

Hi,

PFA day 9 test case, which includes DB validations

Process	Steps To Execute	Expected Results
Add a new Product	1. Login to admin portal 2. Add a new product by filling only the madatory fields. Note: You have to fill the details under PRODUCT, DATA, LINK, IMAGE	1. Validate the presence of the added product. 2. Verify the presence of added product on the database.

```
@Test
public void testcase9() throws InterruptedException, IOException,
SQLException {
    String url = data.get("url");
    String username = data.get("user");
    String password = data.get("password");
    String expectedtooltipmsg = data.get("expectedtooltipmsg");
    String expectednewtext= data.get("expectednewtext");
    String urlnew = data.get("urlnew");
    String expectedsearchtext= data.get("expectedsearchtext");

    // open the url https://atozgroceries.com/admin
    driver.get(url);

    // calling login page method to enter the user name, password and click
on log
        // in button
        LoginPage login = new LoginPage(driver, wait);
        login.enterEmailAddress(username);
        login.enterPassword(password);
        login.clickOnLogin();

    // calling admin page method to click on catalog

        AdminPage admin = new AdminPage(driver, wait);

        admin.clickOnCatalog();

    // calling Product page to perform the action

        ProductPage product = new ProductPage(driver, wait);

        product.addproduct();
        //product.selectTab();
        //product.categories(index);
        product.uploadImage();
}
```

```

        // calling save method from admin page to save the product list added

        admin.clickOnSave();

wait.until(ExpectedConditions.visibilityOfElementLocated(By.id("dataTableExample2")))
;

        String actualtext = product.getTextofProductname();

        System.out.println(actualtext);
        //Assert.assertEquals(expectednewtext, actualtext );

// its redirecting to the user page
        driver.navigate().to(urlnew);

// click on search box
        driver.findElement(By.xpath("//input[@id='search-
box']")).click();

// sending the text in search box

        driver.findElement(By.xpath("//*[@id=\"searchModal\"]/div/div/div/div[1]/input
")).sendKeys("mithai");
        Thread.sleep(5000);

// click on product and click on search

        driver.findElement(By.xpath("//*[@id=\"searchproducts-
div\"]/a/span")).click();
        Thread.sleep(5000);

        String actualsearchtext=
driver.findElement(By.xpath("//div[text()='Quick OverView']")).getText();
        System.out.println(actualsearchtext);

        Assert.assertEquals(actualsearchtext, expectedsearchtext);

        String Productname= product.dbVerification();
        System.out.println(Productname);

        Assert.assertEquals(Productname, actualtext);

        TakesScreenshot ts = (TakesScreenshot) driver;
        File file = ts.getScreenshotAs(OutputType.FILE);
        Date date = new Date();
        String currentDate = date.toString().replace(":", "-
");
        FileUtils.copyFile(file, new
File("./screenshots/Error_" + currentDate + ".png"));

```

```
    }  
}
```

Product Page :

```
package com.ibm.pages;
```

```
import java.sql.Connection;
```

```
import java.sql.DriverManager;
```

```
import java.sql.ResultSet;
```

```
import java.sql.SQLException;
```

```
import java.sql.Statement;
```

```
import java.util.List;
```

```
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.annotations.Parameters;
```

```
import org.testng.annotations.Test;
```

```
import junit.framework.Assert;
```

```

public class ProductPage {

    private static final String Productname = null;

    WebDriver driver;

    WebDriverWait wait;

    JavascriptExecutor js;

    @FindBy(xpath = "//*[@id=\"side-menu\"]/li[2]/ul/li[4]/a")
    WebElement productEle;

    @FindBy(xpath = "//a[@title='Add New']")
    WebElement addProductEle;

    @FindBy(xpath = "//input[@name='pro_name']")
    WebElement proNameEle;

    @FindBy(xpath = "//input[@name='meta_title']")
    WebElement metaEle;

    //@FindBy(linkText="Data")
    @FindBy(xpath="//a[text()='Data']")
    WebElement dataTabEle;

    @FindBy(xpath = "//input[@name='model']")
    WebElement modelEle;

    @FindBy(xpath = "//input[@name='gst']")
    WebElement gstEle;

    @FindBy(xpath = "//input[@name='price']")
    WebElement priceEle;

    @FindBy(xpath = "//input[@name='special_dis']")
    WebElement specDisEle;

    //@FindBy(xpath = "//input[@name='price_after_special_dis']")
    //WebElement afterDisEle;

```

```
@FindBy(xpath = "//a[text()='Link']")
```

```
WebElement linkTabEle;
```

```
@FindBy(xpath = "//select[@name='tabs']/option[3]")
```

```
WebElement TabsEle;
```

```
//@FindBy(xpath = "//select[@name='categories[]']/option[6]")
```

```
@FindBy(xpath="//select[@name='categories[]']/option[14]")
```

```
WebElement catagoriesEle;
```

```
@FindBy(xpath = "//a[text()='Image']")
```

```
WebElement imageTabEle;
```

```
@FindBy(xpath = "//input[@id='pro_image']")
```

```
WebElement productImageEle;
```

```
@FindBy(xpath="//table[@id='dataTableExample2']/tbody/tr/td[2]")
```

```
WebElement productnameEle;
```

```
@FindBy(xpath="//button[@class='btn dropdown-toggle btn-primary']")
```

```
WebElement actionEle;
```

```
@FindBy(xpath="//*[@id=\"dataTableExample2\"]/tbody/tr[6]/td[9]/div/ul/li[1]/a")
```

```
WebElement editEle;
```

```
@FindBy(xpath="//input[@id='quantity']")
```

```
WebElement quantityEle;
```

```
@FindBy(xpath="//input[@id='total_quantity']")
```

```
WebElement totalQuantityEle;
```

```
@FindBy(xpath="//*[contains(text(),'Success: You have successfully updated product!')]")
```

```
WebElement updatemsgEle;
```

```
@FindBy(xpath="//*[contains(text(),'Add To Cart')]")
```

WebElement addtoCartEle;

```
public ProductPage(WebDriver driver, WebDriverWait wait) {  
    PageFactory.initElements(driver, this);  
    this.driver= driver;  
    this.wait= wait;  
    js= (JavascriptExecutor)driver;  
  
}
```

```
public void addproduct() throws InterruptedException {  
  
    // click on product tab  
    productEle.click();  
  
    wait.until(ExpectedConditions.presenceOfElementLocated(By.xpath("//a[@title='Add  
New']")));  
    Thread.sleep(5000);  
  
    // click on add product  
    addProductEle.click();  
  
    //enter the name in product field  
    //proNameEle.sendKeys("NewTest(working on this)");  
    proNameEle.sendKeys("mithaimate");  
  
    //Enter the text in meta title field
```

```
//metaEle.sendKeys("NewTest");
```

```
metaEle.sendKeys("Milk");
```

```
Thread.sleep(5000);
```

```
// Click on data tab link
```

```
//dataTabEle.click();
```

```
js.executeScript("arguments[0].click();", dataTabEle);
```

```
//click on model
```

```
//modelEle.sendKeys("NewTest");
```

```
modelEle.sendKeys("Test");
```

```
// enter data for GST
```

```
gstEle.clear();
```

```
gstEle.sendKeys("5");
```

```
// enter data for price
```

```
//priceEle.sendKeys("450");
```

```
priceEle.sendKeys("80");
```

```
// enetring data for discount
```

```
//specDisEle.sendKeys("10");
```

```
specDisEle.sendKeys("0");
```

```
// click on link tab
```

```
//linkTabEle.click();
```

```

        js.executeScript("arguments[0].click();", linkTabEle);

        Thread.sleep(5000);

        // select Tabs

        TabsEle.click();

        //click on categories

        catagoriesEle.click();

    }

    // method to upload image

    public void uploadImage() throws InterruptedException {

        // click on ImageTab
        //Thread.sleep(10000);
        js.executeScript("arguments[0].click();", imageTabEle);
        //imageTabEle.click();

        // upload image

        productImageEle.sendKeys("C:\\Users\\IBM_ADMIN\\Downloads\\SeleniumFramework\\SeleniumFramework\\TestData\\amulmithaimate.jpg");
    }

```



```
}
```

```
//method to get the text of addedproductname
```

```
public String gettextofProductname() {
```

```
String actualtext= productnameEle.getText();
```

```
return actualtext;
```

```
}
```

```
// DB method to verify the product name that is added is displaying in the  
DB also
```

```
public String dbVerification() throws SQLException {
```

```
// connection to the DB
```

```
Connection c=
```

```
DriverManager.getConnection("jdbc:mysql://foodsonfinger.com:3306/foodsonfinger_atozgroceries",
```

```
"foodsonfinger_atoz","welcome@123");
```

```
Statement s= c.createStatement();
```

```
ResultSet rs= s.executeQuery("SELECT name FROM  
`as_products` WHERE name='mithaimate' ");
```

```
String Productname=null;
```

```
while(rs.next())
```

```
{
```

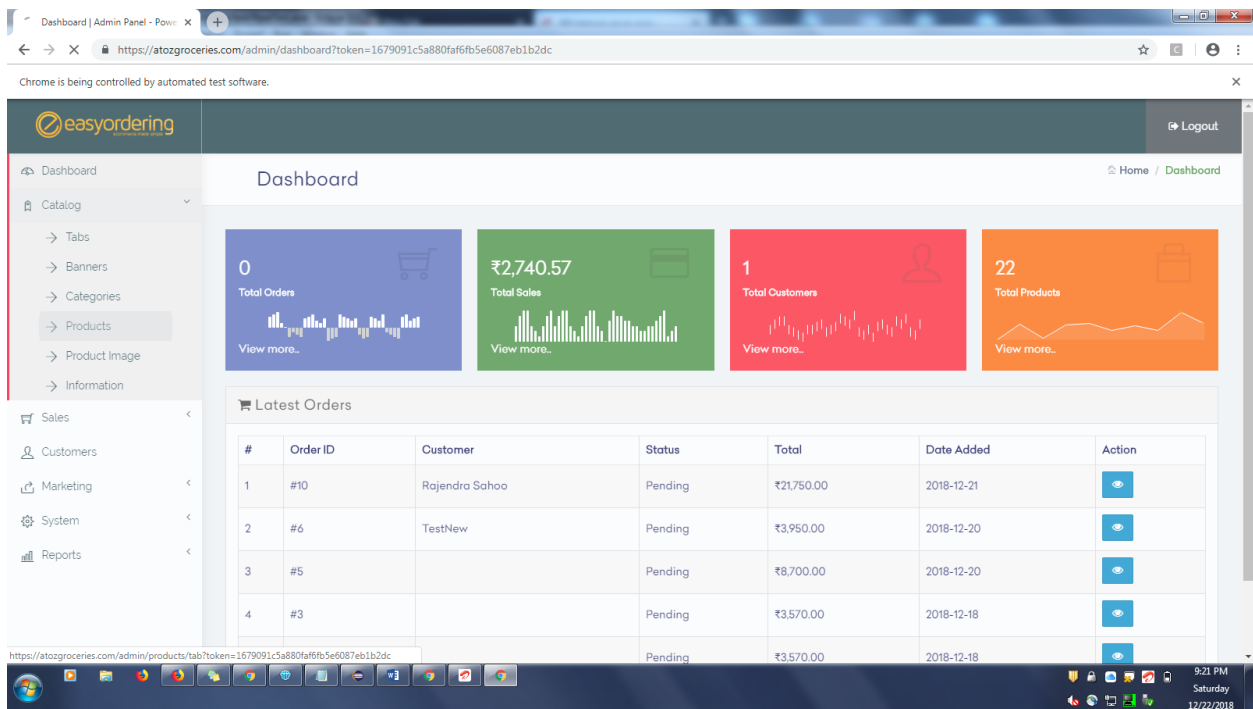
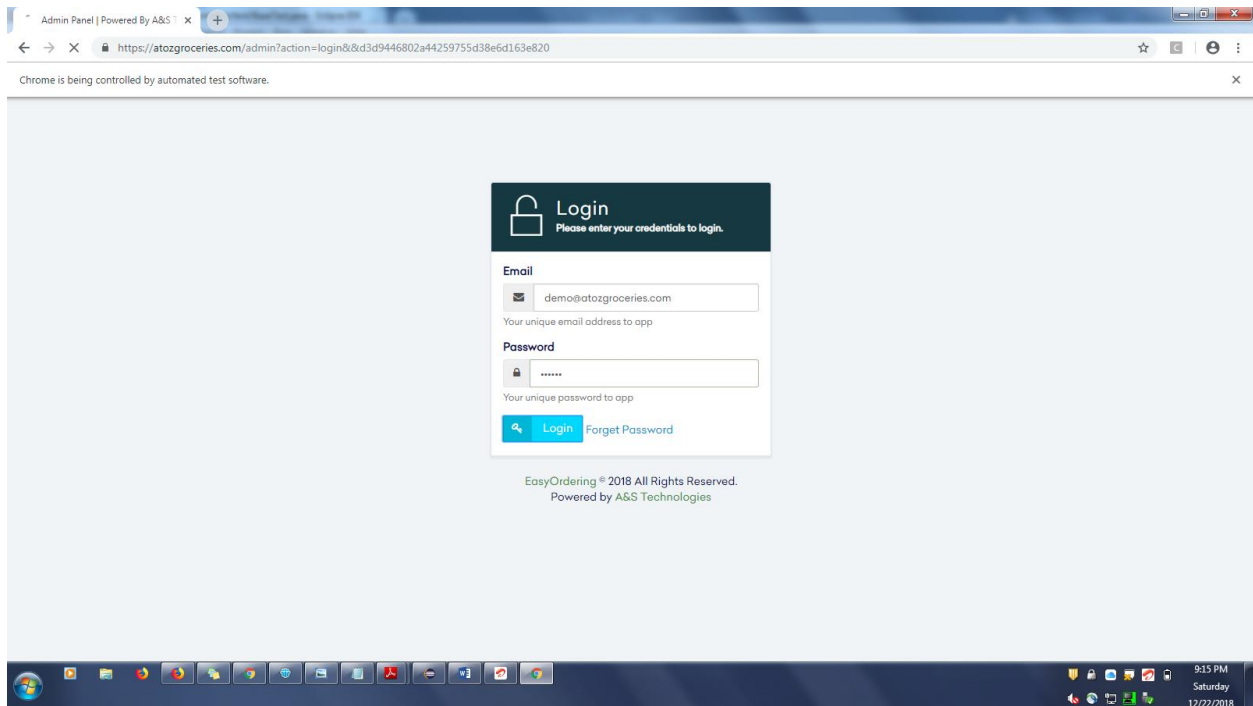
```
Productname=rs.getString("name");
```

```
System.out.println(Productname);
```

```
}
```

```
return Productname;
```


```
}
```



Products | Admin Panel - Power: X

← → ↻ https://atozgroceries.com/admin/products/tab?token=d3d9446802a44259755d38e6d163e820

Chrome is being controlled by automated test software.

 Logout

Dashboard

Catalog

- Tabs
- Banners
- Categories
- Products
- Product Image
- Information

Sales

Customers

Marketing

System

Reports

Products








+ / Home / Products

Product List

Show 1n entries

Copy CSV Excel PDF Print

Search:


#	Product Name	Image	Model	Price	Total Quantity	Sort Order	Status	Action
1	Fresh Bodham		Fresh	1,050.00 61		1	Enabled	Action
2	prod_chocolate		Dairy Milk	88.00 88		1	Enabled	Action
3	New Prod Req		1098	1,890.00 4		1	Disabled	Action
4	Cashew-Harisha		cashew	425.25 422.5		2	Enabled	Action
5	Basmathi Rice-harisha		IndiaGate Rice	174.96 194.4	1	1	Enabled	Action
6	NewTest(working on this)		NewTest	425.25 422.5	5	1	Enabled	Action
7	Bisleri Water		Model-W	19.95 21	25	0	Enabled	Action

10:13 PM
Monday
12/24/2018

Products | Admin Panel - Power: X

← → ↻ https://atozgroceries.com/admin/products/new?token=cfd208495d565ef66e7dff9f98764da

Chrome is being controlled by automated test software.

 Logout

Dashboard

Catalog

- Tabs
- Banners
- Categories
- Products
- Product Image
- Information

Sales

Customers

Marketing

System

Reports

Products

+ / Home / Products


Add Product

Product Data Link Image

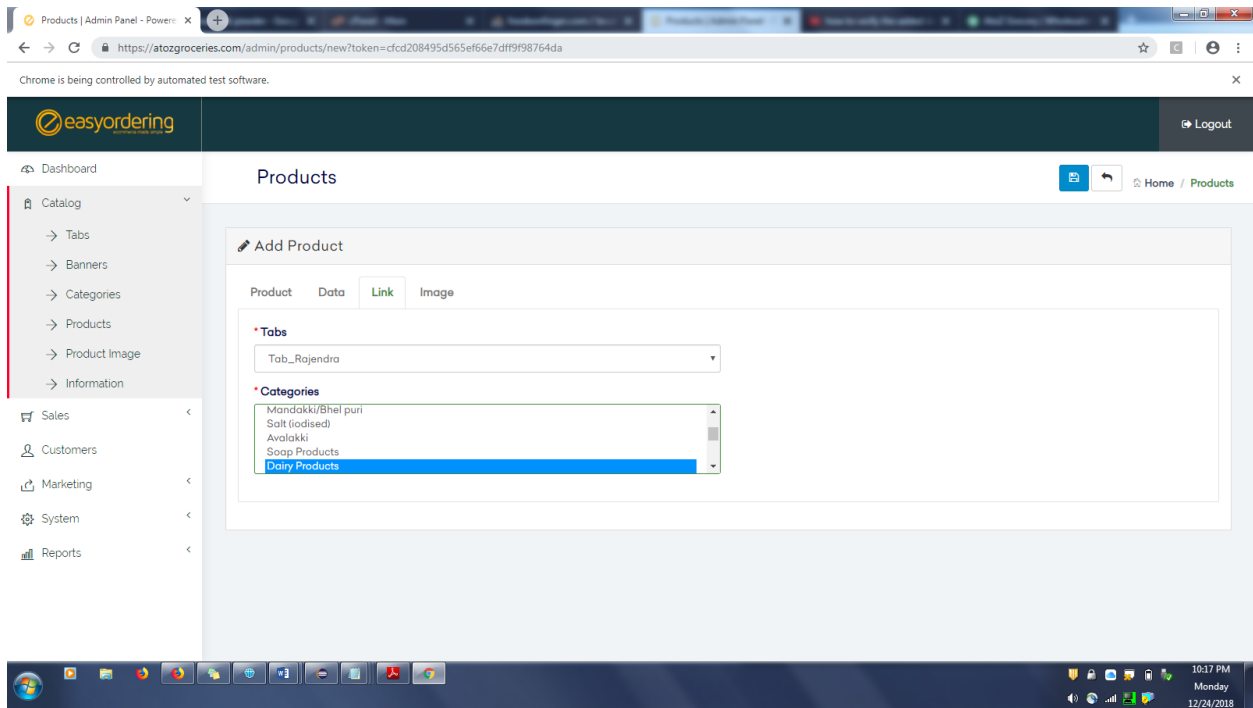
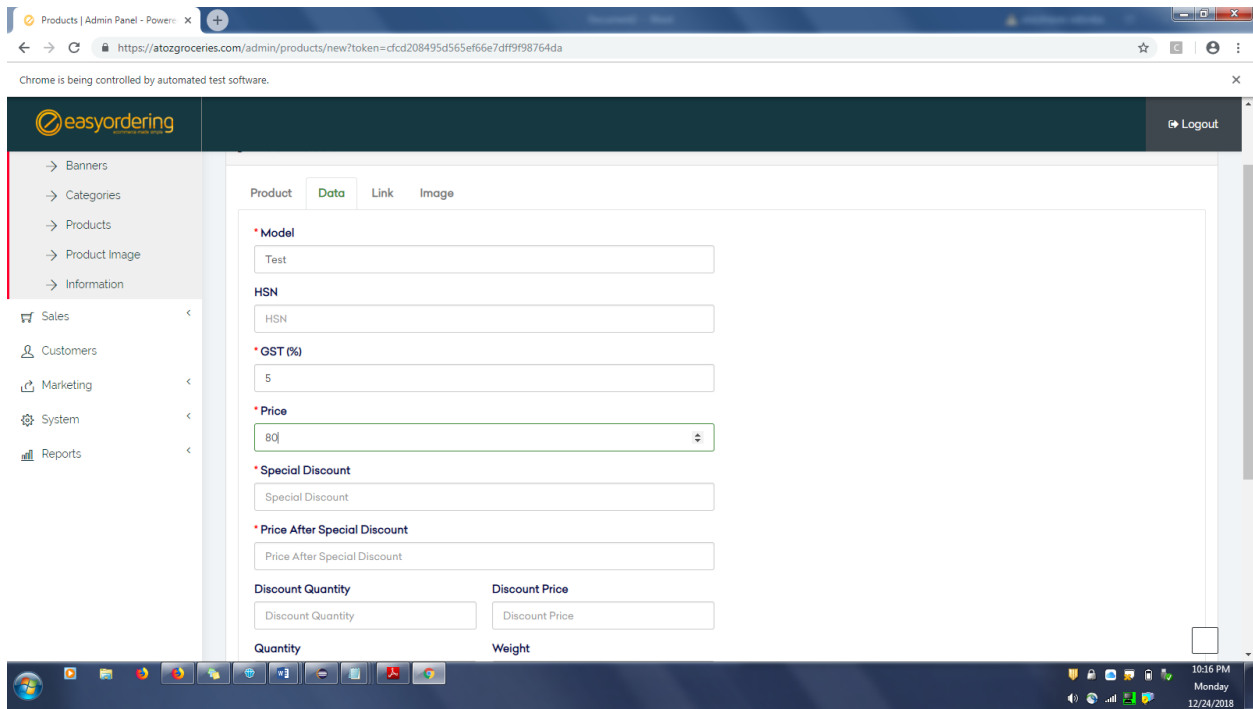
Product Name

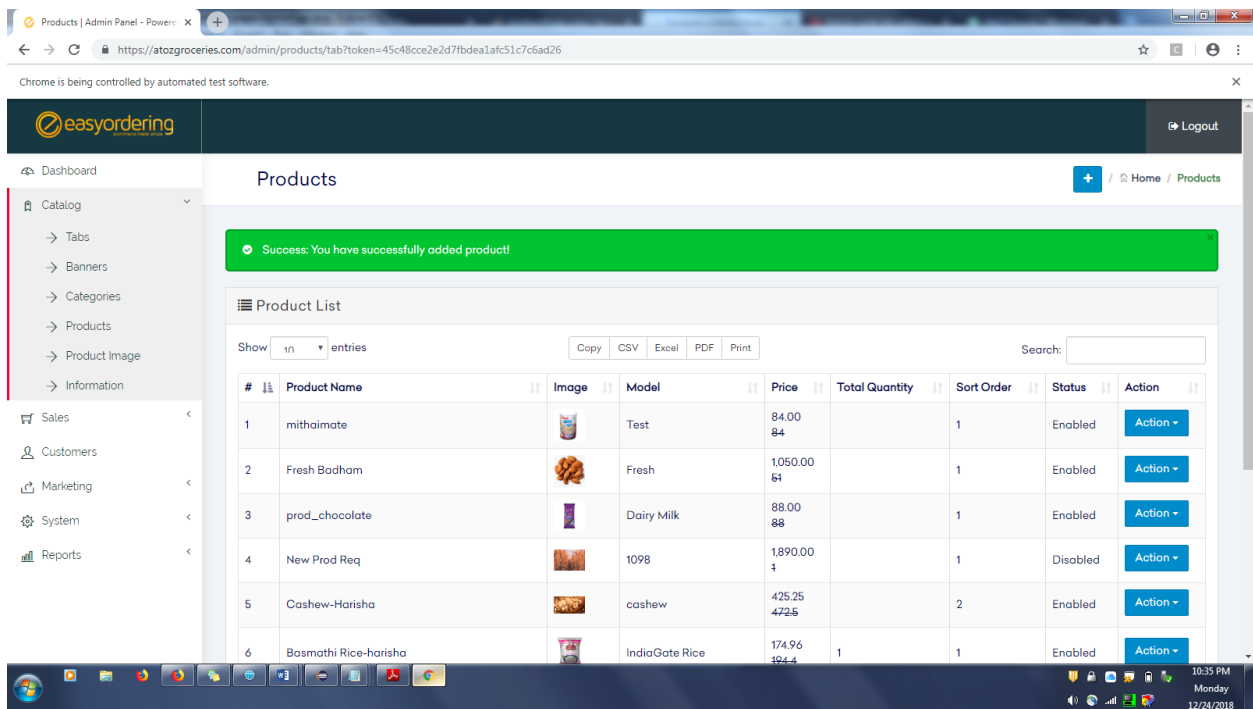
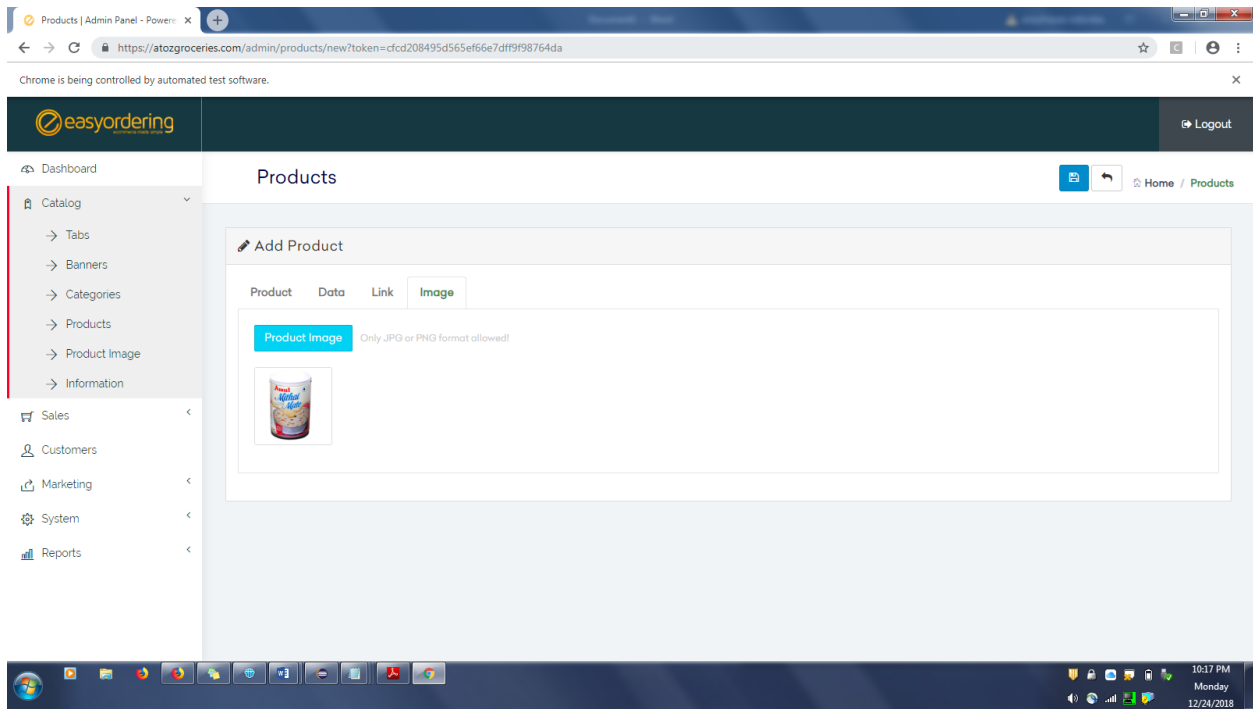
mithalmotel

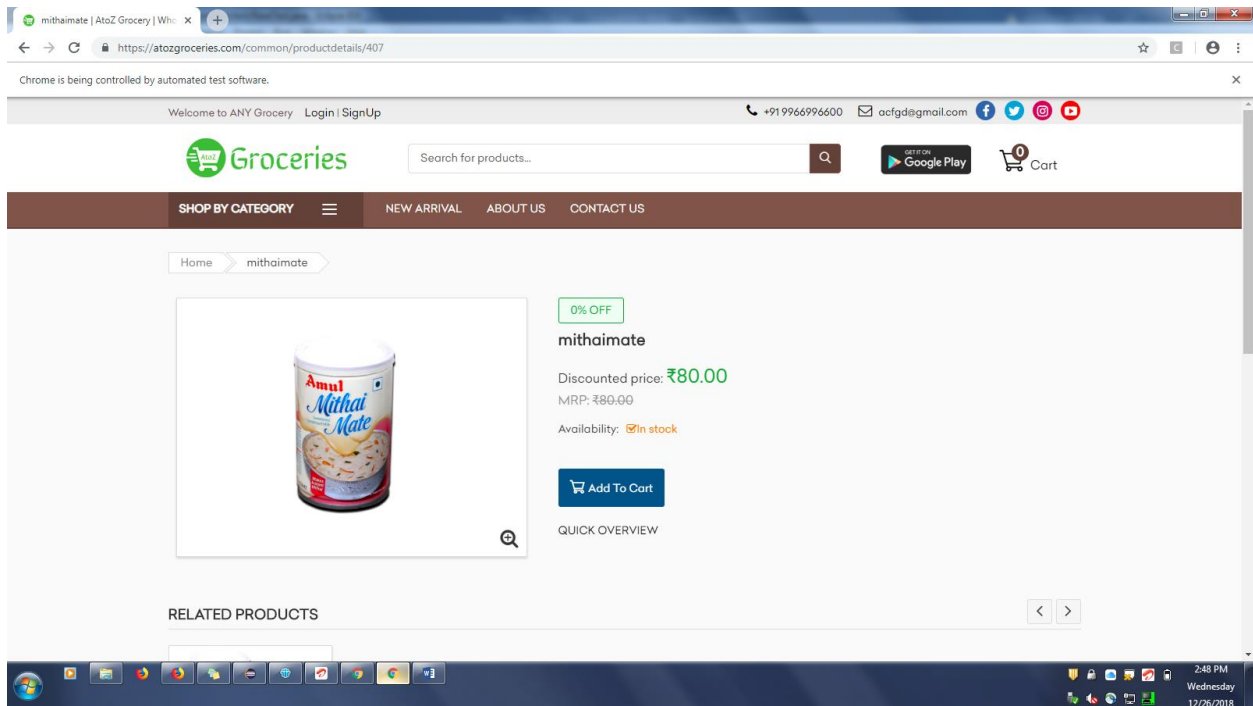
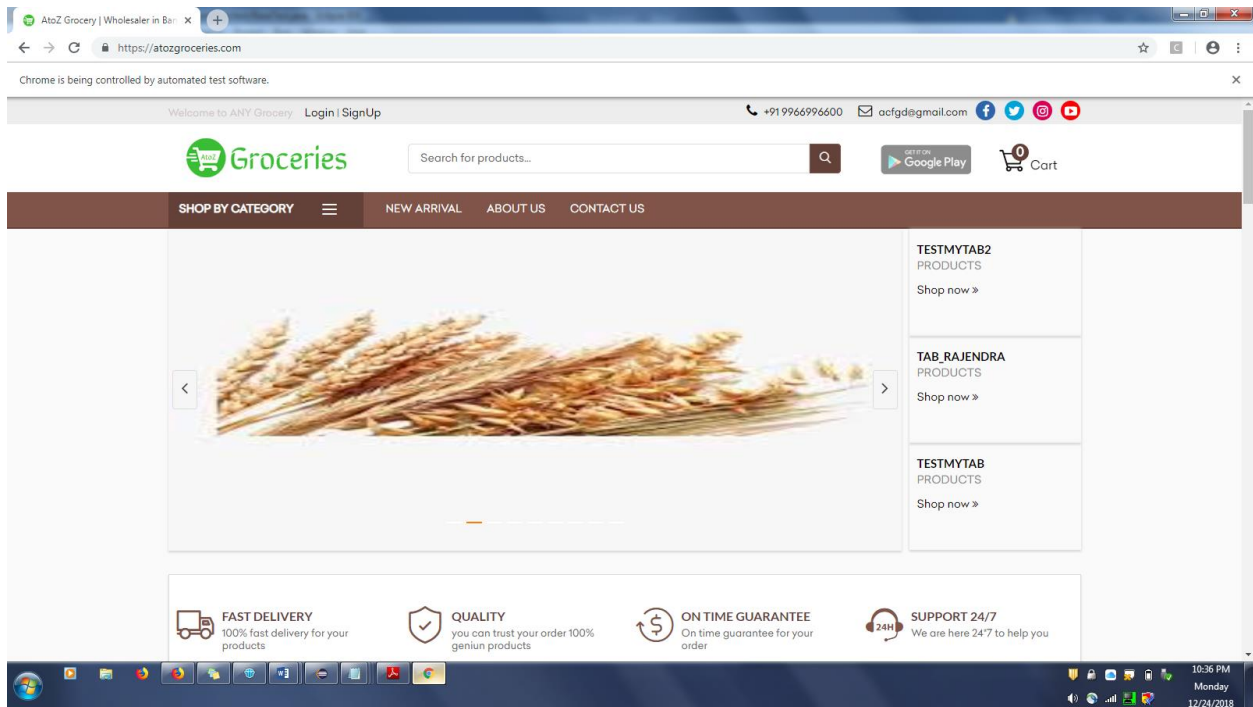
Description



10:14 PM
Monday
12/24/2018







Console Screen Shot :

```
Console  Results of running class BaseTest
<terminated> BaseTest [TestNG] C:\Program Files\Java\jdk-11.0.1\bin\javaw.exe (Dec 26, 2018, 2:37:44 PM)
[RemoteTestNG] detected TestNG version 6.14.3
Starting ChromeDriver 2.44.609538 (b655c5a60b0b544917107a59d4153d4bf78e1b90) on port 34976
Only local connections are allowed.
Dec 26, 2018 2:37:49 PM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected dialect: OSS
mithaimate
QUICK OVERVIEW
mithaimate
mithaimate
PASSED: testcase9

=====
Default test
Tests run: 1, Failures: 0, Skips: 0
=====

=====
Default suite
Total tests run: 1, Failures: 0, Skips: 0
=====
```