

Star Agile Capstone project

Project name : Project – Human Resource Management(Automation)

Project Done By : Mallika(mallikakpsgvc@gmail.com)

Date: 6th November 2024

Modules In Scope for Automation Testing

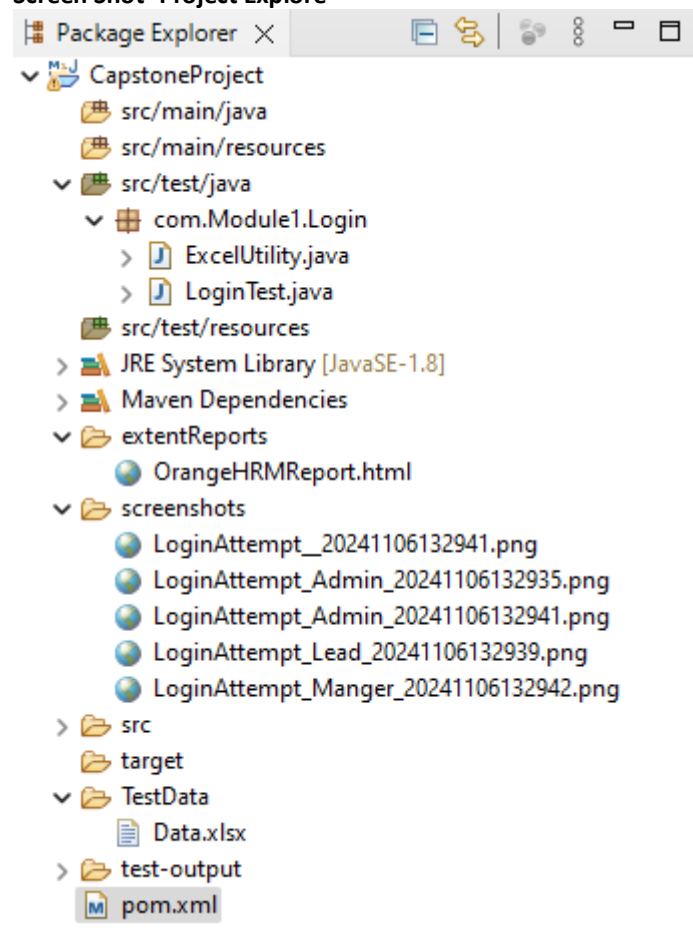
1. Login
2. Admin

Module 1.Login

Scenario1: Automate Orange HRM login and logout with 5 different data set including valid and invalid data set.

1. Save data set in Excel file.
2. Write a script to read data from excel.
3. Prepare script for Login and logout.
4. Perform assertion for valid data set (use Username: Admin and Password:admin123) test case should be pass and invalid data set (other than given data) test case should be fail.
5. Capture Screenshot for every login functionality.
6. Generate Extent Report for the same.

Screen Shot -Project Explore



Screen Shot -Pom.xml file:

Added necessary dependencies(Selenium,Maven,TestNG, Apache poi and Extent Reports).

```

1  CapstoneProject/pom.xml X
2  <modelVersion>4.0.0</modelVersion>
3  <groupId>CapstoneProject</groupId>
4  <artifactId>CapstoneProject</artifactId>
5  <version>0.0.1-SNAPSHOT</version>
6  <dependencies>
7  <!--
8  https://mvnrepository.com/artifact/org.seleniumhq.selenium/selenium-java -->
9  <dependency>
10 <groupId>org.seleniumhq.selenium</groupId>
11 <artifactId>selenium-java</artifactId>
12 <version>4.25.0</version>
13 </dependency>
14 <!-- https://mvnrepository.com/artifact/org.testng/testng -->
15 <dependency>
16 <groupId>org.testng</groupId>
17 <artifactId>testng</artifactId>
18 <version>7.10.2</version>
19 <scope>test</scope>
20 </dependency>
21 <!-- https://mvnrepository.com/artifact/org.apache.poi/poi -->
22 <dependency>
23 <groupId>org.apache.poi</groupId>
24 <artifactId>poi</artifactId>
25 <version>5.3.0</version>
26 </dependency>
27 <!-- https://mvnrepository.com/artifact/org.apache.poi/poi-ooxml -->
28 <dependency>
29 <groupId>org.apache.poi</groupId>
30 <artifactId>poi-ooxml</artifactId>
31 <version>5.3.0</version>
32 </dependency>
33 <!-- https://mvnrepository.com/artifact/com.aventstack/extentreports -->
34 <dependency>
35 <groupId>com.aventstack</groupId>
36 <artifactId>extentreports</artifactId>
37 <version>5.1.2</version>
38 </dependency>

```

Screen Shot: Excel File Data

[illegible]

Screen Shot- Excel Utility java class

```
ExcelUtility.java X
2 import java.io.File;
3 import java.io.FileInputStream;
4 import java.io.IOException;
5 import org.apache.poi.xssf.usermodel.XSSFWorkbook;
6 import org.testng.annotations.DataProvider;
7 public class ExcelUtility {
8     private XSSFWorkbook wb;
9     private FileInputStream fs;
10    private File f1;
11    @DataProvider(name="excelLoginData")
12    public Object[][] excelData()
13    {
14        f1=new File(System.getProperty("user.dir")+"//TestData//Data.xlsx");
15        Object arr[][]=null;
16        try {
17            fs = new FileInputStream(f1);
18            //Workbook
19            wb=new XSSFWorkbook(fs);
20            String uName1=wb.getSheet("userData").getRow(1).getCell(0).getStringCellValue();
21            System.out.println("User name: "+uName1);
22            int rows=wb.getSheet("userData").getPhysicalNumberOfRows();
23            System.out.println("No. of Rows: "+rows);
24            int cells=wb.getSheet("userData").getRow(0).getPhysicalNumberOfCells();
25            System.out.println("No. of Columns: "+cells);
26            arr=new Object[rows-1][cells];
27            for(int i=1;i<rows;i++)//to skip heading i starts with 1
28            {
29                for(int j=0;j<cells;j++)
30                {
31                    arr[i-1][j]=wb.getSheet("userData").getRow(i).getCell(j).getStringCellValue();
32                    System.out.print(arr[i-1][j]+" ");
33                }System.out.println();
34            }
35        } catch (IOException e) {
36            e.printStackTrace();
37        }
38        return arr;
39    }
40    public void closeWorkbook() {
41        if (wb != null) {
42            try {
43                wb.close();
44            } catch (IOException e) {
45                e.printStackTrace();
46            }
47        } else {
48            System.out.println("Workbook is already null or was never initialized.");
49        }
50    }
51 }
```

Screen shot- LoginTest TestNg File

```
3 import java.io.File;
24
25 Run All
26 public class LoginTest {
27     WebDriver driver;
28     ExcelUtility excelData;
29     ExtentReports extent;
30     ExtentTest test;
31
32    @BeforeClass
33    public void setUp() {
34        excelData= new ExcelUtility();
35        driver = new ChromeDriver();
36        driver.get("https://opensource-demo.orangehrmlive.com/web/index.php/auth/login");
37        driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(10));
38        driver.manage().window().maximize();
39        ExtentSparkReporter sparkReporter = new ExtentSparkReporter("extentReports/OrangeHRMReport.html");
40        extent = new ExtentReports();
41        extent.attachReporter(sparkReporter);
42    }
43 }
```

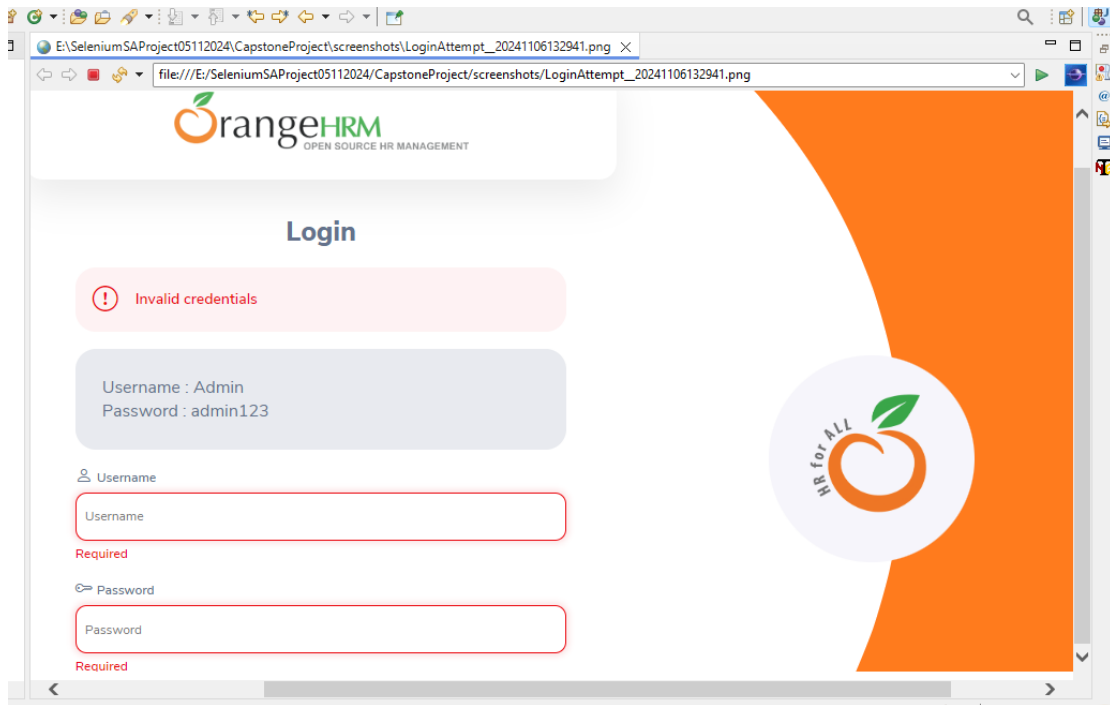
```

44 @Test(dataProvider = "excelLoginData", dataProviderClass = ExcelUtility.class)
45 Run | Debug
46 public void loginTest(String username, String password) {
47     test = extent.createTest("Login Test with username: " + username);
48     driver.findElement(By.name("username")).sendKeys(username);
49     driver.findElement(By.name("password")).sendKeys(password);
50     driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(20));
51     driver.findElement(By.xpath("//button[normalize-space(text()='Login')]")).click();
52     driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(30));
53
54     // Assertion and validation
55     if (username.equals("Admin") && password.equals("admin123")) {
56         Assert.assertEquals(driver.getCurrentUrl(), "https://opensource-demo.orangehrmlive.com/web/index.php/dashboard/index");
57         WebDriverWait wait = new WebDriverWait(driver, Duration.ofSeconds(20));
58         wait.until(ExpectedConditions.visibilityOfElementLocated(By.xpath("//h6[text()='Dashboard']")));
59         takeScreenshot("LoginAttempt_" + username);
60         test.pass("Valid credentials - Login successful.");
61         logout();
62     }
63     else {
64         WebElement errorMessage =
65             driver.findElement(By.xpath("//p[text()='Invalid credentials']"));
66         Assert.assertTrue(errorMessage.isDisplayed(), "Invalid credentials");
67         driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(30));
68         takeScreenshot("LoginAttempt_" + username);
69         test.fail("Invalid credentials - Login failed as expected.");
70     }
71 }
72
73 public void logout() {
74     driver.findElement(By.xpath("//span[@class='oxd-userdropdown-tab']")).click();
75     driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(10));
76     driver.findElement(By.xpath("//li/a[text()='Logout']")).click();
77     driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(10));
78 }
79
80 public void takeScreenshot(String testName) {
81     File screenshot = ((TakesScreenshot) driver).getScreenshotAs(OutputType.FILE);
82     String timestamp = new SimpleDateFormat("yyyyMMddHHmmss").format(new Date());
83     String filePath = "Screenshots/" + testName + "_" + timestamp + ".png";
84     try {
85         FileUtils.copyFile(screenshot, new File(filePath));
86         test.addScreenCaptureFromPath(filePath);
87     } catch (IOException e) {
88         e.printStackTrace();
89     }
90 }
91
92 @AfterClass
93 public void tearDown() {
94     excelData.closeWorkbook();
95     if (driver != null) {
96         driver.quit();
97     }
98     extent.flush();
99 }
100 }

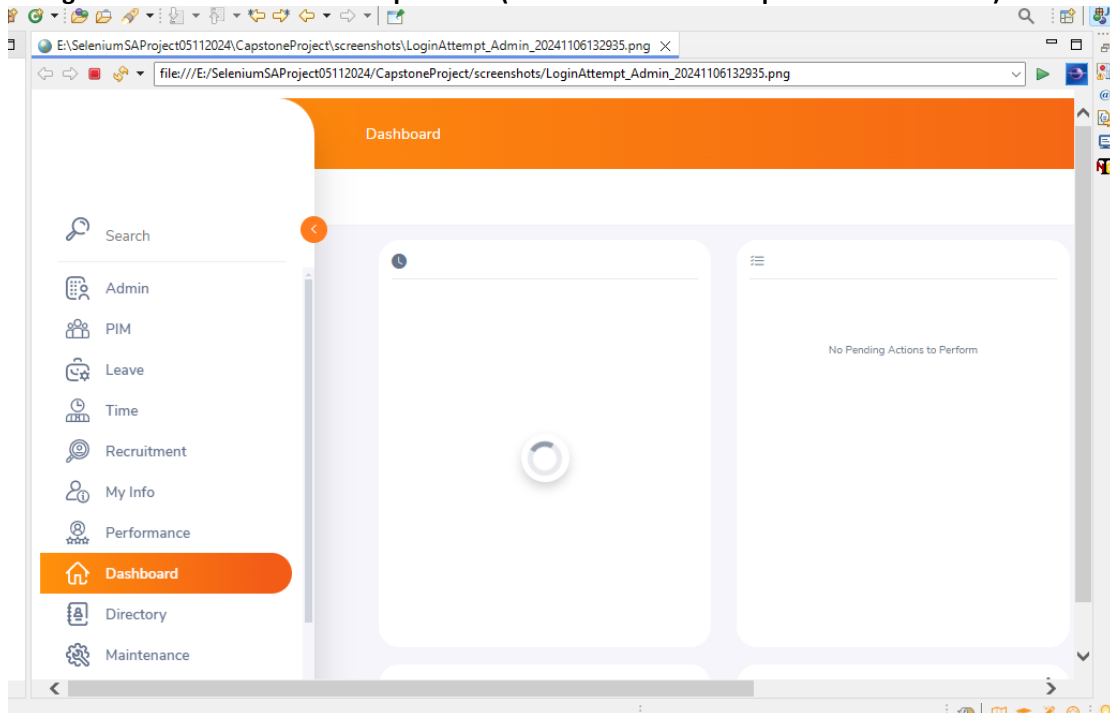
```

Screen Shot- Captured Screenshot for every login functionality

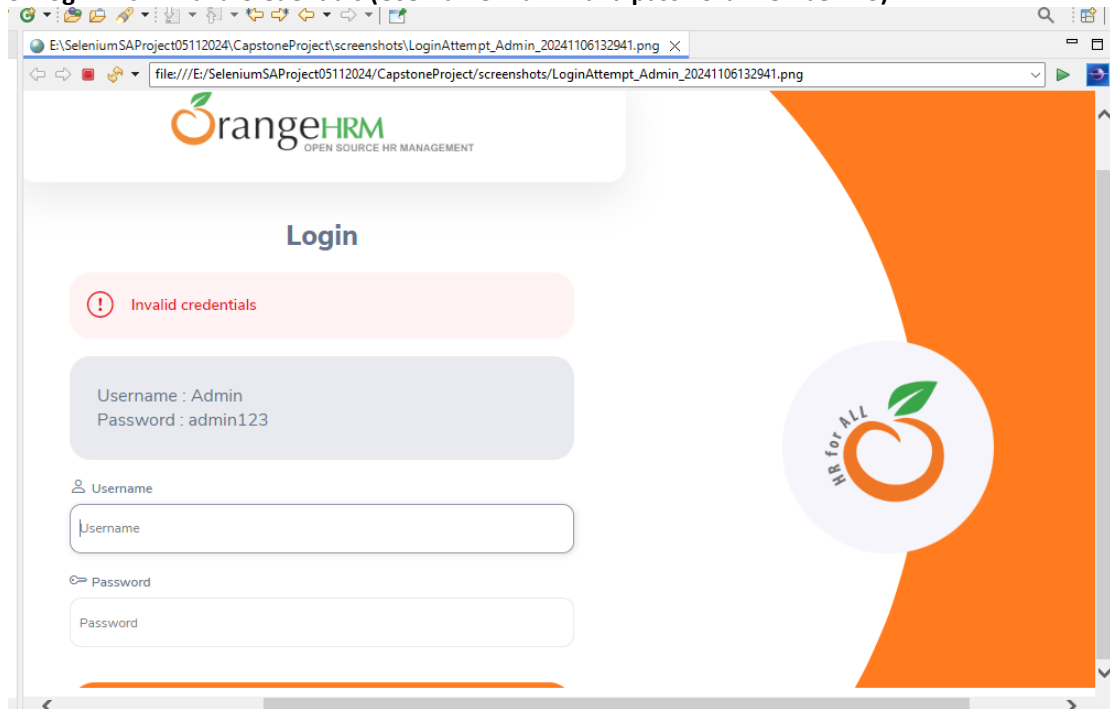
1. Login with blank data



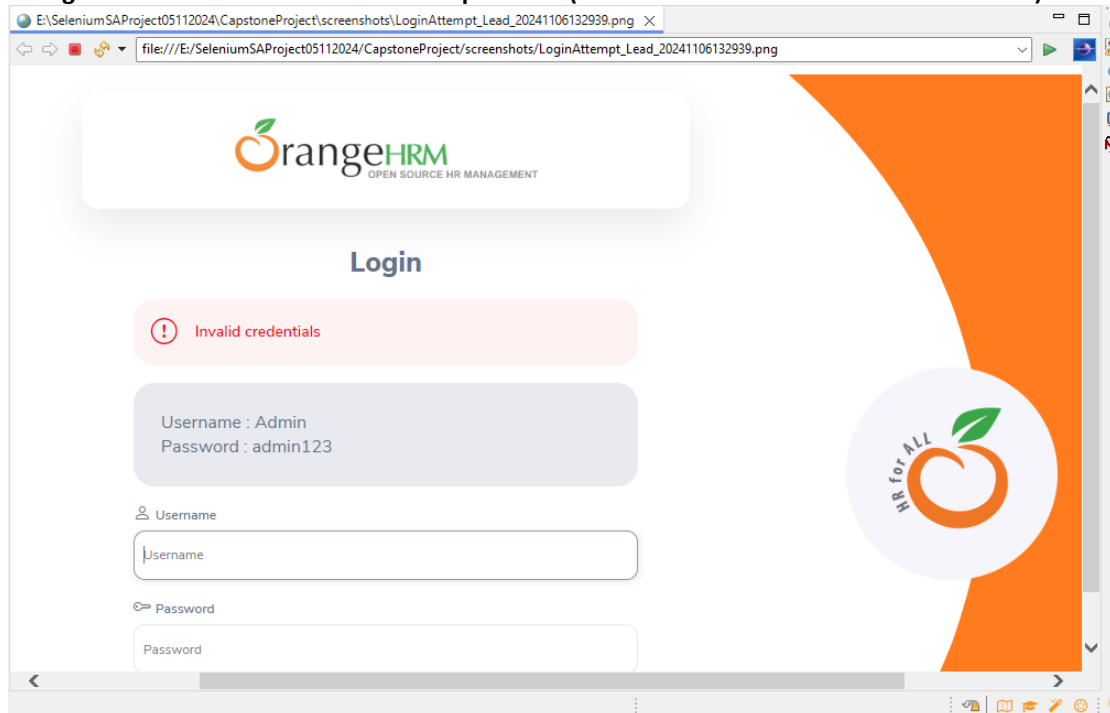
2. Login with Valid User name and password(Username: Admin and password: admin123)



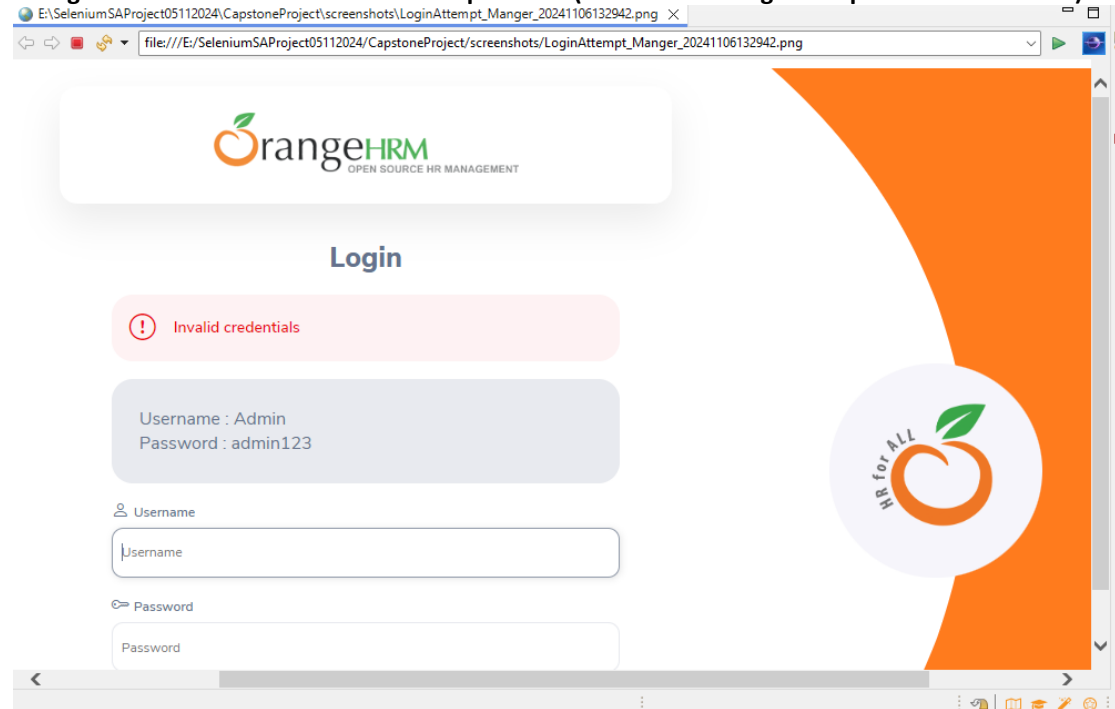
3. Login with Invalid Credentials (Username: Admin and password:member123)



4. Login with Invalid username and valid password (Username: Lead and Password: admin123)



5. Login with Invalid username and valid password (Username: Manger and password: admin123)



Screen shot-Console Output:

```
Problems Javadoc Declaration Console X Results of running class LoginTest
<terminated> LoginTest [TestNG] C:\Program Files\Java\jdk-21\bin\javaw.exe (06-Nov-2024, 1:29:24 pm - 1:29:43 pm) [pid: 33184]
SLF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder".
SLF4J: Defaulting to no-operation (NOP) logger implementation
SLF4J: See http://www.slf4j.org/codes.html#StaticLoggerBinder for further details.
[RemoteTestNG] detected TestNG version 7.10.2
Nov 06, 2024 1:29:28 PM org.openqa.selenium.devtools.CdpVersionFinder findNearestMatch
WARNING: Unable to find an exact match for CDP version 130, returning the closest version; found: 129; Please update to a Selenium version that supports CDP version 130
2024-11-06T07:59:32.346043200Z main ERROR Log4j2 could not find a logging implementation. Please add log4j-core to the classpath. Using SimpleLogger to log to the console...
User name: Admin
No. of Rows: 6
No. of Columns: 2
Admin admin123
Lead lead123
Admin member123

Manger admin123
Workbook is already null or was never initialized.
PASSED: com.Module1.Login.LoginTest.loginTest("Lead", "lead123")
PASSED: com.Module1.Login.LoginTest.loginTest("Admin", "admin123")
PASSED: com.Module1.Login.LoginTest.loginTest("", "")
PASSED: com.Module1.Login.LoginTest.loginTest("Manger", "admin123")
PASSED: com.Module1.Login.LoginTest.loginTest("Admin", "member123")

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

=====
Default suite
Total tests run: 5, Passes: 5, Failures: 0, Skips: 0
=====
```

Screen shot - Extent Reports

