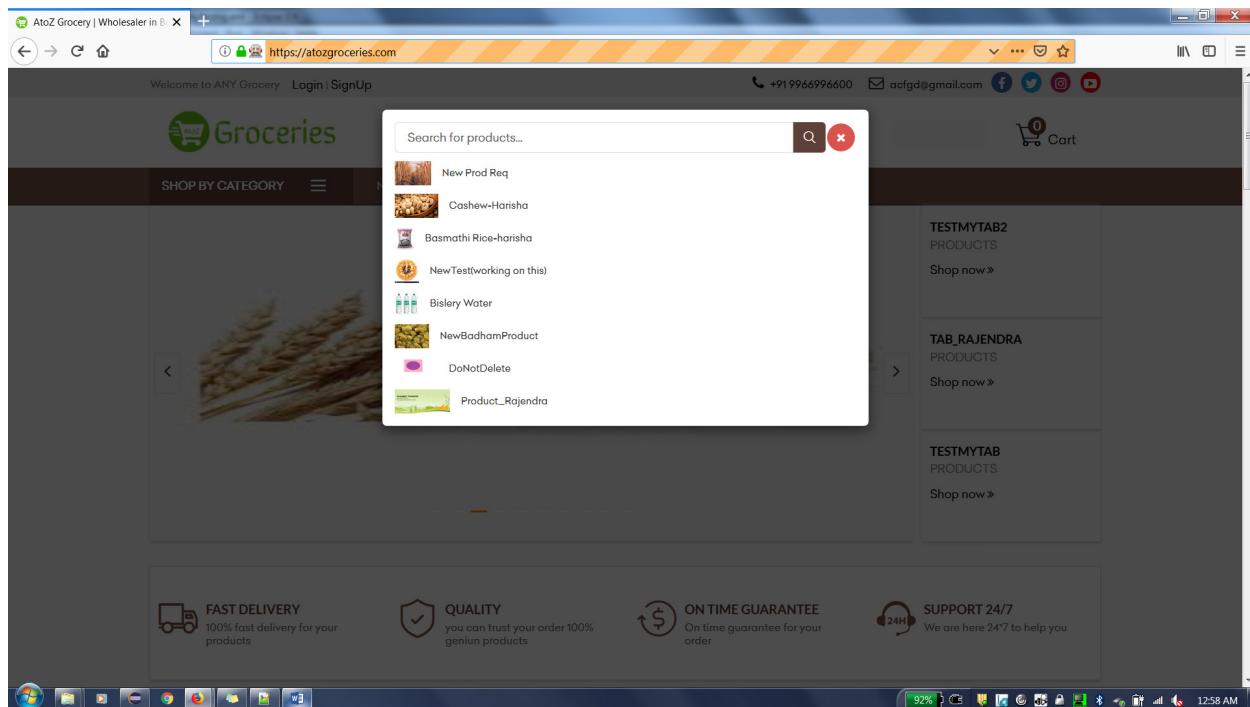
Login [Forgot Password](#)

EasyOrdering © 2018 All Rights Reserved.  
Powered by A&S Technologies

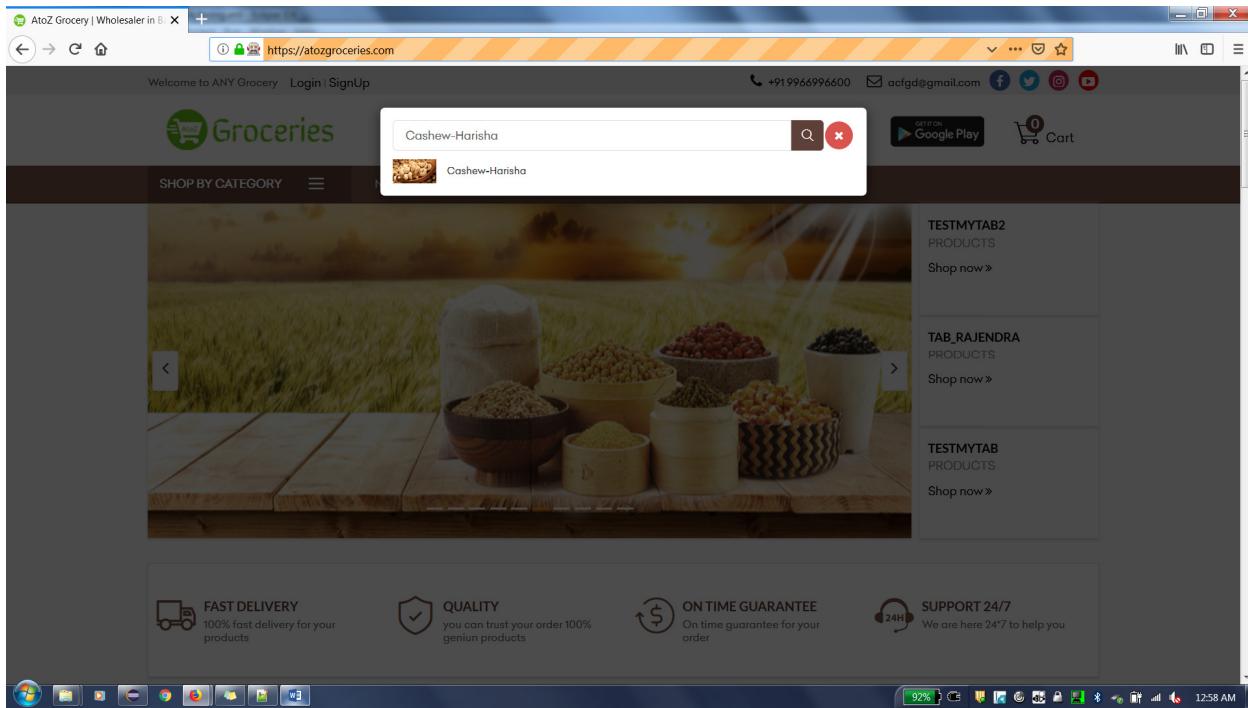
## 18. Navigating to User Page and Clicking on the Search for products



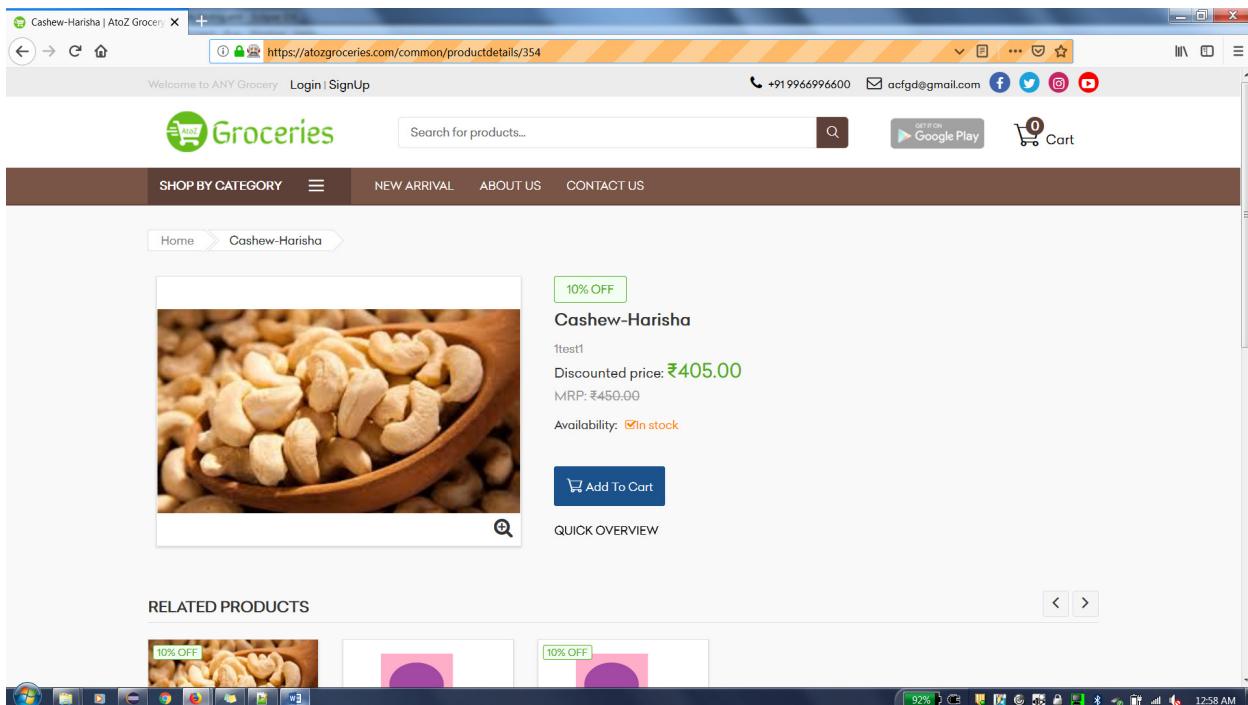
## 19. Clicking on the Search for products will open another search for products



## 20. Sending the value of edited product for validation and clicking on the desired product



## 21. Validation of Edited Product in User Page



## 22. Output from console

The screenshot shows the Eclipse IDE interface with the following details:

- File Structure:** The left pane shows the project structure under "SeleniumFramework". It includes packages like AgeException, Collection, DataDrivenFramework, DataDrivenFramework\_PF, Even, HighestNumber, HybridFramework, HybridProject, MagentoApplication, MagentoForgotPwdInTestNG, Operators, PatternX, PrintingX, SampleProg, selenium-webdriver-config, and SeleniumFramework. Under SeleniumFramework, there's an IRE System Library [jdk-11.0.1] and a src folder containing com.ibm, com.ibm.pages, com.ibm.test, com.ibm.utilities, ExcelReaderJava, ExcelUtilJava, JDBC\_Connection.java, and PropertiesfileHandler.java.
- TestNG XML:** The main editor shows the TestNG XML configuration:
 

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE suite SYSTEM "http://testng.org/testng-1.0.dtd">
<suite name="Suite">
<test thread-count="5" name="test">
<parameter name="browser" value="ff"></parameter>
<classes>
<class name="com.ibm.test.BaseTest">
<methods><include name="Day8TestCase"></include></methods>
</class>
</classes>
</test>
</suite> <!-- Test -->
<!-- Suite -->
```
- Console Output:** The bottom pane displays the command-line output of the test execution, showing various Java errors related to WebDriver and TestNG, and the final test results:
 

```
JavaScript error: https://atogroceries.com/assets/admin/dist/js/app.min.js, line 1: TypeError: i is null
JavaScript error: https://atogroceries.com/admin/products/tb?token=45c40cc2e2d7fbde1ef51c76ad26, line 7751: TypeError: document.getElementById(...) is null
1545993407971 Marionette INFO Stopped listening on port 5673
[Parent 5168, Gecko_IOThread] WARNING: pipe error: 109: file z:/build/build/src/ipc/chromium/src/chrome/common/ipc_channel_win.cc, line 346
[Child 14152, Chrome_ChildThread] WARNING: pipe error: 109: file z:/build/build/src/ipc/chromium/src/chrome/common/ipc_channel_win.cc, line 346
[Parent 5168, Gecko_IOThread] WARNING: pipe error: 109: file z:/build/build/src/ipc/chromium/src/chrome/common/ipc_channel_win.cc, line 346
[Child 12672, Chrome_ChildThread] WARNING: pipe error: 109: file z:/build/build/src/ipc/chromium/src/chrome/common/ipc_channel_win.cc, line 346
[Child 12672, Chrome_ChildThread] WARNING: pipe error: 109: file z:/build/build/src/ipc/chromium/src/chrome/common/ipc_channel_win.cc, line 346
[Child 12672, Chrome_ChildThread] WARNING: pipe error: 109: file z:/build/build/src/ipc/chromium/src/chrome/common/ipc_channel_win.cc, line 346
#####
[Child][RunMessage] Error: Channel closing: too late to send/recv, messages will be lost
PU 4112, Chrome_ChildThread] WARNING: pipe error: 109: file z:/build/build/src/ipc/chromium/src/chrome/common/ipc_channel_win.cc, line 346
#####
[Child][MessageChannel::SendAndWait] Error: Channel error: cannot send/recv
=====
Suite
Total tests run: 1, Failures: 0, Skips: 0
=====
```

## 23. Automated Index.html report

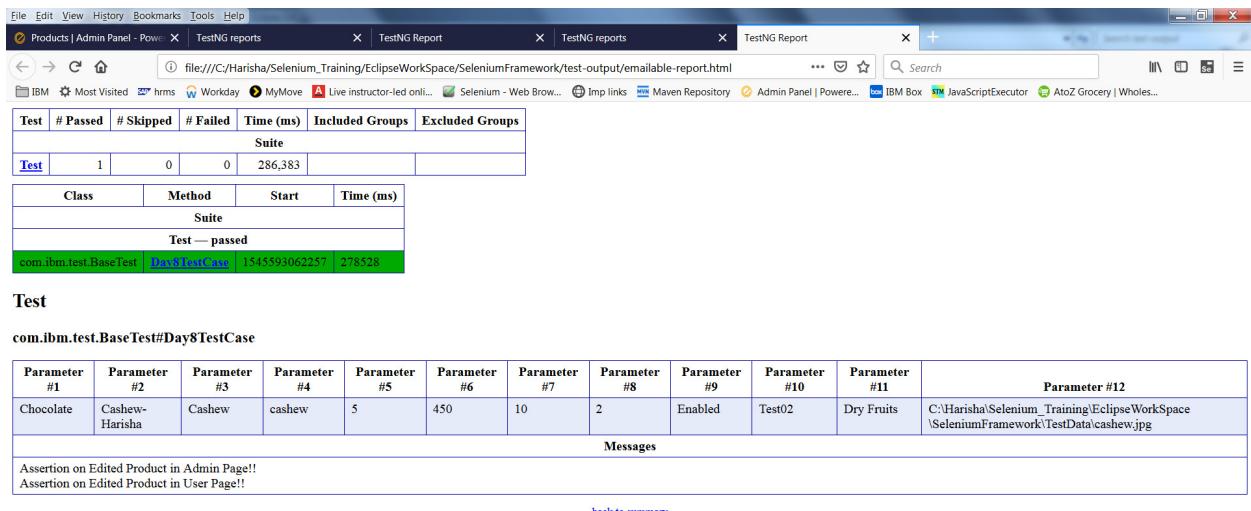
The screenshot shows a web browser window displaying the TestNG index.html report. The URL is file:///C/Harisha/Selenium\_Training/EclipseWorkSpace/SeleniumFramework/test-output/index.html#.

The report structure is as follows:

- All suites:**
  - Suite:**
    - Info:** C:\Harisha\Selenium\_Training\EclipseWorkSpace\SeleniumFramework\testing.xml
    - Result:**
      - method: 1 passed
      - Passed methods (hide): Day8TestCase(Chocolate, Cashew-Harisha, Cashew, cashew, 5, 450, 10, 2, Enabled, Test02, Dry Fruits, C:\Harisha\Se...)
- Test results:** 1 suite
  - Reporter output for Suite:**

```
Day8TestCase(Chocolate, Cashew-Harisha, Cashew, cashew, 5, 450, 10, 2, Enabled, Test02, Dry Fruits, C:\Harisha\Se...) Assertion on Edited Product in Admin Page!! Assertion on Edited Product in User Page!!
```

## 24. Automated Emailable report



The screenshot shows a web browser window with multiple tabs open. The active tab displays an 'Emailable report' generated by TestNG. The report includes the following sections:

- Summary Table:** A table showing the overall test results:

Test	# Passed	# Skipped	# Failed	Time (ms)	Included Groups	Excluded Groups
Suite						
Test	1	0	0	286,383		
- Detailed Test Table:** A table showing individual test cases:

Class	Method	Start	Time (ms)
Suite			
Test — passed			
com.ibm.test.BaseTest	Day8TestCase	1545593062257	278523
- Parameter Table:** A table showing parameters for a specific test case:

Parameter #1	Parameter #2	Parameter #3	Parameter #4	Parameter #5	Parameter #6	Parameter #7	Parameter #8	Parameter #9	Parameter #10	Parameter #11	Parameter #12
Chocolate	Cashew-Harisha	Cashew	cashew	5	450	10	2	Enabled	Test02	Dry Fruits	C:\Harisha\Selenium_Training\EclipseWorkSpace\SeleniumFramework\ TestData\cashew.jpg
- Messages:** A section containing two assertions:

Assertion on Edited Product in Admin Page!!  
Assertion on Edited Product in User Page!!

At the bottom of the browser window, the status bar shows a battery level of 92%, a signal strength icon, the time 12:59 AM, and a taskbar with various icons.

**\*\*\* End of the document \*\*\***