

Day 6 - 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.ReturnsPage;
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(dataProvider="EditCategory",testName="EditCategoryValidation")
    public void Day6TestCase(String Old_name,String CategoryName,String
TagTitle,String SortOrder,String Status,String Image) throws IOException
    {
        LoginPage login=new LoginPage(driver,wait);
        DashboardPage dash=new DashboardPage(driver,wait);
        CategoryPage category=new CategoryPage(driver,wait);
        UserPage user=new UserPage(driver,wait);

        try
        {
            //getting the data from the properties file using Hash map
            String adminurl=hashData.get("adminurl");
            String username=hashData.get("username");
            String password=hashData.get("password");
            String userurl=hashData.get("userurl");
            String showdata=hashData.get("show");
            //navigating to the Admin URL of atozgroceries
            driver.get(adminurl);
```

```

        //login page is used to provide the input of username and
password
        login.EnterEmail(username);
        login.EnterPassword(password);
        login.clickLogin();

        //In catalog page we will find the catalog element and click on
that to open categories
        dash.OpenCatalog();

        //category page
        category.clickCategory();
        //Sending input to edit category

        category.editCategory(Old_name,CategoryName,TagTitle,SortOrder,Status,Image);

        //Selecting to show All in categories
        category.displayAllCategories(showdata);

        //validating the Edited category in Admin Page
        String actualMsg=category.dataPageSource();
        Assert.assertTrue(actualMsg.contains(CategoryName), "Assertion on
edited catergory in Admin Page!!!");
        Reporter.Log("Assertion on edited category in Admin Page!!!");
        login.ClickLogout();

        //Validation of Edited category in USER Page
        driver.get(userurl);
        user.CatergoryValidation(CategoryName);

    }
    catch(Exception e)
    {
        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/Day6_TestCase_Error_" + currentDate + ".jpg"));
        System.out.println(e.getMessage());
        Assert.fail();
    }
    finally
    {
        driver.quit();
    }
}
@DataProvider(name="EditCategory")
public Object[][][] Category() throws IOException
{
    return ExcelReader.DataTable("./TestData/TestData.xlsx", "Category");
}
}

```

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();
    }
}

CategoryPage.java
package com.ibm.pages;

```

```

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.Select;
import org.openqa.selenium.support.ui.WebDriverWait;

public class CategoryPage
{
    //Finding the Elements
    @FindBy(xpath="/html/body/div[1]/div[1]/div/ul/li[2]/ul/li[3]/a")
    WebElement categoryEle;

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

    @FindBy(name="name")
    WebElement cateNameEle;

    @FindBy(name="tag_title")
    WebElement tagTitleEle;

    @FindBy(name="sort")
    WebElement sortEle;

    @FindBy(name="status")
    WebElement statusEle;

    @FindBy(xpath="//*[@id='files']")
    WebElement imageEle;

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

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

    @FindBy(xpath="/html/body/div[1]/div[3]/div/form/div[2]/div/div/div[1]/div/h4")
)
    WebElement addCatEle;

    WebDriver driver;
    WebDriverWait wait;
    JavascriptExecutor js;

    //Constructor
    public CategoryPage(WebDriver driver,WebDriverWait wait)
    {
        this.driver=driver;

```

```

        PageFactory.initElements(driver, this);
        js=(JavascriptExecutor) driver;
    }

    //this method is used to click on the Categories under the catalog
    public void clickCategory()
    {
        categoryEle.click();
    }

    // This method is used to add New Category to the catalog
    public void addNewCategory(String name,String tagtitle,String sortorder,String
status)
    {
        //clicking on + sign for adding new category
        addNewEle.click();
        //providing the new category details
        cateNameEle.click();
        cateNameEle.sendKeys(name);
        tagTitleEle.sendKeys(tagtitle);
        js.executeScript("window.scrollBy(0,500)");
        sortEle.sendKeys(sortorder);
        js.executeScript("window.scrollBy(500,1000)");
        Select selectStatus= new Select(statusEle);
        selectStatus.selectByVisibleText(status);

        imageEle.sendKeys("C:\\\\Harisha\\\\Selenium_Training\\\\EclipseWorkSpace\\\\SeleniumF
ramework\\\\TestData\\\\milk_products.jpg");
        driver.findElement(By.id("toTop")).click();
        saveEle.click();
    }
    public String getCategoriesTitle()
    {
        String title= driver.getTitle();
        return title;
    }
    // After saving this will redirect to categories page.

    //showSel will select "ALL" items or display
    public void displayAllCategories(String showvalue) throws InterruptedException
    {
        Select showSel=new Select(showEle);
        showSel.selectByVisibleText(showvalue);
        //scrolling the window to see the products
        js.executeScript("window.scrollBy(0,500)");
        js.executeScript("window.scrollBy(500,1000)");
        js.executeScript("window.scrollBy(1000,2000)");
    }
    public String dataPageSource()
    {
        String dataPgSrc=driver.getPageSource();
        return dataPgSrc;
    }
    //this method will edit a category after clearing the existing data

```

```

        public void editCategory(String old_name,String new_name,String
tagtitle,String sort,String status,String imgPath)
{
    List<WebElement>
rowEle=driver.findElements(By.xpath("//table[@id='dataTableExample2']/tbody/tr"));
    int rows=rowEle.size();
    for(int i=1;i<=rows;i++)
    {
        //getting the category name to edit
        String
name=driver.findElement(By.xpath("//table[@id='dataTableExample2']/tbody/tr["+i+"]/td
[2])).getText().trim();

        if(name.equals(old_name))
        {
            //clicking on the Action button of the desired category to
edit

            driver.findElement(By.xpath("//table[@id='dataTableExample2']/tbody/tr["+i+"]/td[6]/descendant::button")).click();
                //clicking on the Edit button of the desired category to
edit

            driver.findElement(By.xpath("//table[@id='dataTableExample2']/tbody/tr["+i+"]/td[6]/descendant::ul/li)[1]")).click();
                //Clears the existing data
            cateNameEle.clear();
            tagTitleEle.clear();
            sortEle.clear();

            js.executeScript("arguments[0].scrollIntoView(true);",imageEle);
            Select selectStatus= new Select(statusEle);
            selectStatus.selectByVisibleText("");

            js.executeScript("arguments[0].scrollIntoView(true);",cateNameEle);
            //Sending new data for the category
            cateNameEle.sendKeys(new_name);
            tagTitleEle.sendKeys(tagtitle);

            js.executeScript("arguments[0].scrollIntoView(true);",tagTitleEle);
            sortEle.sendKeys(sort);

            js.executeScript("arguments[0].scrollIntoView(true);",imageEle);
            selectStatus.selectByVisibleText(status);
            imageEle.sendKeys(imgPath);
            driver.findElement(By.id("toTop")).click();
            //clicking on Save, will redirect to All categories page
            saveEle.click();
            break;
        }
    }
}
}

```

UserPage.java

```
package com.ibm.pages;

import org.openqa.selenium.By;
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="//*[@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 catHdrEle;
    WebDriver driver;
    WebDriverWait wait;

    //Constructor
    public UserPage(WebDriver driver,WebDriverWait wait)
    {
        this.driver=driver;
        this.wait=wait;
        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=catHdrEle.getText().trim();
        Assert.assertTrue(validCate.contains(catename), "Assertion on Edited
        Category in User Page!!!");
        Reporter.log("Assertion on Edited Category 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;
    }
}

```

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 loginSheet=TestDataWB.getSheet(sheetName);
    Object[][] loginInfo=new Object[loginSheet.getPhysicalNumberOfRows()-
1][loginSheet.getRow(0).getPhysicalNumberOfCells()];
    int rowcount=loginSheet.getPhysicalNumberOfRows();
    int cellcount=loginSheet.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(loginSheet.getRow(i).getCell(j));
            loginInfo[i-1][j]=cellvalue;
        }
    }

    return loginInfo;
}

}

```

data.properties

```

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

```

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="Day6TestCase"></include></methods>
            </class>
        </classes>
    </test> <!-- Test -->
</suite> <!-- Suite -->

```

TestData.xlsx

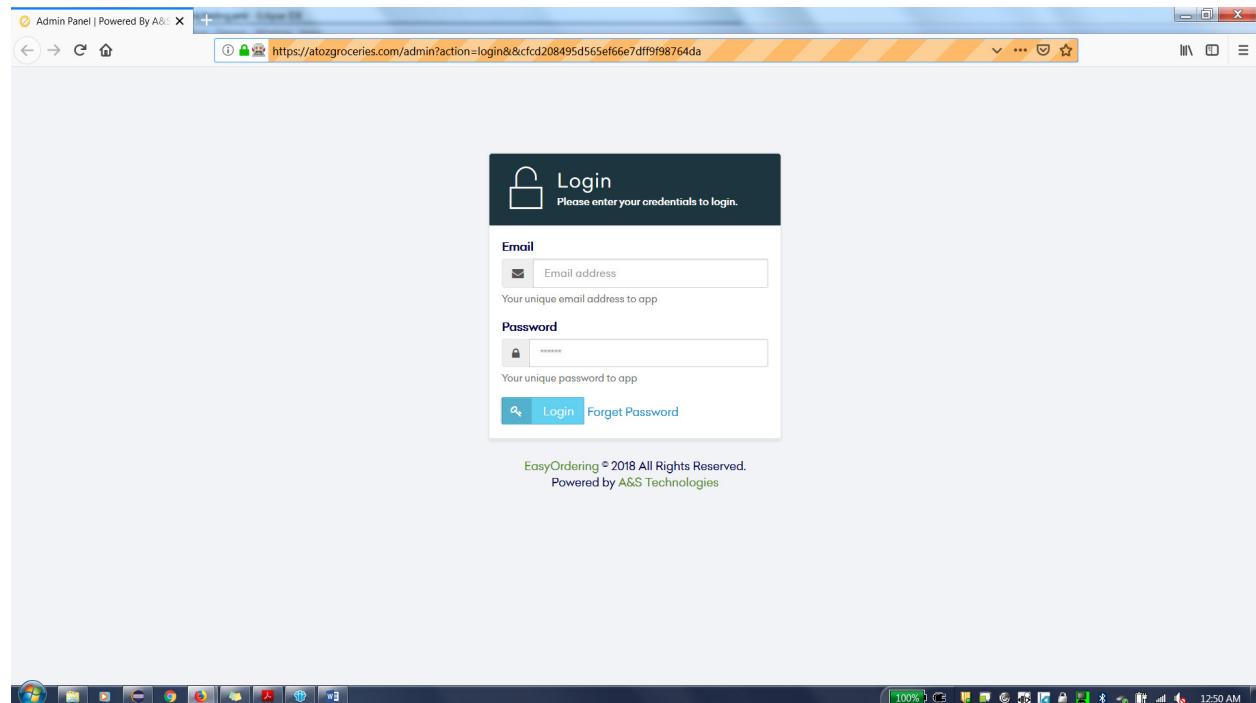
The screenshot shows the Eclipse IDE interface with the following details:

- Project Explorer:** Displays the project structure under "com.ibm".
- Java Editor:** Shows the file `BaseTestJava.java` with imports for `testing.xml`, `CategoryPage.java`, `ExcelReader.java`, `UserPage.java`, and `DriverLaunch.java`.
- Toolbar:** Includes standard Eclipse icons for file operations like Open, Save, and Print.
- Clipboard:** Shows the copied content from the Excel spreadsheet.
- Font and Alignment:** Font dropdown set to "Calibri" and size "11". Alignment dropdown set to "General".
- Cells:** A table with the following data:

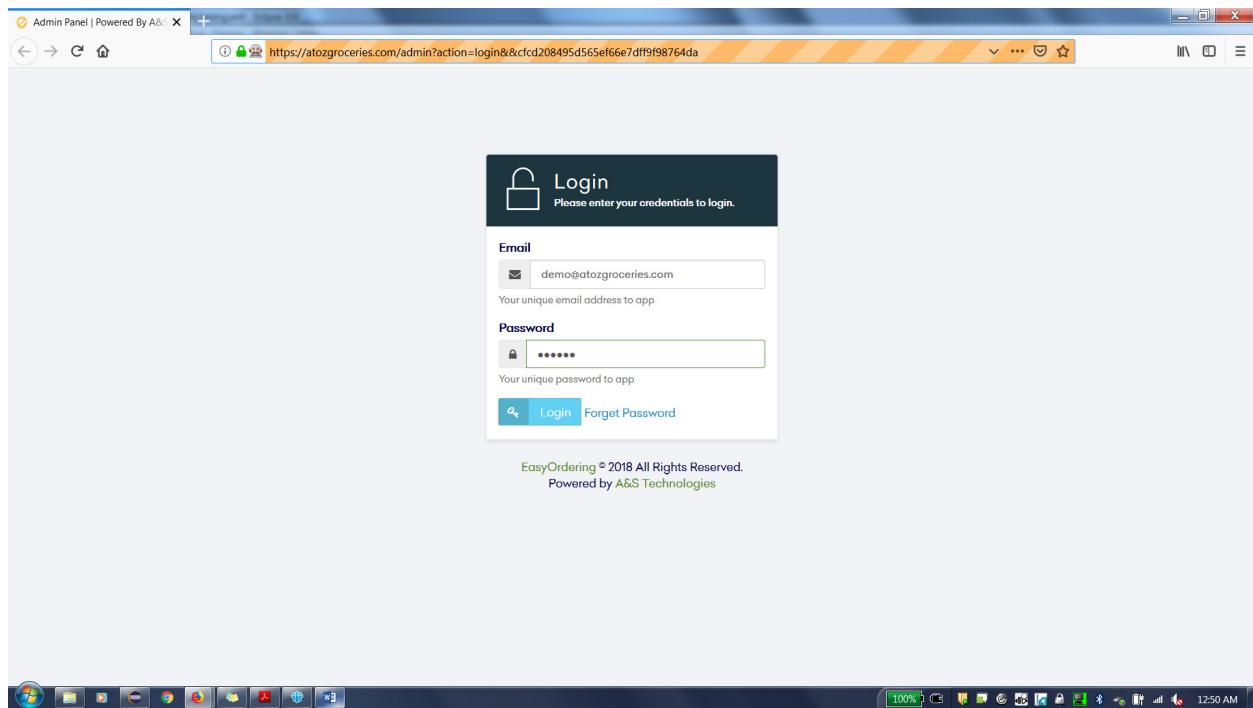
	A	B	C	D	E	F	G	H
1	Old_name	CategoryName	TagTitle	SortOrder	Status	Image		
2	Flours	Wheat flour	Wheat flour	5	Enabled	C:\Harisha\Selenium_Training\EclipseWorkSpace\SeleniumFramework\TestData\wheatFlour.jpg		
3								
4								
5								
6								
7								
8								
9								
10								
11								
12								
13								
14								
15								
16								
17								
18								
19								
20								
21								
22								
23								
24								
25								
26								
- Quick Access:** Shows the status bar with "98%" battery and "1:20 AM" time.

Step by step output:

1. Navigating to the Admin Page of “atozgroceries”



2. Providing login credentials



3. Dashboard after logging into the site

A screenshot of the Admin Panel dashboard. The URL is https://atozgroceries.com/admin/dashboard?token=d3d9446802a44259755d38e6d163e820. The dashboard has a dark header with the 'easyordering' logo and a 'Logout' button. On the left is a sidebar with links: Dashboard, Catalog, Sales, Customers, Marketing, System, and Reports. The main area is titled 'Dashboard' and features four cards: 'Total Orders' (3), 'Total Sales' (₹5,694.87), 'Total Customers' (8), and 'Total Products' (137). Below this is a section titled 'Latest Orders' with a table:

#	Order ID	Customer	Status	Total	Date Added	Action
1	#6	TestNew	Pending	₹3,950.00	2018-12-20	
2	#5		Pending	₹8,700.00	2018-12-20	
3	#3		Pending	₹3,570.00	2018-12-18	
4	#2		Pending	₹3,570.00	2018-12-18	
5	#1		Pending	₹1,425.00	2018-12-10	

4. Opening catalog

The screenshot shows the Admin Panel dashboard for easyordering. On the left, a sidebar menu includes 'Dashboard', 'Catalog' (selected), 'Sales', 'Customers', 'Marketing', 'System', and 'Reports'. The main area displays four cards: 'Total Orders' (5), 'Total Sales' (₹9,215.00), 'Total Customers' (13), and 'Total Products' (223). Below these is a table titled 'Latest Orders' with columns for #, Order ID, Customer, Status, Total, Date Added, and Action. The table lists five pending orders from customers 'TestNew', '#5', '#3', '#2', and '#1'.

#	Order ID	Customer	Status	Total	Date Added	Action
1	#6	TestNew	Pending	₹3,950.00	2018-12-20	
2	#5		Pending	₹8,700.00	2018-12-20	
3	#3		Pending	₹3,570.00	2018-12-18	
4	#2		Pending	₹3,570.00	2018-12-18	
5	#1		Pending	₹1,425.00	2018-12-10	

5. Opening Categories

The screenshot shows the Admin Panel Categories page. The sidebar menu is identical to the dashboard. The main area is titled 'Category List' and displays a table of categories with columns for #, Category Name, Image, Sort Order, Status, and Action. The table lists six categories: 'Pulses Items123', 'Grains', 'Spices', 'Flours', 'Dry Fruits', and 'Dry Red Chilli', each with a corresponding image and status of 'Enabled' or 'Disabled'.

#	Category Name	Image	Sort Order	Status	Action
1	Pulses Items123		2	Disabled	
2	Grains		3	Enabled	
3	Spices		4	Enabled	
4	Flours		5	Enabled	
5	Dry Fruits		6	Enabled	
6	Dry Red Chilli		7	Enabled	

6. Clicking on Action of the desired category for editing

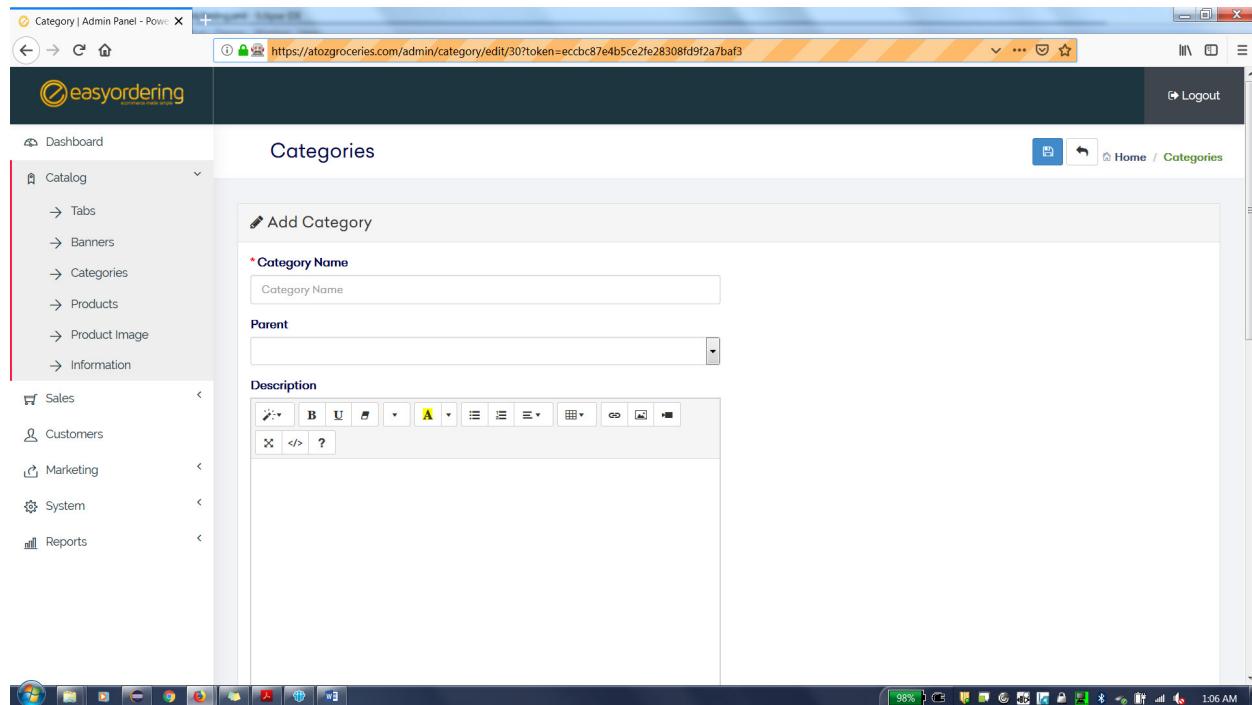
The screenshot shows the 'Categories' list in the easyordering Admin Panel. The table has columns for #, Category Name, Image, Sort Order, Status, and Action. Row 4 represents the category 'Flours'. An 'Action' dropdown menu is open next to the 'Action' column for this row, showing 'Edit' and 'Delete' options.

#	Category Name	Image	Sort Order	Status	Action
1	Pulses Items123		2	Disabled	Action
2	Grains		3	Enabled	Action
3	Spices		4	Enabled	Action
4	Flours		5	Enabled	Action
5	Dry Fruits		6	Enabled	Action
6	Dry Red Chilli		7	Enabled	Action

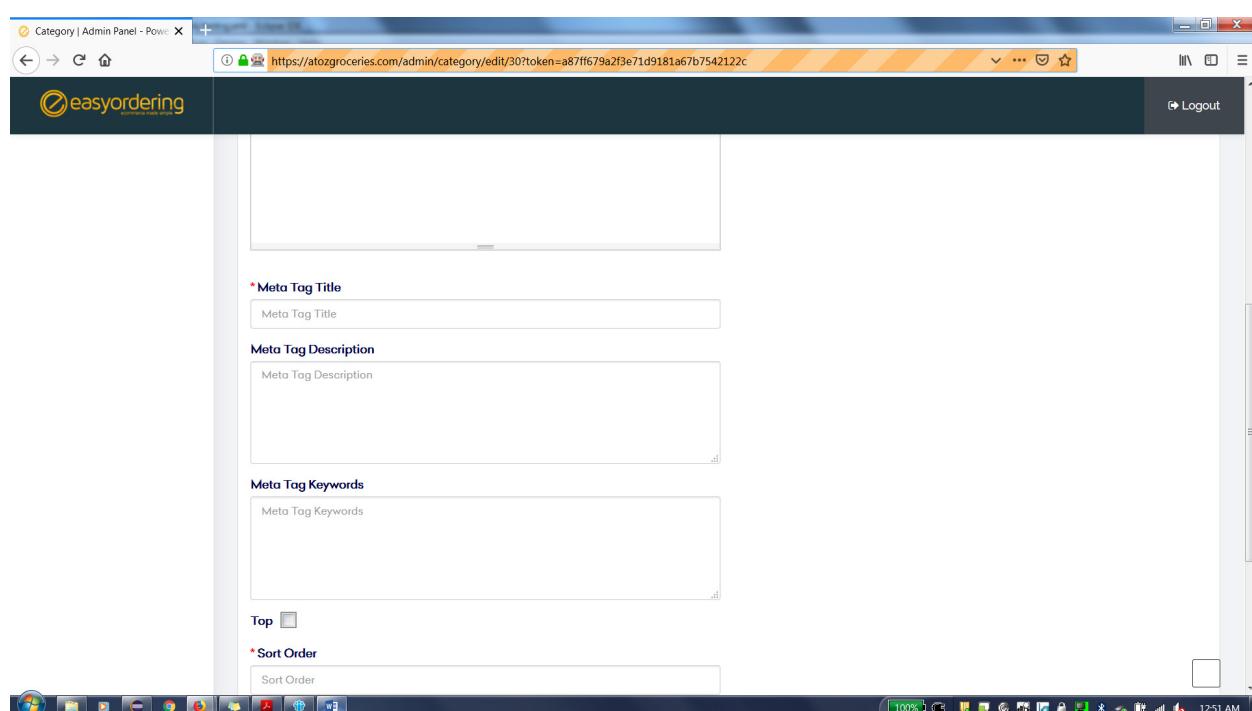
7. Clicking on Edit of the desired category for editing

The screenshot shows the 'Add Category' form in the easyordering Admin Panel. The 'Category Name' field is filled with 'Flours'. The 'Parent' dropdown is open, showing a list of categories. Below the form is a rich text editor for the 'Description' field.

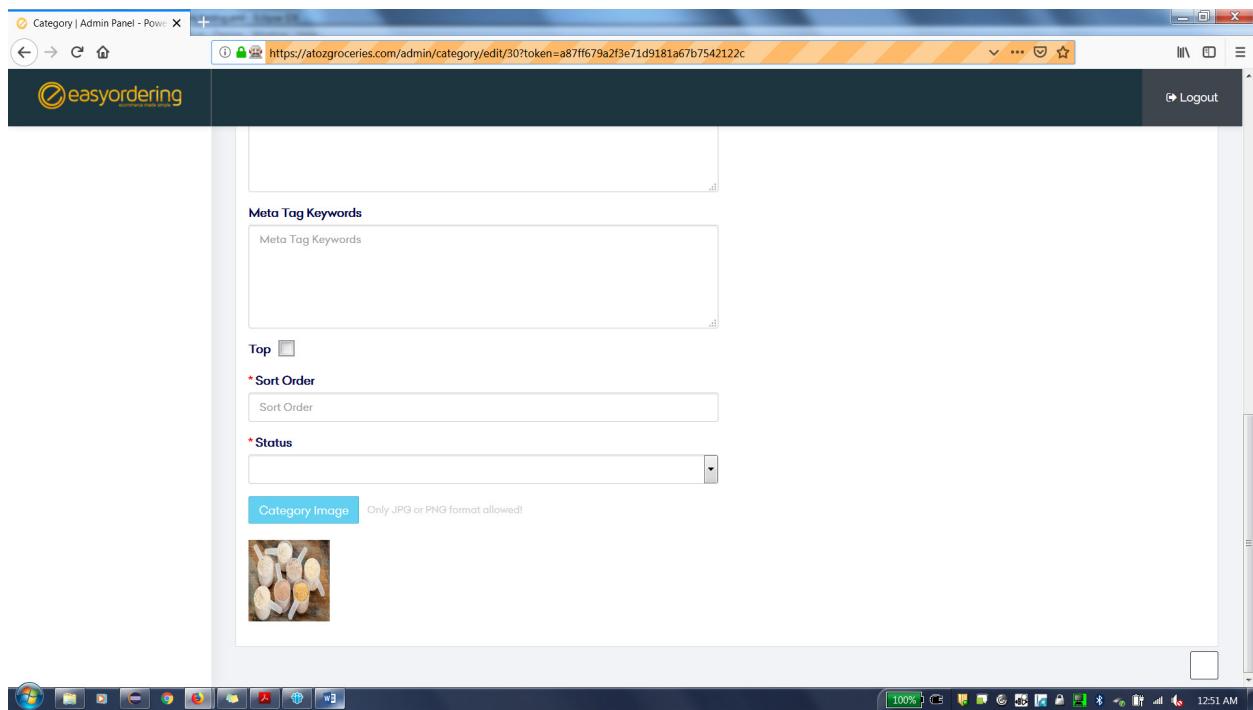
8. Clearing the existing data og the category



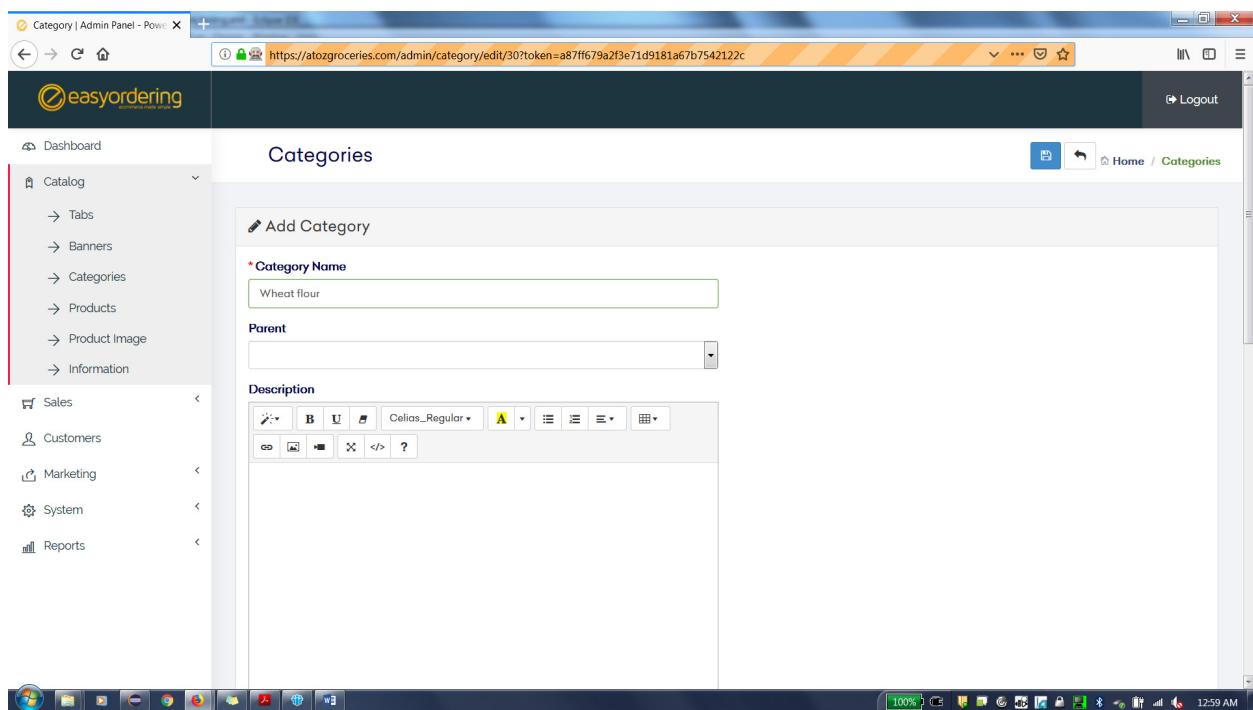
The screenshot shows the 'Categories' section of the easyordering Admin Panel. On the left, a sidebar menu includes 'Dashboard', 'Catalog' (selected), 'Tabs', 'Banners', 'Categories', 'Products', 'Product Image', 'Information', 'Sales', 'Customers', 'Marketing', 'System', and 'Reports'. The main content area is titled 'Add Category' and contains fields for 'Category Name' (with placeholder 'Category Name'), 'Parent' (with a dropdown menu), and 'Description' (with a rich text editor toolbar). The status bar at the bottom shows system icons and the time '10:06 AM'.



The screenshot shows the 'Edit Category' form for a category with ID 30. It includes fields for 'Meta Tag Title' (placeholder 'Meta Tag Title'), 'Meta Tag Description' (placeholder 'Meta Tag Description'), 'Meta Tag Keywords' (placeholder 'Meta Tag Keywords'), a 'Top' checkbox (unchecked), and a 'Sort Order' field (placeholder 'Sort Order'). The status bar at the bottom shows system icons and the time '12:51 AM'.



9. Sending the new Data for the category



Category | Admin Panel - Power

https://atogroceries.com/admin/category/edit/30?token=a87ff679a2f3e71d9181a67b754212c

Logout

easyordering

Product Image

Information

Sales

Customers

Marketing

System

Reports

Description

Meta Tag Title

Wheat flour

Meta Tag Description

Meta Tag Description

100% 12:51 AM

Category | Admin Panel - Power

https://atogroceries.com/admin/category/edit/30?token=a87ff679a2f3e71d9181a67b754212c

Logout

easyordering

Meta Tag Keywords

Meta Tag Keywords

Top

Sort Order

5

Status

Enabled

Category Image

Only JPG or PNG format allowed!

100% 12:51 AM

10. Clicking on Save, it will redirect to All categories page

The screenshot shows the 'Categories' page in the Admin Panel. A green success message at the top right says 'Success: You have successfully updated category!'. Below it is a table titled 'Category List' with columns: #, Category Name, Image, Sort Order, Status, and Action. There are four entries:

#	Category Name	Image	Sort Order	Status	Action
1	Pulses Items123		2	Disabled	Action ▾
2	Grains		3	Enabled	Action ▾
3	Spices		4	Enabled	Action ▾
4	Wheat flour		5	Enabled	Action ▾

11. Selecting "All" from Show to display all categories to validate the presence of modified category

The screenshot shows the 'Categories' page in the Admin Panel. A green success message at the top right says 'Success: You have successfully updated category!'. Below it is a table titled 'Category List' with columns: #, Category Name, Image, Sort Order, Status, and Action. The 'Show' dropdown is set to 'All'. There are four entries:

#	Category Name	Image	Sort Order	Status	Action
1	Pulses Items123		2	Disabled	Action ▾
2	Grains		3	Enabled	Action ▾
3	Spices		4	Enabled	Action ▾
4	Wheat flour		5	Enabled	Action ▾

Category | Admin Panel - Power

<https://atozgroceries.com/admin/category/tab?token=d3d9446802a44259755d38e6d163e820>

26	Vegetables		22	Enabled	Action
27	Soveiries(Mixtures) & Sweets		23	Enabled	Action
28	Masala Items		24	Enabled	Action
29	Papad all types		25	Enabled	Action
30	Vermicel,maggi all fast cookings		25	Enabled	Action
31	Masala , Powders& Sauce		27	Enabled	Action
32	Colours(Edible)		28	Enabled	Action

Previous 1 Next

12. Logging out of the Admin page after successful validation

Admin Panel | Powered By A&S Technologies

<https://atozgroceries.com/admin?action=login&&8114e45fceea167a5a36dedd4bea2543>

ⓘ Success: You have successfully logged out!

Login
 Please enter your credentials to login.

Email

 Your unique email address to app

Password

 Your unique password to app

Login
Forget Password

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

13. Opening User Page for validating the modified Category

The screenshot shows a web browser window for 'AtoZ Grocery | Wholesaler in B...' at <https://atozgroceries.com>. The page features a top navigation bar with links for 'Welcome to ANY Grocery', 'Login | SignUp', contact information (+91 9898989898), email (infotest@atozgroceries.com), social media icons (Facebook, Twitter, Instagram, YouTube), and a 'Cart' icon. Below the navigation is a search bar with placeholder text 'Search for products...'. A 'GET IT ON Google Play' button is also present. The main content area includes a 'SHOP BY CATEGORY' menu, 'NEW ARRIVAL', 'ABOUT US', and 'CONTACT US' links. A large banner for 'DYNAMIC TRADERS' is displayed, stating 'Wholesale Dealers in Rice, Sugar, Salt, Soft Drinks, Pulse'. To the right, there are three promotional boxes: 'TAB_TO_DELETE PRODUCTS' (Shop now), 'TAB_RAJENDRA PRODUCTS' (Shop now), and 'TESTMYTAB PRODUCTS' (Shop now). The bottom of the screen shows a Windows taskbar with various pinned icons and the system tray.

14. Validated the modified category unde Shop By Category in User Page

The screenshot shows the same web browser window as the previous one, but the 'SHOP BY CATEGORY' dropdown menu is open, displaying a list of categories: 'Grains', 'Spices', 'Wheat Flour', 'Dry Fruits', 'Dry Red Chilli', 'Coriander/Daniya', and 'Xxx'. At the bottom of this list is a 'More Categories >' link. The rest of the page content, including the banner, promotional tabs, and footer information, remains the same as in the previous screenshot.

15. Clicking on Wheat Flour for validation

16. Output from console

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE suite SYSTEM "http://testng.org/testng-1.0.dtd">
<suite name="Suite">
<parameter name="browser" value="ff"></parameter>
<classes>
<class name="com.ibm.test.BaseTest">
<methods><include name="Day6TestCase"></include></methods>
</class>
</classes>
</test> <!-- Test -->
</suite> <!-- Suite -->

```

Dec 21, 2018 2:53:26 PM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected dialect: W3C
JavaScript error: https://atozgroceries.com/assets/admin/dist/js/app.min.js, line 1: TypeError: i is null
JavaScript error: https://atozgroceries.com/assets/admin/dist/js/page/dashboard.min.js, line 1: ReferenceError: toastr is not defined
JavaScript error: https://atozgroceries.com/assets/admin/dist/js/app.min.js, line 107: TypeError: document.getElementById(...) is null
JavaScript error: https://atozgroceries.com/assets/admin/dist/js/app.min.js, line 1: TypeError: i is null
JavaScript error: https://atozgroceries.com/assets/admin/dist/js/app.min.js, line 1: ReferenceError: toastr is not defined
JavaScript error: https://atozgroceries.com/assets/admin/dist/js/app.min.js, line 1309: TypeError: document.getElementById(...) is null
1545384225705 Marionette INFO Stopped listening on port 54361
[Parent 10520, Gecko_IOThread] WARNING: pipe error: 109: file z:/build/build/src/ipc/chromium/src/chrome/common/ipc_channel_win.cc, line 346
[Child 5832, Chrome_ChildThread] WARNING: pipe error: 109: file z:/build/build/src/ipc/chromium/src/chrome/common/ipc_channel_win.cc, line 346
[Child 5832, Chrome_ChildThread] WARNING: pipe error: 109: file z:/build/build/src/ipc/chromium/src/chrome/common/ipc_channel_win.cc, line 346
[Parent 10520, Gecko_IOThread] WARNING: pipe error: 109: file z:/build/build/src/ipc/chromium/src/chrome/common/ipc_channel_win.cc, line 346
[Child 9780, Chrome_ChildThread] WARNING: pipe error: 109: file z:/build/build/src/ipc/chromium/src/chrome/common/ipc_channel_win.cc, line 346
[Child 9780, Chrome_ChildThread] WARNING: pipe error: 109: file z:/build/build/src/ipc/chromium/src/chrome/common/ipc_channel_win.cc, line [GPU 10604, Chr
[Child][MessageChannel::SendAndWait] Error: Channel error: cannot send/recv

Suite
Total tests run: 1, Failures: 0, Skips: 0

17. Automated Emailable report

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "TestNG Report". The content of the page is an automated report generated by TestNG. It includes:

- A summary table:

Test	# Passed	# Skipped	# Failed	Time (ms)	Included Groups	Excluded Groups
Suite						
Test	1	0	0	103,569		
- Detailed test results:

Class	Method	Start	Time (ms)
Suite			
Test — passed			
com.ibm.test.BaseTest	Day6TestCase	1545334133582	97509
- Messages section:

Assertion on edited category in Admin Page!!
Assertion on Edited Category in User Page!!



18. Automated Index report

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "TestNG Report". The content of the page is an automated index report generated by TestNG. It includes:

- A navigation bar with "All suites":
 - Suite**
 - Info
 - C:\Harisha\Selenium_Training\EclipseWorkSpace\SeleniumFramework\testing.xml
 - 1 test
 - 0 groups
 - 0 Times
 - Reporter output
 - Ignored methods
 - Chronological view
 - Results
 - 1 method, 1 passed
 - Passed methods (hide)
 - Day6TestCase(Flours, Wheat flour, Wheat flour, 5, Enabled, C:\Harisha\Selenium_Training\EclipseWorkSpace\Selenium...)
 - Reporter output for Suite:

Day6TestCase(Flours, Wheat flour, Wheat flour, 5, Enabled, C:\Harisha\Selenium_Training\EclipseWorkSpace\Selenium...)
Assertion on edited category in Admin Page!! Assertion on Edited Category in User Page!!



*** End of the document ***