

AUTOMATION TESTING USING SELENIUM TOOL

PROJECT : ONLINE MATRIMONIAL SERVICE

NAME : VASHANTH TS

EMP ID : 12221

W3H ANALYSIS

PROJECT: ONLINE MATRIMONIAL SERVICE	
What?	How?
1.What are the modules Required? ANS: (a). Application Admin Module (b). Match Maker Module (c). User Module	1.Admin: a) Verify Login Method 1: Verify login using email id and password. Method 2: Verify login using Social media account / username and password.
2.What are the functionalities required for Admin module? ANS: 1. Verifies User Login to the application using Email Id and Password. 2. Perform CRUD functions to Users Profile and Match making module.	b) CRUD Method 1: Implement the create/delete / view / update in Application by manually selecting the User email /Registered id as it is unique. Method 2: Implement the create/ delete / view / update for application by selecting username from listings. Method 3: Implement the create/delete / view / update by listing all the details of Grooms and Brides as a table and providing an individual create match for users and edit/delete button for user .
3.What are the functionalities required for Match Maker Module? ANS: 1. Login to the Application using Registered id and password 2. Can view User Profile details . 3. Can Perform CRUD on users .	2.MatchMaker module: a) Login Method 1: Login using Registered id and password. Method 2: login using username and password.
4.What are the Functionalities required for User Module? ANS: 1. Login to the portal using User name/Email id and Password. 2. Select matches for thier profile ,Can view, edit their profile and Provide Ratings. 3. Can Contact with the mutually matched profiles .	b) View and Match the Profiles Method 1: Match profiles of users according to the requests received in application and send notification about it and share the Brides/Grooms profiles as Matched , for further process.
5.What are the Fields required for User Module? ANS: User Name, Email , Password , Mobile number, Address, Age, Job/Business , Salary , Property Details , Family details , Marital Status, Religion, Astrology and Horoscopic Details .	c)Perform CRUD on Matchings Method 1: Manually entering in the Matched Category in Application by listing profile details . Method 2: Perform CRUD as a Entire list for every match.
7.What are the Fields required for Match Maker Module? ANS: Registered Id , Name ,Password , Work Timings , Address , Contact details .	3.User : a) Login Method 1: login using User email id and password Method 2: login using Username and password
8.What are the Fields required for Admin Module? ANS: Admin id , Password , User details , Grooms ,Brides ,Matched Profiles, Match Makers details.	b) Select matching pair's profile, Can View / Edit their Profiles Method 1: Can View and Edit profiles in application by Log in to the portal , Select matches from the specific section or Search by filter options. Method 2: Can view matches in Dashboard and then log in and continue for further proceedings.
WHY ?	WHY NOT ??

<p>1.Admin:</p> <p>a) Verify Login</p> <p>Method 1: Verify login using email id and password. (It makes Admin to verify and authenticate the originality of the user login.)</p> <p>b) CRUD</p> <p>Method 1: Implement the create/delete / view / update in Application by manually selecting the User email /Registered id as it is unique.</p> <p>(It is easy for admin to enter the user id and perform the crud operation on portal)</p> <p>2. Match Maker module:</p> <p>a) Login</p> <p>Method 1: login using Registered Id and password. (Login using id and password is necessary)</p> <p>b) View and Match the Profiles</p> <p>Method 1: Match profiles of users according to the requests received in application and send notification about it and share the Brides/Grooms profiles as Matched , for further process.</p> <p>(It gives clear details to the users to choose correct match and is easy to follow up further)</p> <p>c)Perform CRUD on Matchings</p> <p>Method 1: Manually entering in the Matched Category in Application by listing profile details . (It reduces the time in searching matched profiles from Grooms/Brides category from list)</p> <p>3.User :</p> <p>a) Login</p> <p>Method 1: login using User email id and password (Login using email id and Password is efficient to authenticate the originality of the user.)</p> <p>b) Select matching pair's profile, Can View / Edit their Profiles</p> <p>Method 1: Can View and Edit profiles in application by Log in to the portal , Select matches from the specific section or Search by filter options.</p> <p>(It reduces the time and makes match easy and faster , without login user cant contact or interact and proceed)</p> <p>c) Connect & Interact</p> <p>Method 2: Select the mutually matched profiles and make a connect , then interact ,Proceed further (After selecting the mutually matched profiles and contact them by sharing details will be a good choice , so that user can know each other personally and avoid misunderstandings and can do cross verifications)</p>	<p>Admin:</p> <p>1) To verify the originality of the user it is highly recommended to login using email id/ mobile number and Password so that username and social media account cannot be great option as it involves payment options.</p> <p>2) Crud operations using list takes much time for data retrieval .</p> <p>3) If there is a table maintained for each match of user , it makes the Database clumsy and creates ambiguity issues and takes more memory space to combine entire matches .</p> <p>Match Maker Module :</p> <p>A) It is efficient to login using Registered Id and Password ,so that username cannot be great option.</p> <p>B) Performing CRUD from the entire list takes much time .</p> <p>User Module :</p> <p>A) It is recommended to login using email id and Password , because several users may have similar user name so that username cannot be great option. Avoid using social media account . Login with registration id make admin to know this is a Bride / Groom account and then they can provide profiles according to it..</p> <p>B)Viewing matched profiles without log in is not a efficient method and user can not able to contact.</p> <p>C)Selecting all the profiles at a time and proceeding for connect is wastage of time and service .Updating After Every Process makes easier to manage application and service even more good</p>
---	--

SIGN UP

```
class SignUpTest {
    WebDriver driver;

    @BeforeEach
    public void setUp() {
        driver = new ChromeDriver();
    }

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

    @Test
    @Order(1)
    void testSignup() throws InterruptedException {
        WebDriver driver = new ChromeDriver();
        driver.get("http://localhost:3000/");
        driver.manage().window().maximize();
        driver.findElement(By.linkText("SignUp")).click();

        driver.findElement(By.id("uname")).sendKeys("vasi");

        driver.findElement(By.id("email")).sendKeys("vasi@gmail.com");

        driver.findElement(By.id("pwd")).sendKeys("vasi@123");

        driver.findElement(By.id("btn")).submit();

        Thread.sleep(2000);
        assertEquals("Registered Successfully", driver.switchTo().alert().getText());
    }

    @Test
    void testSignupfail() throws InterruptedException {
        // WebDriver driver = new ChromeDriver();

        driver.get("http://localhost:3000/");
        driver.manage().window().maximize();
        driver.findElement(By.linkText("SignUp")).click();

        driver.findElement(By.id("uname")).sendKeys("Ravi");

        driver.findElement(By.id("email")).sendKeys("ravi@gmail.com");

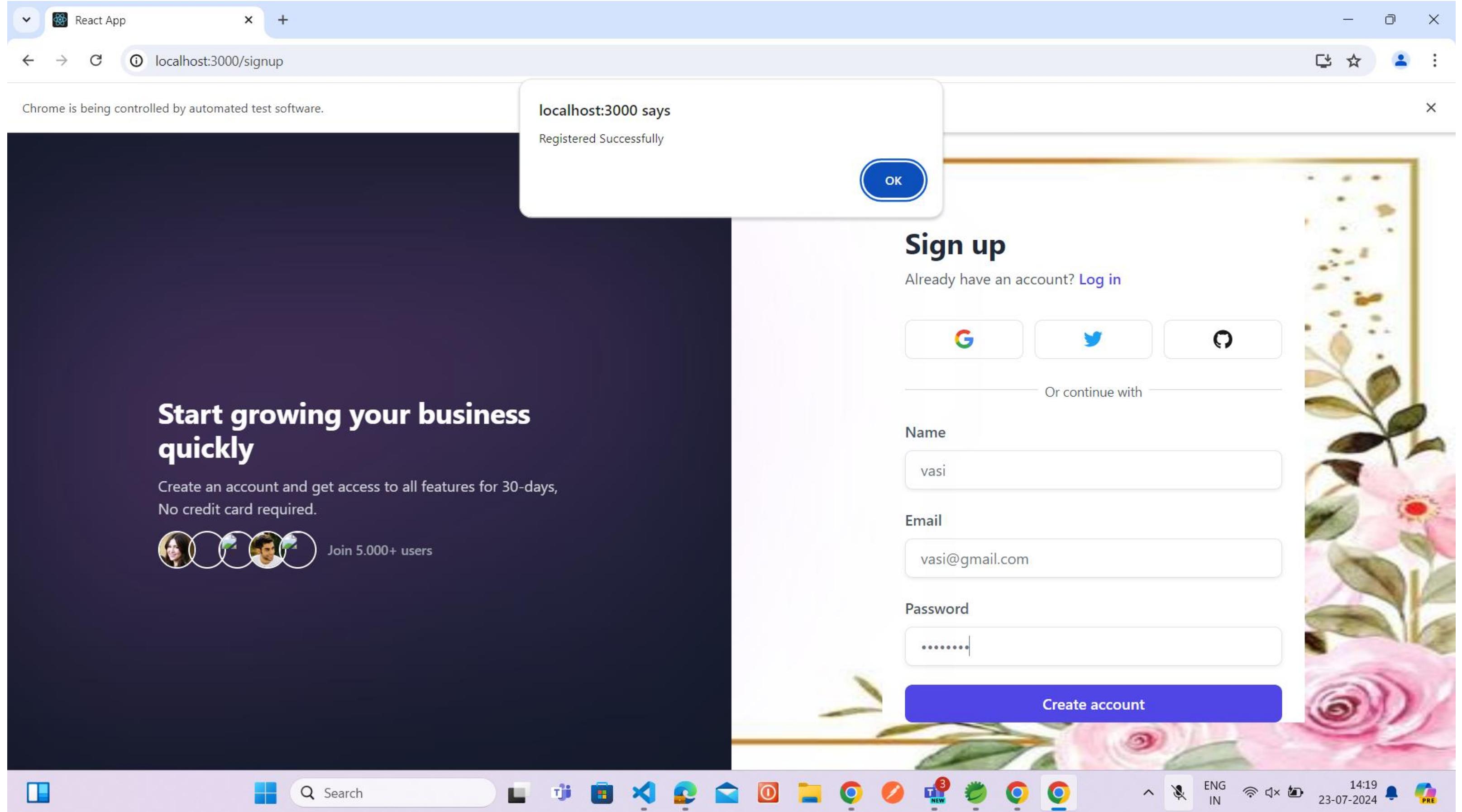
        driver.findElement(By.id("pwd")).sendKeys(" ");

        driver.findElement(By.id("btn")).submit();

        Thread.sleep(2000);

        assertEquals("Password Cannot be empty value !! It Should Contain atleast 8 characters!!!",
            driver.switchTo().alert().getText());
        Thread.sleep(10000);
    }
}
```

SCREENSHOTS OF TESTCASE



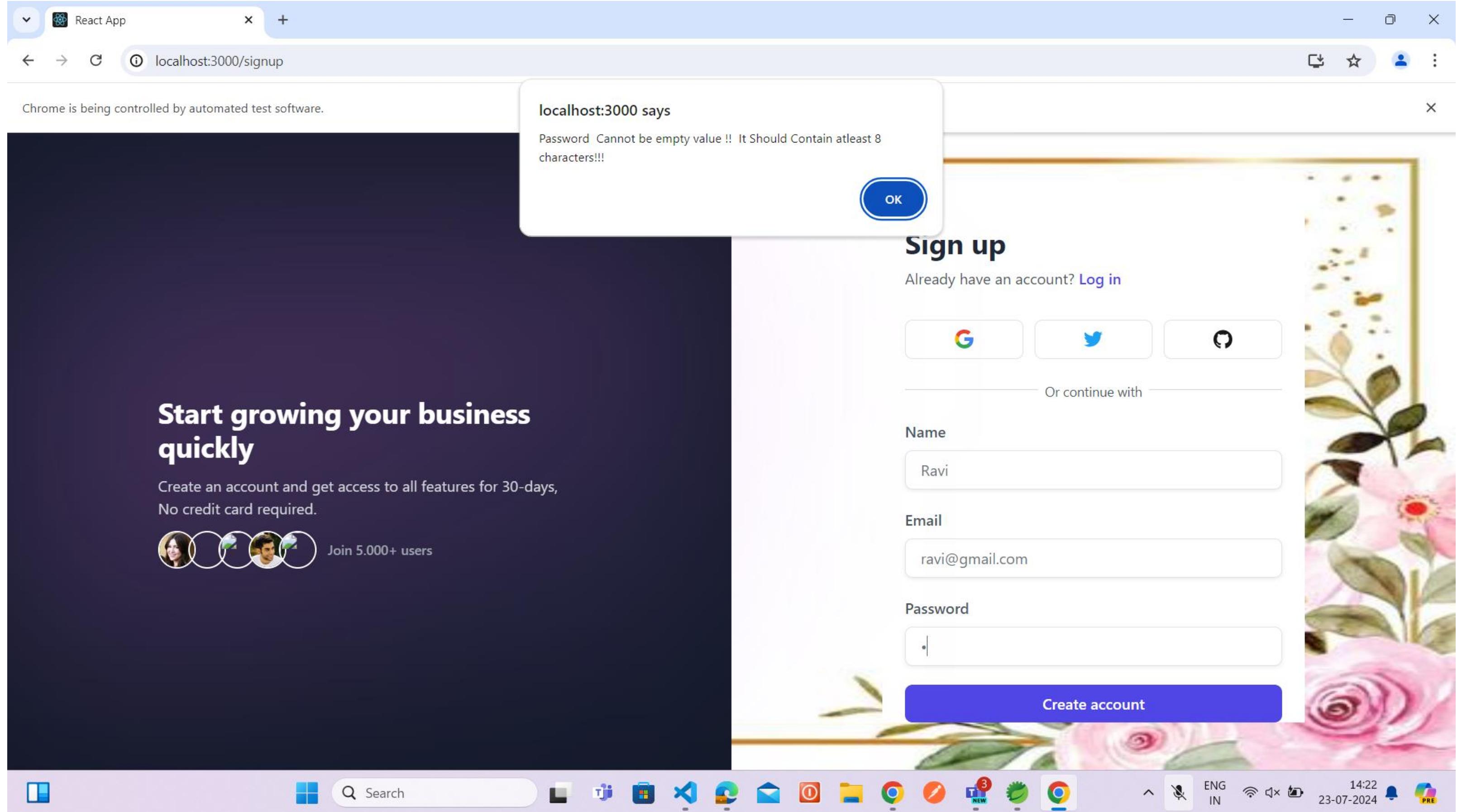
Runs: 1/1 Errors: 0 Failures: 0

AppTest2 [Runner: JUnit 5] (9.942 s)

testSignup() (9.942 s)

```
40
41
42
43
44 @AfterEach
45
46 public void tearDown() {
47
48     driver.quit();
49 }
50
51 @Test
52 @Order(1)
53 void testSignup() throws InterruptedException {
54     WebDriver driver = new ChromeDriver();
55     driver.get("http://localhost:3000/");
56     driver.manage().window().maximize();
57     driver.findElement(By.linkText("SignUp")).click();
58
59     driver.findElement(By.id("uname")).sendKeys("vasi");
60
61     driver.findElement(By.id("email")).sendKeys("vasi@gmail.com");
62
63     driver.findElement(By.id("pwd")).sendKeys("vasi@123");
64
65     driver.findElement(By.id("btn")).submit();
66 }
```

Failure Trace



```
FINISHED IN 17.251 SECONDS
Runs: 1/1      Errors: 0      Failures: 0
[{"id": 1, "label": "SignUpTest [Runner: JUnit 5] (17.251 s)", "type": "suite"}, {"id": 2, "label": "testSignupfail() (17.251 s)", "type": "method"}]
void testSignupfail() throws InterruptedException {
    WebDriver driver = new ChromeDriver();
    driver.get("http://localhost:3000/");
    driver.manage().window().maximize();
    driver.findElement(By.linkText("SignUp")).click();
    driver.findElement(By.id("uname")).sendKeys("Ravi");
    driver.findElement(By.id("email")).sendKeys("ravi@gmail.com");
    driver.findElement(By.id("pwd")).sendKeys(" ");
    driver.findElement(By.id("btn")).submit();
    Thread.sleep(2000);
    assertEquals("Password Cannot be empty value !! It Should Contain atleast 8 characters", driver.switchTo().alert().getText());
    Thread.sleep(10000);
}
```


LOGIN

```
class LoginTestCase {

    WebDriver driver;
    @BeforeEach
    public void setUp() {
        driver = new ChromeDriver();
    }

    @AfterEach
    public void tearDown() {
        driver.quit();
    }
    @Test

    void testlogin() {
        WebDriver driver = new ChromeDriver();

        driver.get("http://localhost:3000/");
        driver.manage().window().setSize(new Dimension(1552, 832));
        driver.findElement(By.linkText("Login")).click();
        driver.findElement(By.name("email")).click();
        driver.findElement(By.name("email")).sendKeys("raj@gmail.com");
        driver.findElement(By.name("password")).click();
        driver.findElement(By.name("password")).sendKeys("Raja@123");
        driver.findElement(By.cssSelector(".text-white")).click();
        assertEquals("Login Success!!", driver.switchTo().alert().getText());
    }

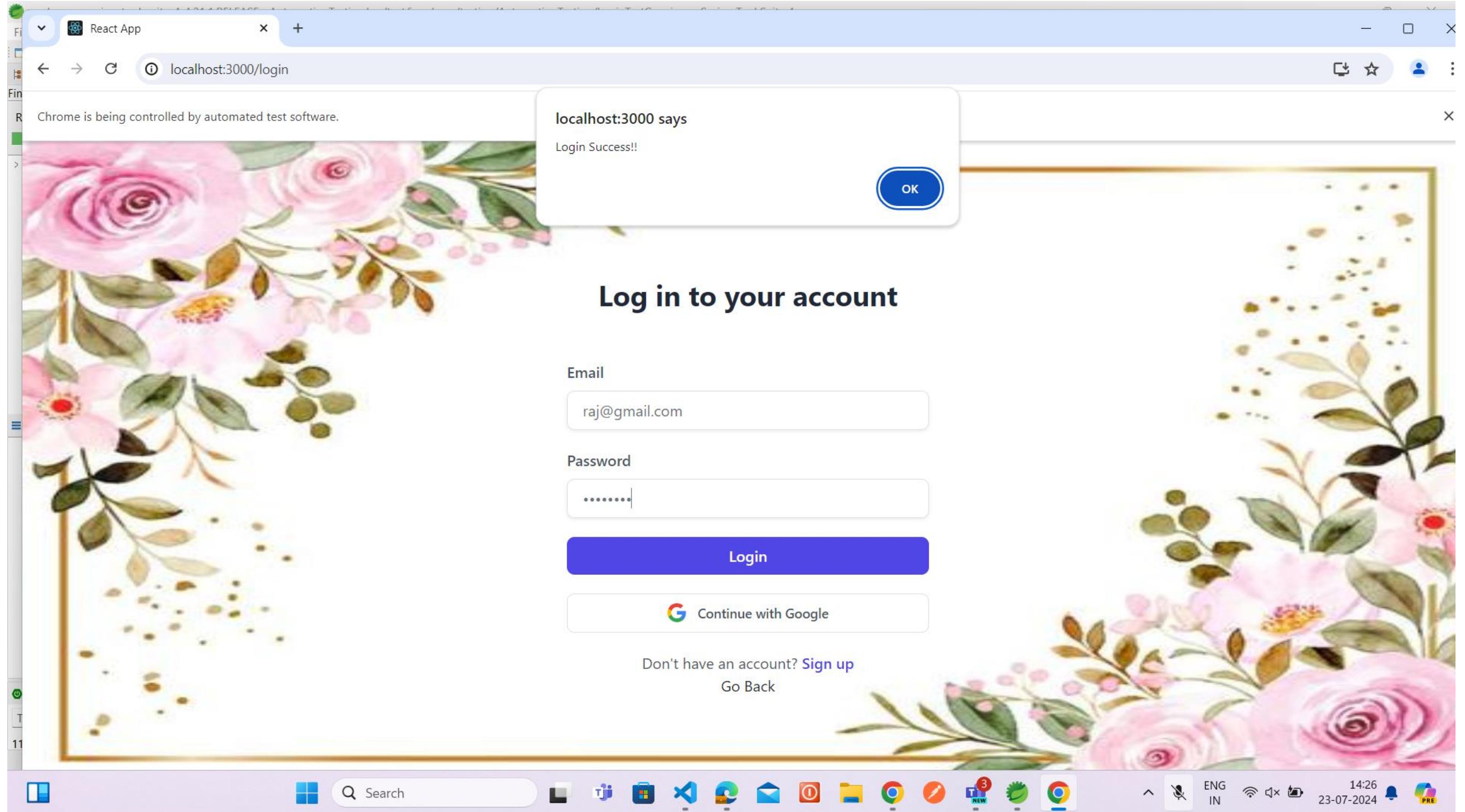
    @Test

    void testLoginWithInvalidCredentials() throws InterruptedException {

        driver.get("http://localhost:3000/");
        driver.manage().window().setSize(new Dimension(1552, 832));
        driver.findElement(By.linkText("Login")).click();
        driver.findElement(By.name("email")).click();
        driver.findElement(By.name("email")).sendKeys("raju@gmail.com");
        driver.findElement(By.name("password")).click();
        driver.findElement(By.name("password")).sendKeys("Raju@123");
        driver.findElement(By.cssSelector(".text-white")).click();

        assertEquals("Invalid credentials. Please try again.", driver.switchTo().alert().getText());
        Thread.sleep(10000);
    }
}
```

SCREENSHOTS



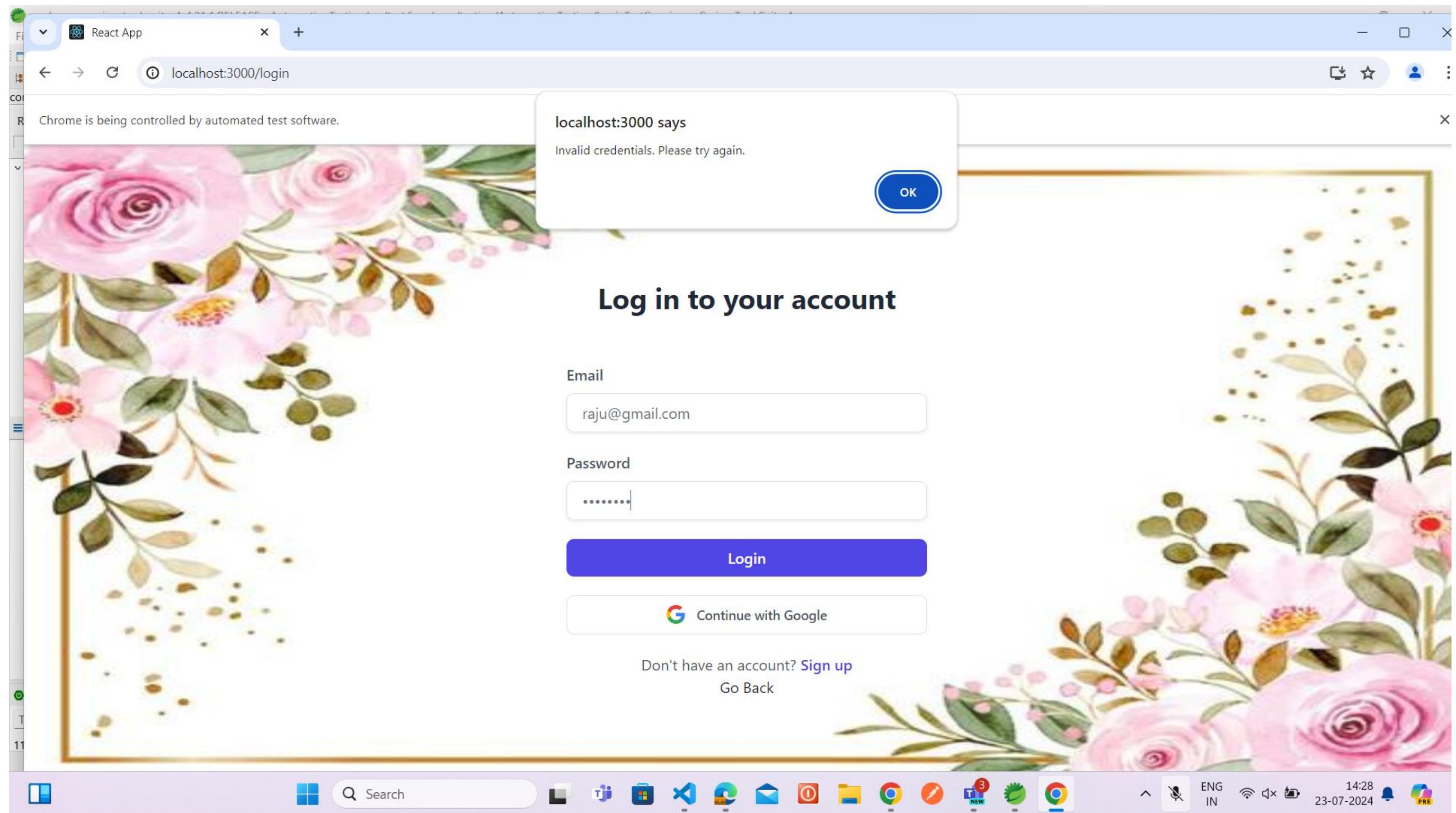
Runs: 1/1 Errors: 0 Failures: 0

>LoginTestCase [Runner: JUnit 5] (14.990 s)

testlogin() (14.990 s)

```
20  @AfterEach
21  public void tearDown() {
22      driver.quit();
23  }
24  @Test
25
26  void testlogin() {
27      WebDriver driver = new ChromeDriver();
28
29      driver.get("http://localhost:3000/");
30      driver.manage().window().setSize(new Dimension(1552, 832));
31      driver.findElement(By.linkText("Login")).click();
32      driver.findElement(By.name("email")).click();
33      driver.findElement(By.name("email")).sendKeys("raj@gmail.com");
34      driver.findElement(By.name("password")).click();
35      driver.findElement(By.name("password")).sendKeys("Raja@123");
36      driver.findElement(By.cssSelector(".text-white")).click();
37      assertEquals("Login Success!!", driver.switchTo().alert().getText());
38
39
40
41
42 }
```

FAILURE CASE



Runs: 1/1 Errors: 0 Failures: 0

>LoginTestCase [Runner: JUnit 5] (15.723 s)

testLoginWithInvalidCredentials() (15.723 s)

```
40
41
42 }
43
44 @Test
45
46 void testLoginWithInvalidCredentials() throws InterruptedException {
47
48
49     driver.get("http://localhost:3000/");
50     driver.manage().window().setSize(new Dimension(1552, 832));
51     driver.findElement(By.linkText("Login")).click();
52     driver.findElement(By.name("email")).click();
53     driver.findElement(By.name("email")).sendKeys("raju@gmail.com");
54     driver.findElement(By.name("password")).click();
55     driver.findElement(By.name("password")).sendKeys("Raju@123");
56     driver.findElement(By.cssSelector(".text-white")).click();
57
58     assertEquals("Invalid credentials. Please try again.", driver.switchTo().alert().getText());
59     Thread.sleep(10000);
60
61 }
62
63 }
```

ADD USER

CODE

```
class AddUserTC {  
    WebDriver driver;  
  
    @BeforeEach  
    public void setUp() {  
        driver = new ChromeDriver();  
    }  
}
```

```

@AfterEach
public void tearDown() {
    driver.quit();
}
@Test
void ValidateAddUser() throws InterruptedException {

    driver.get("http://localhost:3000/");
    driver.manage().window().maximize();
    driver.findElement(By.linkText("Features")).click();
    driver.findElement(By.linkText("Our Customers")).click();
    driver.findElement(By.linkText("Add +")).click();
    driver.findElement(By.name("uname")).sendKeys("poornima");
    driver.findElement(By.name("actype")).sendKeys("Bride");
    driver.findElement(By.name("addr")).sendKeys("kaalvasal,Madurai-08");
    driver.findElement(By.name("age")).sendKeys("25");

    driver.findElement(By.name("sal")).sendKeys("15000");

    driver.findElement(By.name("job")).sendKeys("Tailor");
    driver.findElement(By.name("pay")).sendKeys("Monthly");

    driver.findElement(By.name("religion")).sendKeys("Hindu");

    driver.findElement(By.name("sts")).sendKeys("Single");
    driver.findElement(By.xpath("//option[. = '4']")).click();
    driver.findElement(By.id("submit")).submit();
    Thread.sleep(2000);
    assertEquals("Data added Successfully", driver.switchTo().alert().getText());
    // driver.switchTo().alert().accept();
    Thread.sleep(10000);
}

```

SCREENSHOTS :



```

30   driver.findElement(By.linkText("Features")).click();
31   driver.findElement(By.linkText("Our Customers")).click();
32   driver.findElement(By.linkText("Add +")).click();
33   driver.findElement(By.name("uname")).sendKeys("Dharshini");
34   driver.findElement(By.name("actype")).sendKeys("Bride");
35   driver.findElement(By.name("addr")).sendKeys("Therkuvusal,Madurai-08");
36   driver.findElement(By.name("age")).sendKeys("25");

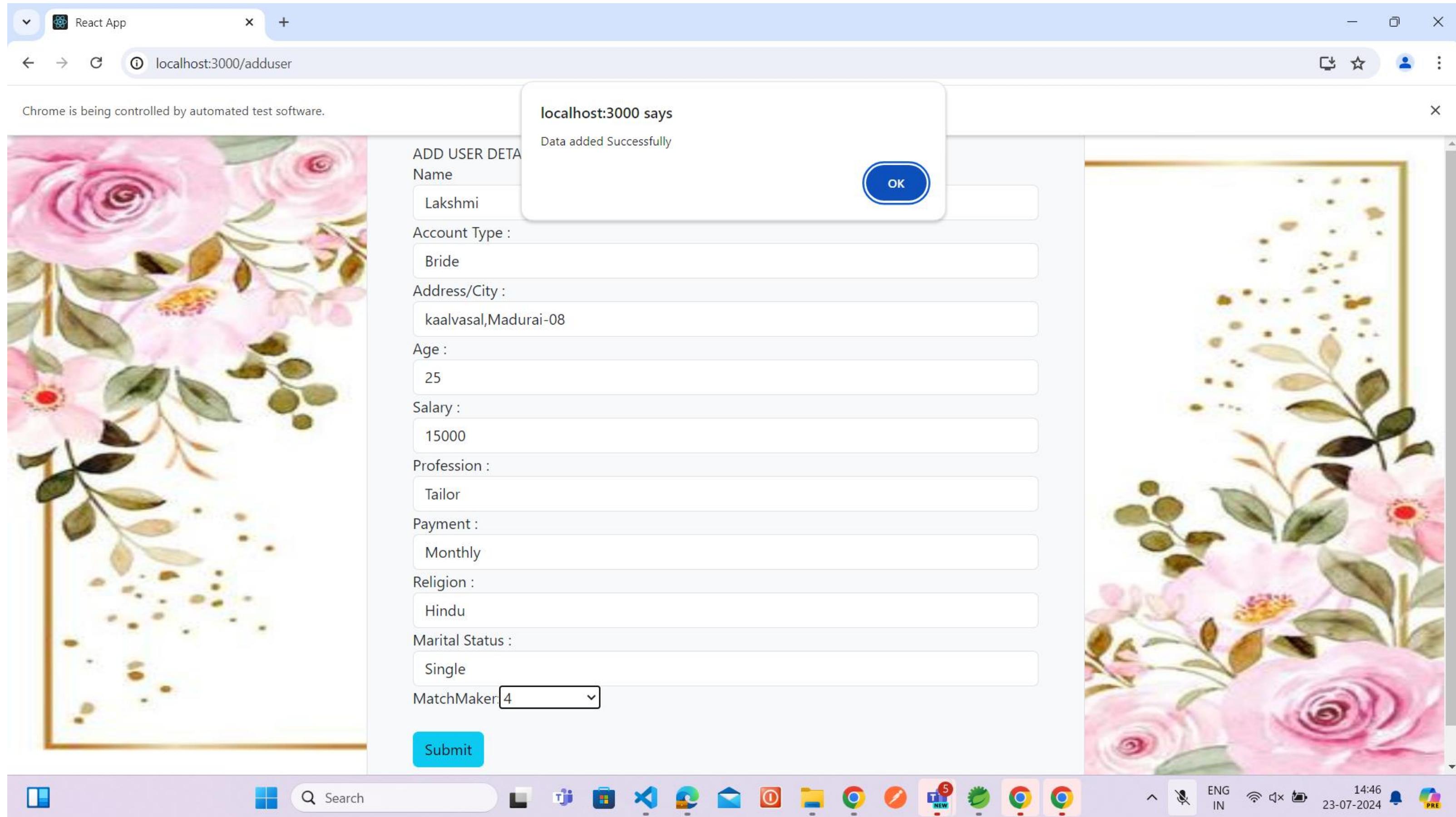
37   driver.findElement(By.name("sal")).sendKeys("25000");

38   driver.findElement(By.name("job")).sendKeys("Trainee");
39   driver.findElement(By.name("pay")).sendKeys("Monthly");

40   driver.findElement(By.name("religion")).sendKeys("Hindu");

41   driver.findElement(By.name("sts")).sendKeys("Single");
42   driver.findElement(By.xpath("//option[. = '9']")).click();
43   driver.findElement(By.id("submit")).submit();
44   Thread.sleep(2000);
45   assertEquals("Data added Successfully", driver.switchTo().alert().getTe,
46   driver.switchTo().alert().accept();
47   Thread.sleep(10000);
48
49
50
51

```



```
@Test
void testInvalidAddUserOrNull() throws InterruptedException {

    driver.get("http://localhost:3000/");
    driver.manage().window().maximize();
    driver.findElement(By.linkText("Features")).click();
    driver.findElement(By.linkText("Our Customers")).click();
    driver.findElement(By.linkText("Add +")).click();
    driver.findElement(By.name("uname")).sendKeys("ponni");
    driver.findElement(By.name("actype")).sendKeys("Bride");
    driver.findElement(By.name("addr")).sendKeys("keelavasal,Madurai-08");
    driver.findElement(By.name("age")).sendKeys("16");

    driver.findElement(By.name("sal")).sendKeys("15000");

    driver.findElement(By.name("job")).sendKeys("Tailor");
    driver.findElement(By.name("pay")).sendKeys("Monthly");

    driver.findElement(By.name("religion")).sendKeys("Hindu");
```

```

        driver.findElement(By.name("sts")).sendKeys("Single");
        driver.findElement(By.xpath("//option[. = '4']")).click();
        driver.findElement(By.id("submit")).submit();
        Thread.sleep(2000);
        assertEquals("Minimum Age is 18 !!! Please Enter the Valid Inputs!!!", driver.switchTo().alert().getText());
        driver.switchTo().alert().accept();
        Thread.sleep(10000);
    }
}

```

SCREENSHOTS

Runs: 1/1 Errors: 0 Failures: 0

AddUserTC [Runner: JUnit 5] (35.664 s)

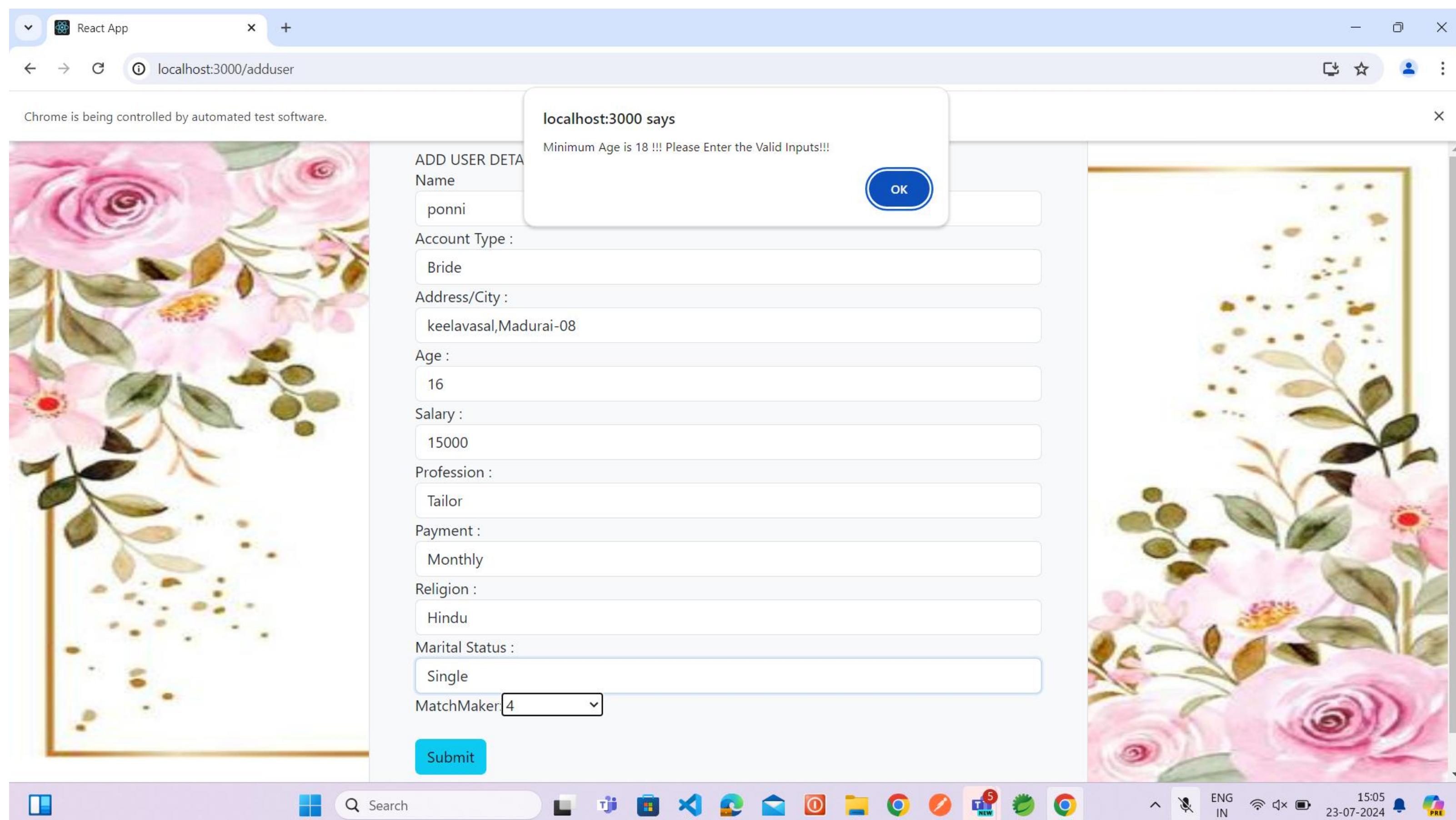
testInvalidAddUserOrNull() (35.664 s)

Failure Trace

```

56
57     void testInvalidAddUserOrNull() throws InterruptedException {
58
59
60         driver.get("http://localhost:3000/");
61         driver.manage().window().maximize();
62         driver.findElement(By.linkText("Features")).click();
63         driver.findElement(By.linkText("Our Customers")).click();
64         driver.findElement(By.linkText("Add +")).click();
65         driver.findElement(By.name("uname")).sendKeys("ponni");
66         driver.findElement(By.name("actype")).sendKeys("Bride");
67         driver.findElement(By.name("addr")).sendKeys("keelavasal,Madurai-08");
68         driver.findElement(By.name("age")).sendKeys("16");
69
70         driver.findElement(By.name("sal")).sendKeys("15000");
71
72         driver.findElement(By.name("job")).sendKeys("Tailor");
73         driver.findElement(By.name("pay")).sendKeys("Monthly");
74
75         driver.findElement(By.name("religion")).sendKeys("Hindu");
76
77         driver.findElement(By.name("sts")).sendKeys("Single");
78         driver.findElement(By.xpath("//option[. = '4']")).click();
79         driver.findElement(By.id("submit")).submit();

```



```
void testInvalidAddUserDataOrNull() throws InterruptedException {

    driver.get("http://localhost:3000/");
    driver.manage().window().maximize();
    driver.findElement(By.linkText("Features")).click();
    driver.findElement(By.linkText("Our Customers")).click();
    driver.findElement(By.linkText("Add +")).click();
    driver.findElement(By.name("uname")).sendKeys(" ");
    driver.findElement(By.name("actype")).sendKeys("Bride");
    driver.findElement(By.name("addr")).sendKeys("keelavasal,Madurai-08");
    driver.findElement(By.name("age")).sendKeys("16");

    driver.findElement(By.name("sal")).sendKeys("15000");

    driver.findElement(By.name("job")).sendKeys("Tailor");
    driver.findElement(By.name("pay")).sendKeys("Monthly");

    driver.findElement(By.name("religion")).sendKeys("Hindu");

    driver.findElement(By.name("sts")).sendKeys("Single");
    driver.findElement(By.xpath("//option[. = '4']")).click();
    driver.findElement(By.id("submit")).submit();
    Thread.sleep(2000);
    assertEquals("Please enter Name !!! Please Enter the Valid Inputs!!!", driver.switchTo().alert().getText());
    driver.switchTo().alert().accept();
    Thread.sleep(10000);
}
```

Finished after 23.609 seconds

Runs: 1/1 Errors: 0 Failures: 0

AddUserTC [Runner: JUnit 5] (23.471 s)

testInvalidAddUserDataOrNull() (23.471 s)

```

96     driver.findElement(By.linkText("Features")).click();
97     driver.findElement(By.linkText("Our Customers")).click();
98     driver.findElement(By.linkText("Add +")).click();
99     driver.findElement(By.name("uname")).sendKeys(" ");
100    driver.findElement(By.name("actype")).sendKeys("Bride");
101    driver.findElement(By.name("addr")).sendKeys("keelavasal,Madurai-08");
102    driver.findElement(By.name("age")).sendKeys("16");
103
104    driver.findElement(By.name("sal")).sendKeys("15000");
105
106    driver.findElement(By.name("job")).sendKeys("Tailor");
107    driver.findElement(By.name("pay")).sendKeys("Monthly");
108
109    driver.findElement(By.name("religion")).sendKeys("Hindu");
110
111    driver.findElement(By.name("sts")).sendKeys("Single");
112    driver.findElement(By.xpath("//option[. = '4']")).click();
113    driver.findElement(By.id("submit")).submit();
114    Thread.sleep(2000);
115    assertEquals("Please enter Name !!! Please Enter the Valid Inputs!!!", driver.switchTo().alert().accept());
116    Thread.sleep(10000);
117
118 }
119
120 }
121 }
```

Failure Trace

React App localhost:3000/adduser

Chrome is being controlled by automated test software.

localhost:3000 says

Please enter Name !!! Please Enter the Valid Inputs!!!

OK

15:08 23-07-2024 ENG IN

ADD MATCH MAKER

```

class AddMatchMAkerTC {
    WebDriver driver;

    @BeforeEach
    public void setUp() {
        driver = new ChromeDriver();
    }

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

```
@Test
void testAddMatchMaker() throws InterruptedException {

    driver.get("http://localhost:3000/");
    driver.manage().window().maximize();
    driver.findElement(By.linkText("Features")).click();
    driver.findElement(By.linkText("Match Makers")).click();
    driver.findElement(By.id("add")).click();
    driver.findElement(By.name("mname")).click();
    driver.findElement(By.name("mname")).sendKeys("Ramana");
    driver.findElement(By.name("addr")).click();
    driver.findElement(By.name("addr")).sendKeys("Sinthamani, Madurai-609906");
    driver.findElement(By.name("cmobile")).click();
    driver.findElement(By.name("cmobile")).sendKeys("8912345678");

    driver.findElement(By.id("submit")).submit();
    Thread.sleep(2000);
    assertEquals("Data added Successfully", driver.switchTo().alert().getText());
    Thread.sleep(6000);
}

@Test
void testAddMatchMakerIfInvalidOrNULL() throws InterruptedException {

    driver.get("http://localhost:3000/");
    driver.manage().window().maximize();
    driver.findElement(By.linkText("Features")).click();
    driver.findElement(By.linkText("Match Makers")).click();
    driver.findElement(By.id("add")).click();
    driver.findElement(By.name("mname")).click();
    driver.findElement(By.name("mname")).sendKeys("");
    driver.findElement(By.name("addr")).click();
    driver.findElement(By.name("addr")).sendKeys("Sinthamani, Madurai-609906");
    driver.findElement(By.name("cmobile")).click();
    driver.findElement(By.name("cmobile")).sendKeys("8912345678");

    driver.findElement(By.id("submit")).submit();
    Thread.sleep(2000);
    assertEquals("Name Cannot be empty value!!! Please Enter Name", driver.switchTo().alert().getText());
    Thread.sleep(6000);
}
}
```

Finished after 19.005 seconds

Runs: 1/1 Errors: 0 Failures: 0

+

AddMatchMAkerTC [Runner: JUnit 5] (19.715 s)

testAddMatchMaker() (19.715 s)

```
20@AfterEach
21 public void tearDown() {
22     driver.quit();
23 }
24 @Test
25 void testAddMatchMaker() throws InterruptedException {
26
27     driver.get("http://localhost:3000/");
28     driver.manage().window().maximize();
29     driver.findElement(By.linkText("Features")).click();
30     driver.findElement(By.linkText("Match Makers")).click();
31     driver.findElement(By.id("add")).click();
32     driver.findElement(By.name("mname")).click();
33     driver.findElement(By.name("mname")).sendKeys("Ramana");
34     driver.findElement(By.name("addr")).click();
35     driver.findElement(By.name("addr")).sendKeys("Sinthamani,Madurai-609906");
36 }
```

React App

localhost:3000/insert

Chrome is being controlled by automated test software.

localhost:3000 says

Data added Successfully

OK

ADD MATCHMAKER DATA

Name : Ramana

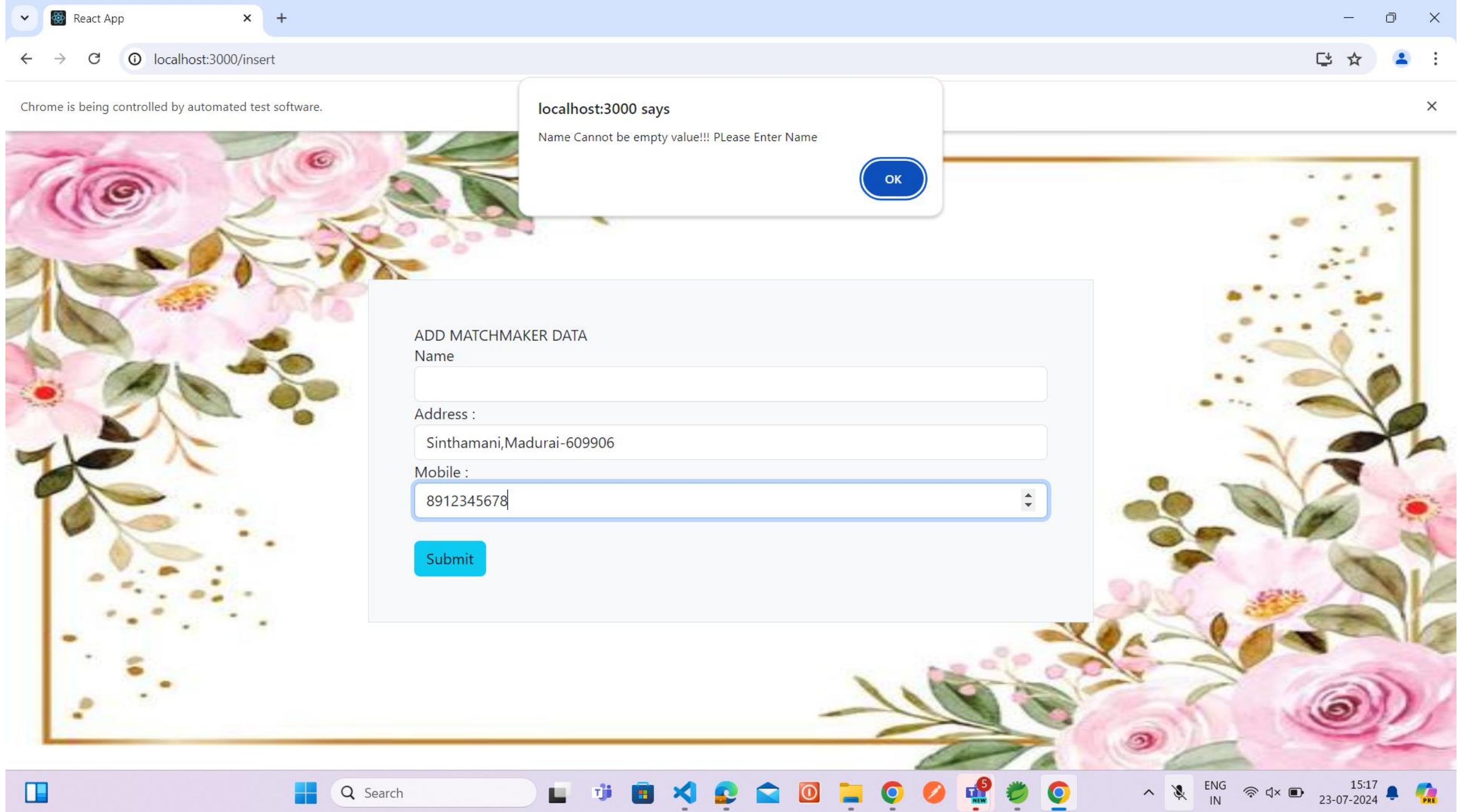
Address : Sinthamani,Madurai-609906

Mobile : 8912345678

Submit

Search

15:12 23-07-2024 ENG IN



Runs: 1/1 Errors: 0 Failures: 0

```

46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67

```

AddMatchMAkerTC [Runner: JUnit 5] (23.380 s)

testAddMatchMakerIfInvalidOrNULL() (23.380 s)

```

@Test
void testAddMatchMakerIfInvalidOrNULL() throws InterruptedException {

    driver.get("http://localhost:3000/");
    driver.manage().window().maximize();
    driver.findElement(By.linkText("Features")).click();
    driver.findElement(By.linkText("Match Makers")).click();
    driver.findElement(By.id("add")).click();
    driver.findElement(By.name("mname")).click();
    driver.findElement(By.name("mname")).sendKeys("");
    driver.findElement(By.name("addr")).click();
    driver.findElement(By.name("addr")).sendKeys("Sinthamani,Madurai-609906");
    driver.findElement(By.name("cmobile")).click();
    driver.findElement(By.name("cmobile")).sendKeys("8912345678");

    driver.findElement(By.id("submit")).submit();
    Thread.sleep(2000);
    assertEquals("Name Cannot be empty value!!! Please Enter Name", driver.switchTo().alert.getText());
    Thread.sleep(6000);
}

```

Failure Trace

EDIT MATCH MAKER DETAILS :

```

class EditMatchMAker {

    WebDriver driver;
    @BeforeEach
    public void setUp() {
        driver = new ChromeDriver();
    }

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

    @Test
    void testEditMatchMaker() throws InterruptedException {

```

```

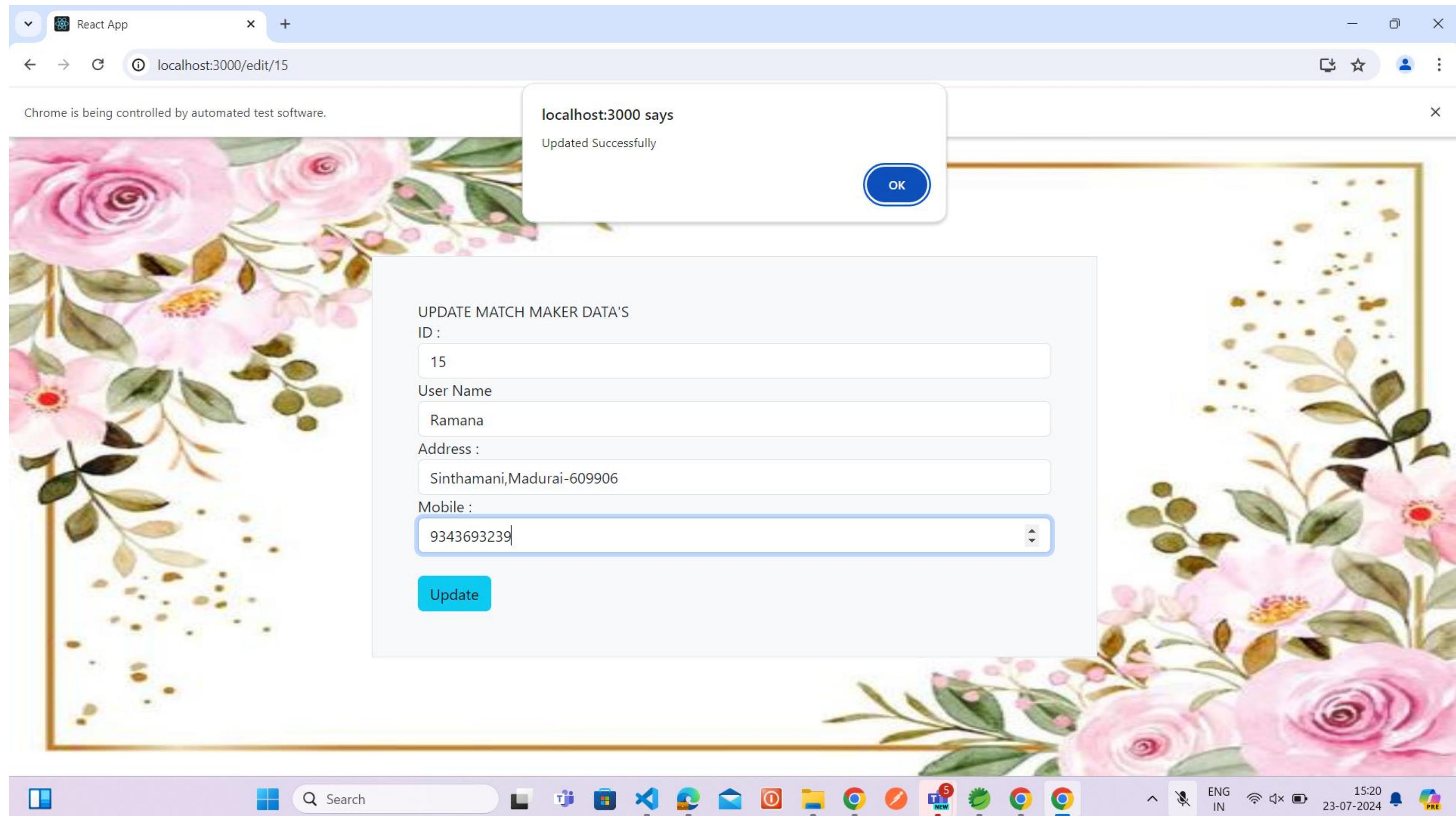
WebDriver driver = new ChromeDriver();
driver.get("http://localhost:3000/");
driver.manage().window().maximize();

driver.get("http://localhost:3000/view");
driver.findElement(By.name("update15")).click();
driver.findElement(By.name("cmobile")).clear();
driver.findElement(By.name("cmobile")).sendKeys("9343693239");

// driver.findElement(By.id("submit")).submit();
//// Thread.sleep(2000);
assertEquals("Updated Successfully", driver.switchTo().alert().getText());
}

}

```



EditMatchMAker [Runner: JUnit 5] (12.753 s)

testEditMatchMaker() (12.753 s)

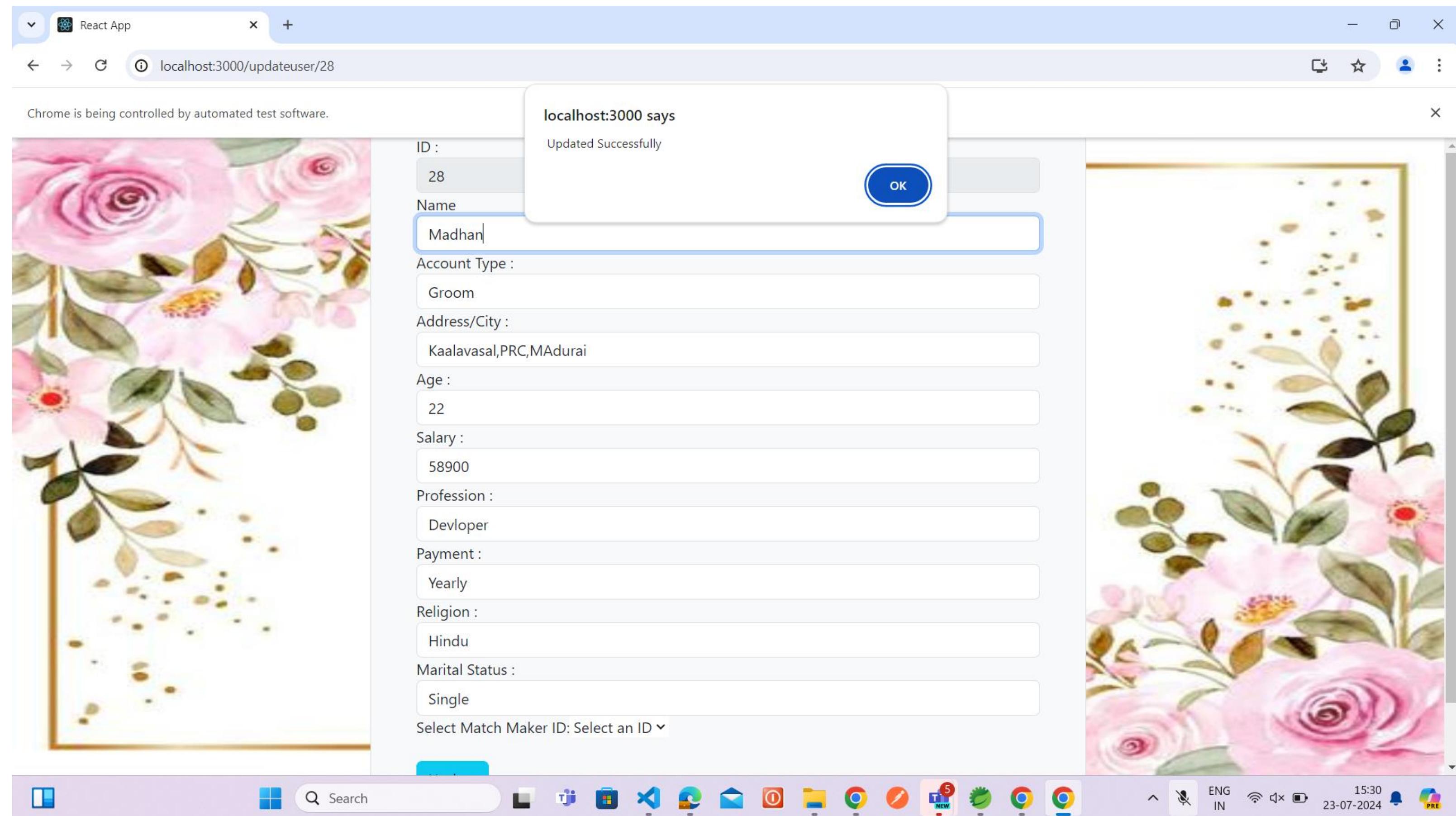
```

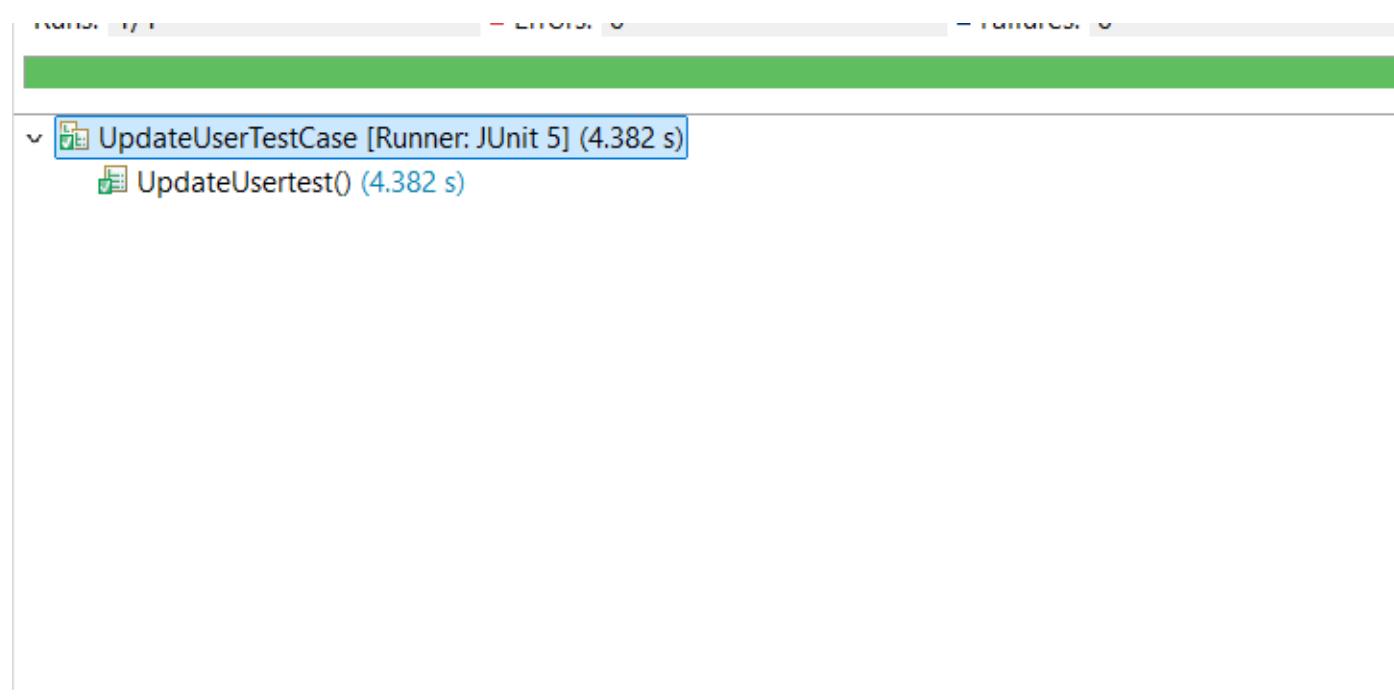
8 import org.openqa.selenium.By;
9 import org.openqa.selenium.WebDriver;
10 import org.openqa.selenium.chrome.ChromeDriver;
11
12 class EditMatchMAker {
13
14     WebDriver driver;
15     @BeforeEach
16     public void setUp() {
17         driver = new ChromeDriver();
18     }
19
20     @AfterEach
21     public void tearDown() {
22         driver.quit();
23     }
24
25
26     @Test
27     void testEditMatchMaker() throws InterruptedException {
28
29         WebDriver driver = new ChromeDriver();
30         driver.get("http://localhost:3000/");

```

EDIT USER :

```
class UpdateUserTestCase {  
  
    WebDriver driver = new ChromeDriver();  
    @After  
    public void tearDown() {  
        driver.quit();  
    }  
  
    @Test  
    void UpdateUserTest() throws InterruptedException {  
        driver.manage().window().maximize();  
        driver.get("http://localhost:3000/viewuser");  
        driver.findElement(By.name("update28")).click();  
        driver.findElement(By.name("uname")).clear();  
        driver.findElement(By.name("uname")).sendKeys("Madhan");  
        driver.findElement(By.id("submit")).submit();  
        Thread.sleep(2000);  
        assertEquals(" Updated Successfully", driver.switchTo().alert().getText());  
    }  
}
```





```
9 import org.openqa.selenium.chrome.ChromeDriver;
10
11 class UpdateUserTestCase {
12
13     WebDriver driver = new ChromeDriver();
14     @After
15     public void tearDown() {
16         driver.quit();
17     }
18
19     @Test
20     void UpdateUsertest() throws InterruptedException {
21         driver.manage().window().maximize();
22         driver.get("http://localhost:3000/viewuser");
23         driver.findElement(By.name("update28")).click();
24         driver.findElement(By.name("uname")).clear();
25         driver.findElement(By.name("uname")).sendKeys("Madhan");
26         driver.findElement(By.id("submit")).submit();

```

DELETE MATCH MAKER

```
import org.junit.After;
import org.junit.Before;
import org.junit.jupiter.api.Test;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.support.ui.ExpectedConditions;
import org.openqa.selenium.support.ui.WebDriverWait;

class DeleteMatchMaker {

    WebDriver driver = new ChromeDriver();
    @After
    public void tearDown() {
        driver.quit();
    }

    @Test
    void Deletetest() throws InterruptedException {
        driver.get("http://localhost:3000/view");
        driver.manage().window().maximize();

        driver.findElement(By.name("delete15")).click();
        Thread.sleep(2000);
        driver.switchTo().alert().accept();

        WebDriverWait waitDriver = new WebDriverWait(driver, Duration.ofSeconds(5));
        waitDriver.until(ExpectedConditions.alertIsPresent());

        assertEquals("record has deleted", driver.switchTo().alert().getText());
    }
}
```

React App

localhost:3000/view

Chrome is being controlled by automated test software.

localhost:3000 says
record has deleted

OK

Add +

ID	Name	Address	Contact detail	Action
1	Viswanathan	KamarajSaalai,Madurai-10	8907221346	<button>Update</button> <button>Delete</button>
4	Sargunam	Madurai-TNHBColony	92900149165	<button>Update</button> <button>Delete</button>
6	SubramaniyanKumaran	Gandhipuram,Coimbatore	8899001123	<button>Update</button> <button>Delete</button>
9	Nirmala	Therkuvusal,Madurai-625008	9975512678	<button>Update</button> <button>Delete</button>
15	Ramana	Sinthamani,Madurai-609906	9343693239	<button>Update</button> <button>Delete</button>

React App

localhost:3000/view

Chrome is being controlled by automated test software.

MATCH MAKER DETAILS

You can open bookmarks, reading mode, and more from the Chrome menu

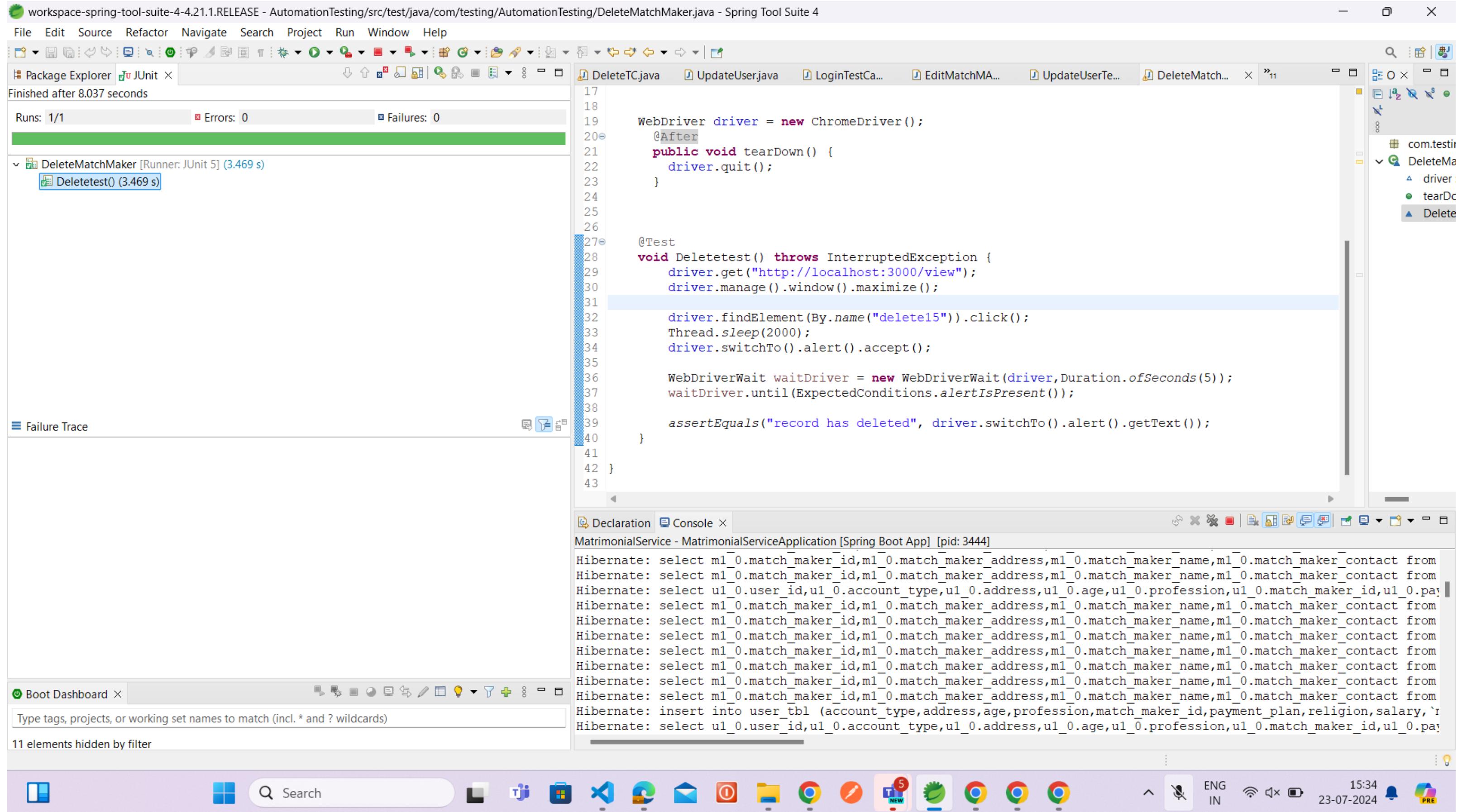
SearchId

Add +

ID	Name	Address	Contact detail	Action
1	Viswanathan	KamarajSaalai,Madurai-10	8907221346	<button>Update</button> <button>Delete</button>
4	Sargunam	Madurai-TNHBColony	92900149165	<button>Update</button> <button>Delete</button>
6	SubramaniyanKumaran	Gandhipuram,Coimbatore	8899001123	<button>Update</button> <button>Delete</button>
9	Nirmala	Therkuvusal,Madurai-625008	9975512678	<button>Update</button> <button>Delete</button>

Snipping Tool

Screenshot copied to clipboard and saved
Select here to mark up and share.



DELETE USER

```
WebDriver driver;
@BeforeEach
public void setUp() {
    driver = new ChromeDriver();
}

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

@Test
void DeleteUser() throws InterruptedException {
    driver.get("http://localhost:3000/viewuser");
    driver.manage().window().maximize();

    driver.findElement(By.name("delete35")).click();
    Thread.sleep(5000);
    driver.switchTo().alert().accept();

    WebDriverWait waitDriver = new WebDriverWait(driver, Duration.ofSeconds(5));
    waitDriver.until(ExpectedConditions.alertIsPresent());
    Thread.sleep(6000);
    assertEquals("record has deleted", driver.switchTo().alert().getText());
}
```

React App

localhost:3000/viewuser

Chrome is being controlled by automated test software.

localhost:3000 says
Do you want to delete

OK Cancel

USER ID	User Name	Age	Address	Religion	Profession	Salary	Account Type	Payment Method	Marital Status	Match Maker Id	Action
1	Karthi	29	Villapuram,Madurai-10	Hindu	RealEstate	25000	Groom	Premium	Single	1	<button>Update</button> <button>Delete</button>
3	Prabhusaran	26	Perungudi,Madurai-22	Hindus	IT	70000	Groom	Yearly	Single	1	<button>Update</button> <button>Delete</button>
11	Nagarjune	25	Munichalai,Madurai	Hindu	Devloper	80000	Groom	Monthly	Single	4	<button>Update</button> <button>Delete</button>
19	NandhaKumaran	40	Hindu	20000	IT-Tester	Monthly	Groom	No-9,NanjuPuram,Virudhunagar	Unmarried	6	<button>Update</button> <button>Delete</button>
22	Vijay	45	Chennai	Christian	Actor	2000000	Groom	Premium	Married	9	<button>Update</button> <button>Delete</button>
26	Seetha	25	Therkuvusal,Madurai-08	Hindu	Teacher	25000	Bride	Monthly	Single	9	<button>Update</button> <button>Delete</button>
28	Madhan	22	Kaalavasal,PRC,MAdurai	Hindu	Devloper	58900	Groom	Yearly	Single	9	<button>Update</button> <button>Delete</button>
31	YuvaSree	25	Therkuvusal,Madurai-08	Hindu	Teacher	25000	Bride	Monthly	Single	9	<button>Update</button> <button>Delete</button>
33	Dharshini	25	Therkuvusal,Madurai-08	Hindu	Trainee	25000	Bride	Monthly	Single	9	<button>Update</button> <button>Delete</button>
35	Lakshmi	25	kaalvasal,Madurai-08	Hindu	Tailor	15000	Bride	Monthly	Single	4	<button>Update</button> <button>Delete</button>

SearchName

Add +

15:41 23-07-2024 ENG IN

React App

localhost:3000/viewuser

Chrome is being controlled by automated test software.

localhost:3000 says
record has deleted

OK

USER ID	User Name	Age	Address	Religion	Profession	Salary	Account Type	Payment Method	Marital Status	Match Maker Id	Action
1	Karthi	29	Villapuram,Madurai-10	Hindu	RealEstate	25000	Groom	Premium	Single	1	<button>Update</button> <button>Delete</button>
3	Prabhusaran	26	Perungudi,Madurai-22	Hindus	IT	70000	Groom	Yearly	Single	1	<button>Update</button> <button>Delete</button>
11	Nagarjune	25	Munichalai,Madurai	Hindu	Devloper	80000	Groom	Monthly	Single	4	<button>Update</button> <button>Delete</button>
19	NandhaKumaran	40	Hindu	20000	IT-Tester	Monthly	Groom	No-9,NanjuPuram,Virudhunagar	Unmarried	6	<button>Update</button> <button>Delete</button>
22	Vijay	45	Chennai	Christian	Actor	2000000	Groom	Premium	Married	9	<button>Update</button> <button>Delete</button>
26	Seetha	25	Therkuvusal,Madurai-08	Hindu	Teacher	25000	Bride	Monthly	Single	9	<button>Update</button> <button>Delete</button>
28	Madhan	22	Kaalavasal,PRC,MAdurai	Hindu	Devloper	58900	Groom	Yearly	Single	9	<button>Update</button> <button>Delete</button>
31	YuvaSree	25	Therkuvusal,Madurai-08	Hindu	Teacher	25000	Bride	Monthly	Single	9	<button>Update</button> <button>Delete</button>
33	Dharshini	25	Therkuvusal,Madurai-08	Hindu	Trainee	25000	Bride	Monthly	Single	9	<button>Update</button> <button>Delete</button>
35	Lakshmi	25	kaalvasal,Madurai-08	Hindu	Tailor	15000	Bride	Monthly	Single	4	<button>Update</button> <button>Delete</button>

SearchName

Add +

15:41 23-07-2024 ENG IN

Screenshot copied to clipboard and saved
Select here to mark up and share.

Finished after 16.136 seconds
 Finished after 16.136 seconds Errors: 0 Failures: 0

>DeleteUserTC [Runner: JUnit 5] (16.069 s)
 DeleteUsertest() (16.069 s)

Failure Trace

```

19
20     driver = new ChromeDriver();
21 }
22
23 @AfterEach
24 public void tearDown() {
25     driver.quit();
26 }
27
28 @Test
29 void DeleteUsertest() throws InterruptedException {
30     driver.get("http://localhost:3000/viewuser");
31     driver.manage().window().maximize();
32
33     driver.findElement(By.name("delete35")).click();
34     Thread.sleep(5000);
35     driver.switchTo().alert().accept();
36
37     WebDriverWait waitDriver = new WebDriverWait(driver, Duration.ofSeconds(5));
38     waitDriver.until(ExpectedConditions.alertIsPresent());
39     Thread.sleep(6000);
40     assertEquals("record has deleted", driver.switchTo().alert().getText());
41 }
42
43
  
```

VIEW PAGE

```

class ViewUser {

WebDriver driver;
@BeforeEach
public void setUp() {
driver = new ChromeDriver();
}

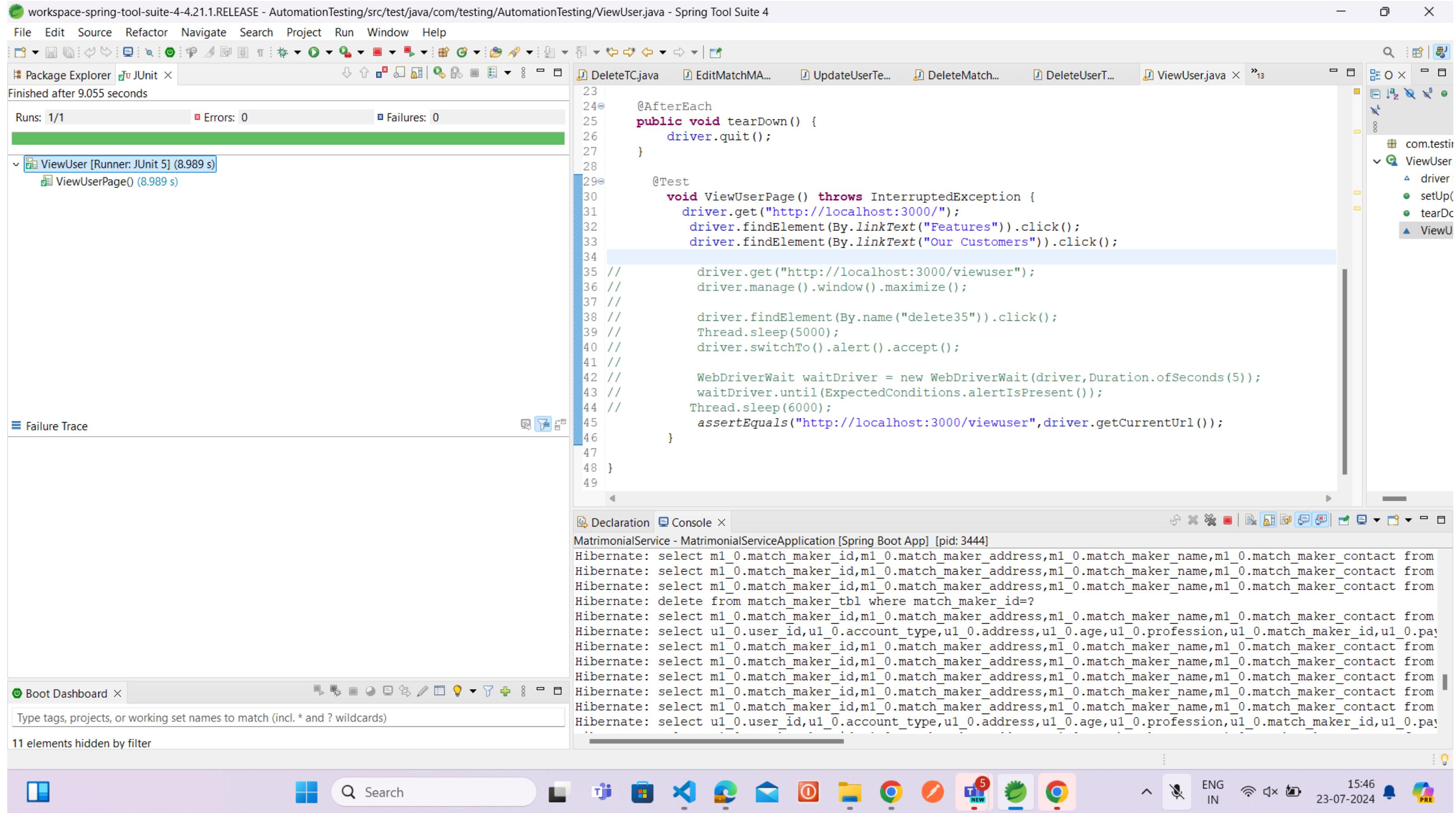
@AfterEach
public void tearDown() {
driver.quit();
}

@Test
void ViewUserPage() throws InterruptedException {
driver.get("http://localhost:3000/");
driver.findElement(By.linkText("Features")).click();
driver.findElement(By.linkText("Our Customers")).click();

// assertEquals("http://localhost:3000/viewuser", driver.getCurrentUrl());
}

@Test
void ViewMatchMakerPage() throws InterruptedException {
driver.get("http://localhost:3000/");
driver.findElement(By.linkText("Features")).click();
driver.findElement(By.linkText("Match Makers")).click();

// assertEquals("http://localhost:3000/view", driver.getCurrentUrl());
}
}
  
```



Selenium IDE

```
// Generated by Selenium IDE

import org.junit.Test;
import org.junit.Before;
import org.junit.After;
import static org.junit.Assert.*;
import static org.hamcrest.CoreMatchers.is;
import static org.hamcrest.core.IsNot.not;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
```

```
import org.openqa.selenium.firefox.FirefoxDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.remote.RemoteWebDriver;
import org.openqa.selenium.remote.DesiredCapabilities;
import org.openqa.selenium.Dimension;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.interactions.Actions;
import org.openqa.selenium.support.ui.ExpectedConditions;
import org.openqa.selenium.support.ui.WebDriverWait;
import org.openqa.selenium.JavascriptExecutor;
import org.openqa.selenium.Alert;
import org.openqa.selenium.Keys;
import java.util.*;
import java.net.MalformedURLException;
import java.net.URL;
public class SignUpTest {
    private WebDriver driver;
    private Map<String, Object> vars;
    JavascriptExecutor js;
    @Before
    public void setUp() {
        driver = new ChromeDriver();
        js = (JavascriptExecutor) driver;
        vars = new HashMap<String, Object>();
    }
    @After
    public void tearDown() {
        driver.quit();
    }
    @Test
    public void signUp() {
        driver.get("http://localhost:3000/");
        driver.manage().window().setSize(new Dimension(1552, 832));
        driver.findElement(By.linkText("SignUp")).click();
        driver.findElement(By.id("uname")).click();
        driver.findElement(By.id("uname")).sendKeys("VashanthTS");
        driver.findElement(By.id("email")).click();
        driver.findElement(By.id("email")).sendKeys("tsv@gmail.com");
        driver.findElement(By.id("pwd")).click();
        driver.findElement(By.id("pwd")).sendKeys("Tsv@1234");
        driver.findElement(By.id("btn")).click();
        assertEquals("Registered Successfully", driver.switchTo().alert().getText());
    }
}
```

Selenium IDE - AutomationTestingLab*

Project: AutomationTestingLab*

Executing ✓ SignUp*

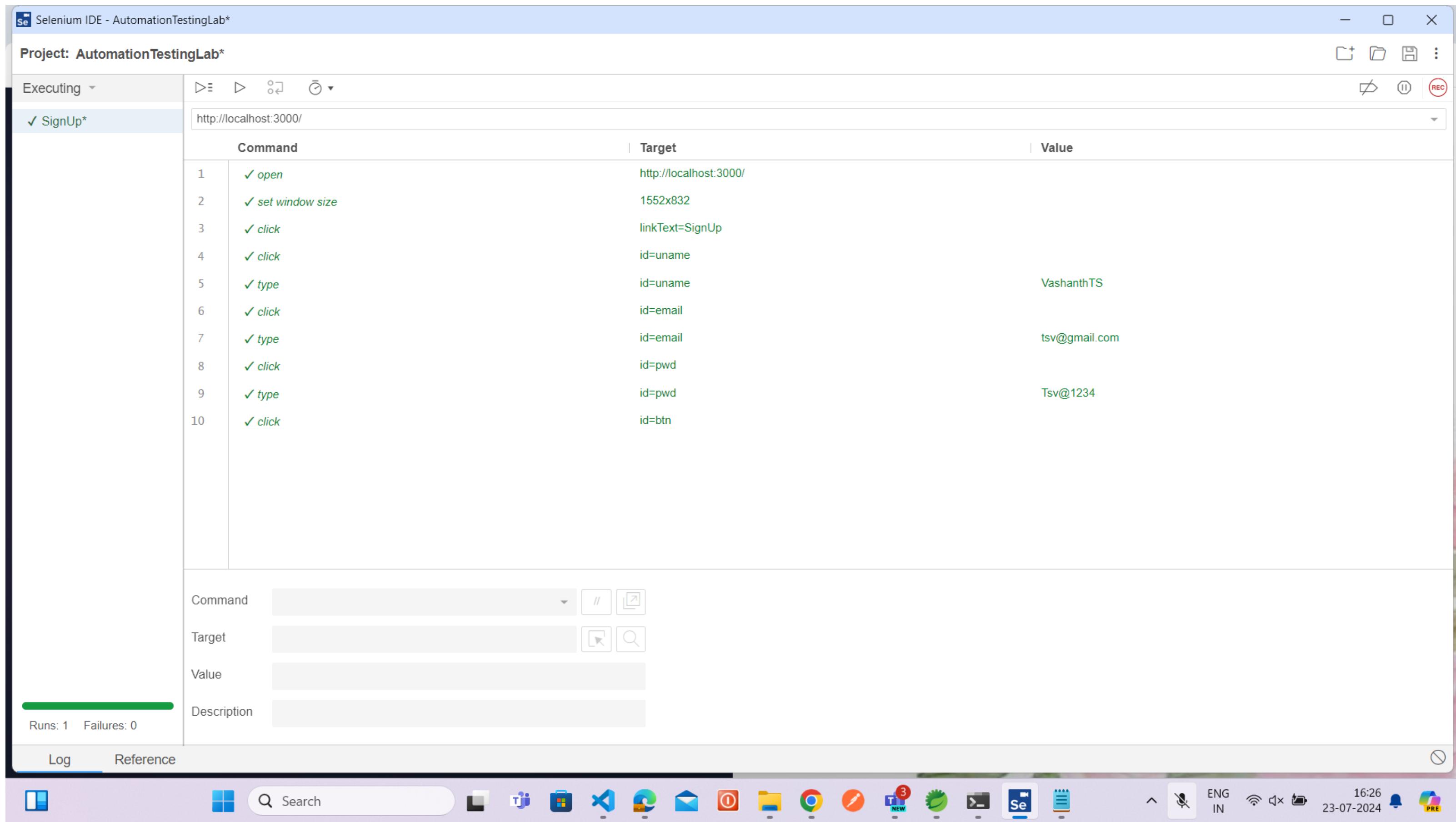
http://localhost:3000/

Command	Target	Value
1 ✓ open	http://localhost:3000/	
2 ✓ set window size	1552x832	
3 ✓ click	linkText=SignUp	
4 ✓ click	id=username	
5 ✓ type	id=username	VashanthTS
6 ✓ click	id=email	
7 ✓ type	id=email	tsv@gmail.com
8 ✓ click	id=password	
9 ✓ type	id=password	Tsv@1234
10 ✓ click	id=btn	

Command Target Value

Runs: 1 Failures: 0

Log Reference



/ Generated by Selenium IDE

```
import org.junit.Test;  
import org.junit.Before;  
import org.junit.After;  
import static org.junit.Assert.*;  
import static org.hamcrest.CoreMatchers.is;  
import static org.hamcrest.core.IsNot.not;  
import org.openqa.selenium.By;  
import org.openqa.selenium.WebDriver;  
import org.openqa.selenium.firefox.FirefoxDriver;  
import org.openqa.selenium.chrome.ChromeDriver;  
import org.openqa.selenium.remote.RemoteWebDriver;  
import org.openqa.selenium.remote.DesiredCapabilities;  
import org.openqa.selenium.Dimension;  
import org.openqa.selenium.WebElement;  
import org.openqa.selenium.interactions.Actions;  
import org.openqa.selenium.support.ui.ExpectedConditions;  
import org.openqa.selenium.support.ui.WebDriverWait;  
import org.openqa.selenium.JavascriptExecutor;  
import org.openqa.selenium.Alert;  
import org.openqa.selenium.Keys;  
import java.util.*;  
import java.net.MalformedURLException;  
import java.net.URL;  
  
public class SignupFailureCaseTest {  
    private WebDriver driver;  
    private Map<String, Object> vars;  
    JavascriptExecutor js;  
    @Before  
    public void setUp() {
```

```

driver = new ChromeDriver();
js = (JavascriptExecutor) driver;
vars = new HashMap<String, Object>();
}
@After
public void tearDown() {
    driver.quit();
}
@Test
public void signupFailureCase() {
    driver.get("http://localhost:3000/");
    driver.manage().window().setSize(new Dimension(1552, 832));
    driver.findElement(By.linkText("SignUp")).click();
    driver.findElement(By.id("uname")).click();
    driver.findElement(By.id("email")).click();
    driver.findElement(By.id("email")).sendKeys("tsv@gmail.com");
    driver.findElement(By.id("pwd")).click();
    driver.findElement(By.id("pwd")).sendKeys("123");
    driver.findElement(By.id("btn")).click();
    driver.findElement(By.id("uname")).click();
    driver.findElement(By.id("uname")).sendKeys("Vasi");
    driver.findElement(By.id("btn")).click();
    assertThat(driver.switchTo().alert().getText(), is("Password Cannot be empty value !! It Should Contain atleast 8 characters!!!"));
}
}

```

Selenium IDE - AutomationTestingLab*

Project: AutomationTestingLab*

Tests + ▶ D E D O C

Search tests...

http://localhost:3000/

	Command	Target	Value
1	✓ open	http://localhost:3000/	
2	✓ set window size	1552x832	
3	✓ click	linkText=SignUp	
4	✓ click	id=uname	
5	✓ click	id=email	
6	✓ type	id=email	tsv@gmail.com
7	✓ click	id=pwd	
8	✓ type	id=pwd	123
9	✓ click	id=btn	
10	✓ click	id=uname	
11	✓ type	id=uname	Vasi
12	✓ click	id=btn	
13	✓ assert alert		Password Cannot be empty value !! It Should Contain atleast 8 characters!!!

Command: open

Target: http://localhost:3000/

Value:

Description:

Log Reference

16:28 23-07-2024 ENG IN WiFi Battery 23-07-2024 Se PRE

```
// Generated by Selenium IDE

import org.junit.Test;
import org.junit.Before;
import org.junit.After;
import static org.junit.Assert.*;
import static org.hamcrest.CoreMatchers.is;
import static org.hamcrest.core.IsNot.not;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.firefox.FirefoxDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.remote.RemoteWebDriver;
import org.openqa.selenium.remote.DesiredCapabilities;
import org.openqa.selenium.Dimension;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.interactions.Actions;
import org.openqa.selenium.support.ui.ExpectedConditions;
import org.openqa.selenium.support.ui.WebDriverWait;
import org.openqa.selenium.JavascriptExecutor;
import org.openqa.selenium.Alert;
import org.openqa.selenium.Keys;
import java.util.*;
import java.net.MalformedURLException;
import java.net.URL;

public class LoginFailedInvalidDetailsTest {

    private WebDriver driver;
    private Map<String, Object> vars;
    JavascriptExecutor js;

    @Before
    public void setUp() {
        driver = new ChromeDriver();
        js = (JavascriptExecutor) driver;
        vars = new HashMap<String, Object>();
    }

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

    @Test
    public void loginFailedInvalidDetails() {
        driver.get("http://localhost:3000/");
        driver.manage().window().setSize(new Dimension(1552, 832));
        driver.findElement(By.cssSelector(".ml-auto > .bg-blue-600")).click();
        driver.findElement(By.linkText("Login")).click();
        driver.findElement(By.name("email")).click();
        driver.findElement(By.name("email")).sendKeys("tsv@gmail.com");
        driver.findElement(By.name("password")).click();
        driver.findElement(By.name("password")).sendKeys("tsv@1234");
        driver.findElement(By.cssSelector(".text-white")).click();
        assertEquals("Invalid credentials. Please try again.", driver.switchTo().alert().getText());
    }
}
```

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

- Project:** AutomationTestingLab*
- Test Case:** LoginFailedInvalidDetails*
- URL:** http://localhost:3000/
- Test Steps (Listed in the Test Editor):**

 1. ✓ open http://localhost:3000/
 2. ✓ set window size 1552x832
 3. ✓ click css=.ml-auto > .bg-blue-600
 4. ✓ click linkText=Login
 5. ✓ click name=email
 6. ✓ type name=email tsv@gmail.com
 7. ✓ click name=password
 8. ✓ type name=password tsv@1234
 9. ✓ click css=.text-white
 10. ✓ assert alert Invalid credentials. Please try again.

- Test Editor Form (Bottom):**

Command	open	//	↶ ↷
Target	http://localhost:3000/	↶ ↷	🔍
Value			
Description			
- Bottom Navigation:** Log, Reference, and a small icon.
- System Taskbar:** Shows various application icons, system status (Wi-Fi, battery), and system time (16:30, 23-07-2024).

Generated by Selenium IDE

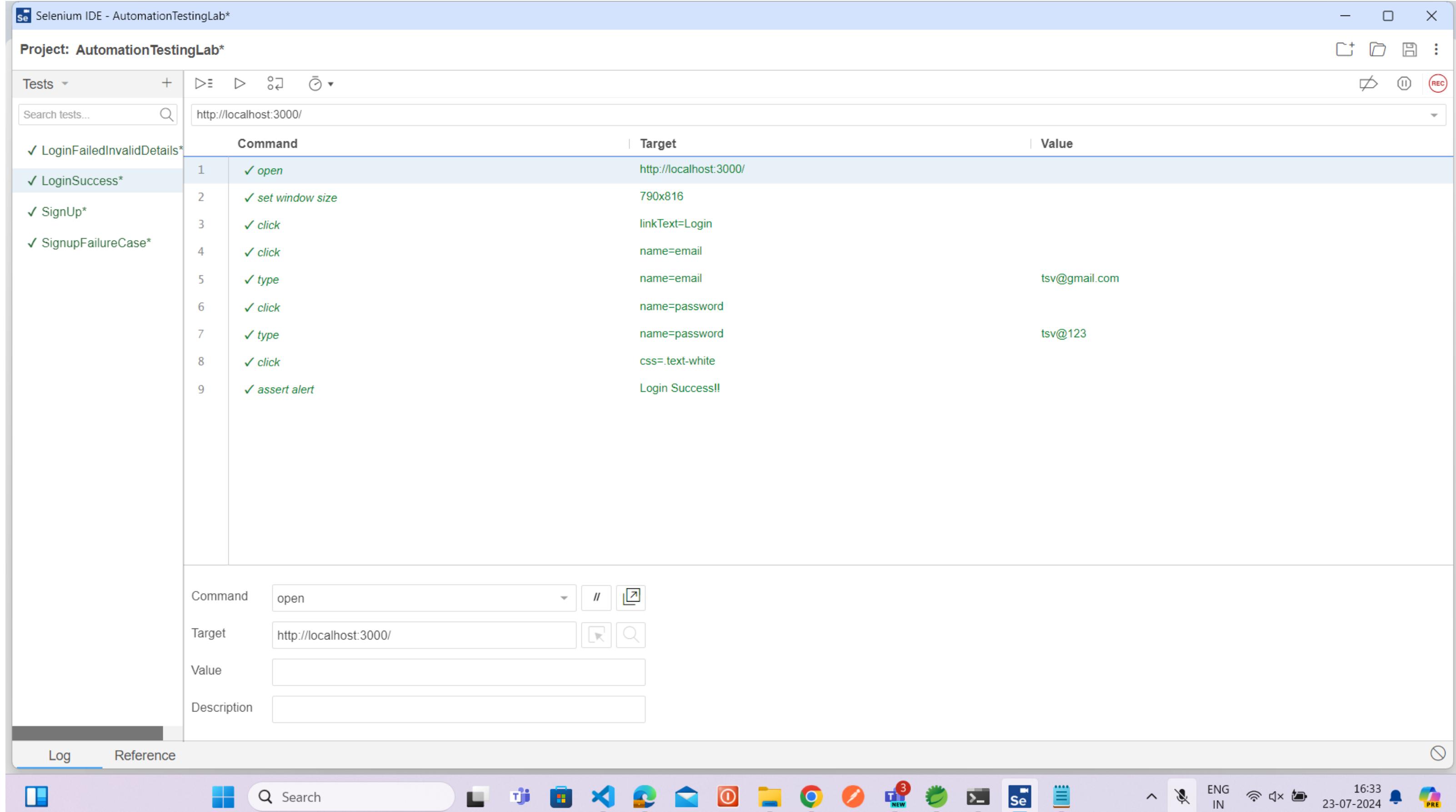
```
import org.junit.Test;  
import org.junit.Before;  
import org.junit.After;  
import static org.junit.Assert.*;  
import static org.hamcrest.CoreMatchers.is;  
import static org.hamcrest.core.IsNot.not;  
import org.openqa.selenium.By;  
import org.openqa.selenium.WebDriver;  
import org.openqa.selenium.firefox.FirefoxDriver;  
import org.openqa.selenium.chrome.ChromeDriver;  
import org.openqa.selenium.remote.RemoteWebDriver;  
import org.openqa.selenium.remote.DesiredCapabilities;  
import org.openqa.selenium.Dimension;  
import org.openqa.selenium.WebElement;  
import org.openqa.selenium.interactions.Actions;  
import org.openqa.selenium.support.ui.ExpectedConditions;  
import org.openqa.selenium.support.ui.WebDriverWait;  
import org.openqa.selenium.JavascriptExecutor;  
import org.openqa.selenium.Alert;  
import org.openqa.selenium.Keys;  
import java.util.*;  
import java.net.MalformedURLException;  
import java.net.URL;  
  
public class LoginSuccessTest{  
    private WebDriver driver;  
    private Map<String, Object> vars;  
    JavascriptExecutor js;
```

```
@Before
```

```
public void setUp() {  
    driver = new ChromeDriver();  
    js = (JavascriptExecutor) driver;  
    vars = new HashMap<String, Object>();  
}  
  
@After  
public void tearDown() {  
    driver.quit();  
}  
  
@Test  
public void loginSuccess() {  
    driver.get("http://localhost:3000/");  
    driver.manage().window().setSize(new Dimension(790, 816));  
    driver.findElement(By.linkText("Login")).click();  
    driver.findElement(By.name("email")).click();  
    driver.findElement(By.name("email")).sendKeys("tsv@gmail.com");  
    driver.findElement(By.name("password")).click();  
    driver.findElement(By.name("password")).sendKeys("tsv@123");  
    driver.findElement(By.cssSelector(".text-white")).click();  
    assertThat(driver.switchTo().alert().getText(), is("Login Success!!"));  
}
```

```
}
```

```
}
```



The screenshot shows the Selenium IDE interface with the project 'AutomationTestingLab'. A test case named 'LoginSuccess' is selected. The test steps are listed in a table:

	Command	Target	Value
1	✓ open	http://localhost:3000/	
2	✓ set window size	790x816	
3	✓ click	linkText=Login	
4	✓ click	name=email	
5	✓ type	name=email	tsv@gmail.com
6	✓ click	name=password	
7	✓ type	name=password	tsv@123
8	✓ click	css=.text-white	
9	✓ assert alert	Login Success!!	

Below the table, there is a form to add a new step:

Command: open
Target: http://localhost:3000/
Value:
Description:

/ Generated by Selenium IDE

```
import org.junit.Test;  
import org.junit.Before;
```

```
import org.junit.After;
import static org.junit.Assert.*;
import static org.hamcrest.CoreMatchers.is;
import static org.hamcrest.core.IsNot.not;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.firefox.FirefoxDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.remote.RemoteWebDriver;
import org.openqa.selenium.remote.DesiredCapabilities;
import org.openqa.selenium.Dimension;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.interactions.Actions;
import org.openqa.selenium.support.ui.ExpectedConditions;
import org.openqa.selenium.support.ui.WebDriverWait;
import org.openqa.selenium.JavascriptExecutor;
import org.openqa.selenium.Alert;
import org.openqa.selenium.Keys;
import java.util.*;
import java.net.MalformedURLException;
import java.net.URL;
public class AddUserTest{
    private WebDriver driver;
    private Map<String, Object> vars;
    JavascriptExecutor js;
    @Before
    public void setUp() {
        driver = new ChromeDriver();
        js = (JavascriptExecutor) driver;
        vars = new HashMap<String, Object>();
    }
    @After
    public void tearDown() {
        driver.quit();
    }
    @Test
    public void addUser() {
        driver.get("http://localhost:3000/");
        driver.manage().window().setSize(new Dimension(1536, 816));
        driver.findElement(By.linkText("Features")).click();
        driver.findElement(By.linkText("Our Customers")).click();
        driver.findElement(By.linkText("Add +")).click();
        driver.findElement(By.name("uname")).click();
        driver.findElement(By.name("uname")).sendKeys("Karan");
        driver.findElement(By.name("actype")).click();
        driver.findElement(By.name("actype")).sendKeys("Groom");
        driver.findElement(By.name("addr")).click();
        driver.findElement(By.name("addr")).sendKeys("Chennai");
        driver.findElement(By.name("age")).click();
        driver.findElement(By.name("age")).click();
        driver.findElement(By.name("age")).sendKeys("32");
        driver.findElement(By.name("sal")).click();
        driver.findElement(By.name("sal")).sendKeys("75000");
        driver.findElement(By.name("job")).click();
        driver.findElement(By.name("job")).sendKeys("Buisness");
        driver.findElement(By.name("pay")).click();
        driver.findElement(By.name("pay")).sendKeys("Premium");
```

```

driver.findElement(By.name("religion")).click();
driver.findElement(By.name("religion")).sendKeys("Hindu");
driver.findElement(By.name("sts")).click();
driver.findElement(By.name("sts")).sendKeys("Unmarried");
driver.findElement(By.id("selectId")).click();
{
    WebElement dropdown = driver.findElement(By.id("selectId"));
    dropdown.findElement(By.xpath("//option[. = '6']")).click();
}
driver.findElement(By.id("submit")).click();
}
}

```

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

- Project:** AutomationTestingLab*
- Test Case:** AddUser*
- Status:** Executing
- URL:** http://localhost:3000/
- Script Log:**

Line	Command	Target	Value
14	✓ type	name=age	32
15	✓ click	name=sal	
16	✓ type	name=sal	75000
17	✓ click	name=job	
18	✓ type	name=job	Buissness
19	✓ click	name=pay	
20	✓ type	name=pay	Premium
21	✓ click	name=religion	
22	✓ type	name=religion	Hindu
23	✓ click	name=sts	
24	✓ type	name=sts	Unmarried
25	✓ click	id=selectId	
26	✓ select	id=selectId	label=6
27	✓ click	id=submit	
- Toolbars and Buttons:** Includes options for executing, recording, and saving the test.
- Bottom Status Bar:** Shows 'Runs: 1 Failures: 0'.
- Taskbar:** Shows various application icons and system status indicators like battery level and network connection.

/ Generated by Selenium IDE

```

import org.junit.Test;
import org.junit.Before;
import org.junit.After;
import static org.junit.Assert.*;
import static org.hamcrest.CoreMatchers.is;
import static org.hamcrest.core.IsNot.not;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.firefox.FirefoxDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.remote.RemoteWebDriver;
import org.openqa.selenium.remote.DesiredCapabilities;
import org.openqa.selenium.Dimension;

```

```
import org.openqa.selenium.WebElement;
import org.openqa.selenium.interactions.Actions;
import org.openqa.selenium.support.ui.ExpectedConditions;
import org.openqa.selenium.support.ui.WebDriverWait;
import org.openqa.selenium.JavascriptExecutor;
import org.openqa.selenium.Alert;
import org.openqa.selenium.Keys;
import java.util.*;
import java.net.MalformedURLException;
import java.net.URL;
public class AddUserIfFieldsareNullTest {
    private WebDriver driver;
    private Map<String, Object> vars;
    JavascriptExecutor js;
    @Before
    public void setUp() {
        driver = new ChromeDriver();
        js = (JavascriptExecutor) driver;
        vars = new HashMap<String, Object>();
    }
    @After
    public void tearDown() {
        driver.quit();
    }
    @Test
    public void addUserIfFieldsareNull() {
        driver.get("http://localhost:3000/");
        driver.manage().window().setSize(new Dimension(1552, 832));
        driver.findElement(By.linkText("Features")).click();
        driver.findElement(By.linkText("Our Customers")).click();
        driver.findElement(By.linkText("Add +")).click();
        driver.findElement(By.cssSelector("div:nth-child(3)").click();
        driver.findElement(By.name("actype")).click();
        driver.findElement(By.name("actype")).sendKeys("Groom");
        driver.findElement(By.name("addr")).click();
        driver.findElement(By.name("age")).click();
        driver.findElement(By.name("age")).sendKeys("23");
        driver.findElement(By.name("sal")).click();
        driver.findElement(By.name("sal")).sendKeys("32345");
        driver.findElement(By.name("job")).click();
        driver.findElement(By.name("job")).sendKeys("Actor");
        driver.findElement(By.name("pay")).click();
        driver.findElement(By.name("religion")).click();
        driver.findElement(By.name("religion")).sendKeys("Christian");
        driver.findElement(By.name("sts")).click();
        driver.findElement(By.name("sts")).sendKeys("Unmarried");
        driver.findElement(By.id("selectId")).click();
        {
            WebElement dropdown = driver.findElement(By.id("selectId"));
            dropdown.findElement(By.xpath("//option[. = '4']")).click();
        }
        driver.findElement(By.id("submit")).click();
        assertEquals(driver.switchTo().alert().getText(), "Please enter Name !!! Please Enter the Valid Inputs!!!!");
    }
}
```

Selenium IDE - AutomationTestingLab*

Project: AutomationTestingLab*

Tests + ▶ ⌂ ⌂ REC

Search tests... Run current test Ctrl+R

	Command	Target	Value
✓ AddUser*	10 ✓ click	name=age	
✓ AddUser(IfFieldsareNull)*	11 ✓ type	name=age	23
✓ LoginFailedInvalidDetails*	12 ✓ click	name=sal	
✓ LoginSuccess*	13 ✓ type	name=sal	32345
✓ SignUp*	14 ✓ click	name=job	
✓ SignupFailureCase*	15 ✓ type	name=job	Actor
	16 ✓ click	name=pay	
	17 ✓ click	name=religion	
	18 ✓ type	name=religion	Christian
	19 ✓ click	name=sts	
	20 ✓ type	name=sts	Unmarried
	21 ✓ click	id=selectId	
	22 ✓ select	id=selectId	label=4
	23 ✓ click	id=submit	
	24 ✓ assert alert	Please enter Name !!! Please Enter the Valid Inputs!!!	

Command Target Value

Target

Description

Log Reference

/ Generated by Selenium IDE

```

import org.junit.Test;
import org.junit.Before;
import org.junit.After;
import static org.junit.Assert.*;
import static org.hamcrest.CoreMatchers.is;
import static org.hamcrest.core.IsNot.not;
import org.openqa.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.firefox.FirefoxDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.remote.RemoteWebDriver;
import org.openqa.selenium.remote.DesiredCapabilities;
import org.openqa.selenium.Dimension;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.interactions.Actions;
import org.openqa.selenium.support.ui.ExpectedConditions;
import org.openqa.selenium.support.ui.WebDriverWait;
import org.openqa.selenium.JavascriptExecutor;
import org.openqa.selenium.Alert;
import org.openqa.selenium.Keys;
import java.util.*;
import java.net.MalformedURLException;
import java.net.URL;
public class AddUserInvalidInputTest {
    private WebDriver driver;
    private Map<String, Object> vars;
}

```

```
JavascriptExecutor js;  
  
{@Before  
  
public void setUp() {  
    driver = new ChromeDriver();  
    js = (JavascriptExecutor) driver;  
    vars = new HashMap<String, Object>();  
}  
  
{@After  
  
public void tearDown() {  
    driver.quit();  
}  
  
{@Test  
  
public void addUserInvalidInput() {  
    driver.get("http://localhost:3000/");  
    driver.manage().window().setSize(new Dimension(1552, 832));  
    driver.findElement(By.linkText("Features")).click();  
    driver.findElement(By.linkText("Our Customers")).click();  
    driver.findElement(By.linkText("Add +")).click();  
    driver.findElement(By.name("uname")).click();  
    driver.findElement(By.name("uname")).sendKeys("KarthiKeyan");  
    driver.findElement(By.name("actype")).click();  
    driver.findElement(By.name("actype")).sendKeys("Groom");  
    driver.findElement(By.name("addr")).click();  
    driver.findElement(By.name("age")).click();  
    {  
        WebElement element = driver.findElement(By.name("age"));  
        Actions builder = new Actions(driver);  
        builder.moveToElement(element).clickAndHold().perform();  
    }  
    {  
        WebElement element = driver.findElement(By.name("age"));  
        Actions builder = new Actions(driver);  
        builder.moveToElement(element).perform();  
    }  
    {  
        WebElement element = driver.findElement(By.name("age"));  
        Actions builder = new Actions(driver);  
        builder.moveToElement(element).release().perform();  
    }  
    driver.findElement(By.cssSelector("form")).click();  
    driver.findElement(By.name("addr")).sendKeys("Chennai");  
    driver.findElement(By.name("age")).click();  
    driver.findElement(By.name("age")).sendKeys("15");  
    driver.findElement(By.name("sal")).click();  
    driver.findElement(By.name("uname")).click();  
    driver.findElement(By.name("uname")).sendKeys("Karthika");  
    driver.findElement(By.name("actype")).click();  
    driver.findElement(By.name("sal")).click();  
    driver.findElement(By.name("sal")).sendKeys("12345");  
    driver.findElement(By.name("job")).click();  
    driver.findElement(By.name("job")).sendKeys("Tailor");  
    driver.findElement(By.name("pay")).click();  
    driver.findElement(By.name("pay")).sendKeys("Monthly");  
    driver.findElement(By.name("religion")).click();  
    driver.findElement(By.name("religion")).sendKeys("Hindu");  
    driver.findElement(By.name("sts")).click();  
    driver.findElement(By.name("sts")).sendKeys("Unmarried");  
}
```

```

driver.findElement(By.id("selectId")).click();
{
    WebElement dropdown = driver.findElement(By.id("selectId"));
    dropdown.findElement(By.xpath("//option[. = '4']")).click();
}
driver.findElement(By.id("submit")).click();
assertThat(driver.switchTo().alert().getText(), is("Minimum Age is 18 !!! Please Enter the Valid Inputs!!!"));
}
}

```

Selenium IDE - AutomationTestingLab*

Project: AutomationTestingLab*

Tests + D E Run current test Ctrl+R

	Command	Target	Value
✓ AddUser*	22 ✓ click	name=actype	
✓ AddUser(IfFieldsareNull)*	23 ✓ click	name=sal	
✓ AddUser(InvalidInput)*	24 ✓ type	name=sal	12345
✓ LoginFailedInvalidDetails*	25 ✓ click	name=job	
✓ LoginSuccess*	26 ✓ type	name=job	Tailor
✓ SignUp*	27 ✓ click	name=pay	
✓ SignupFailureCase*	28 ✓ type	name=pay	Monthly
	29 ✓ click	name=religion	
	30 ✓ type	name=religion	Hindu
	31 ✓ click	name=sts	
	32 ✓ type	name=sts	Unmarried
	33 ✓ click	id=selectId	
	34 ✓ select	id=selectId	label=4
	35 ✓ click	id=submit	
	36 ✓ assert alert		Minimum Age is 18 !!! Please Enter the Valid Inputs!!!

Command

Target

Value

Description

Log Reference

/ Generated by Selenium IDE

```

import org.junit.Test;
import org.junit.Before;
import org.junit.After;
import static org.junit.Assert.*;
import static org.hamcrest.CoreMatchers.is;
import static org.hamcrest.core.IsNot.not;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.firefox.FirefoxDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.remote.RemoteWebDriver;
import org.openqa.selenium.remote.DesiredCapabilities;
import org.openqa.selenium.Dimension;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.interactions.Actions;

```

```
import org.openqa.selenium.support.ui.ExpectedConditions;
import org.openqa.selenium.support.ui.WebDriverWait;
import org.openqa.selenium.JavascriptExecutor;
import org.openqa.selenium.Alert;
import org.openqa.selenium.Keys;
import java.util.*;
import java.net.MalformedURLException;
import java.net.URL;

public class AddMatchMakerTest {

    private WebDriver driver;
    private Map<String, Object> vars;
    JavascriptExecutor js;

    @Before
    public void setUp() {
        driver = new ChromeDriver();
        js = (JavascriptExecutor) driver;
        vars = new HashMap<String, Object>();
    }

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

    @Test
    public void addMatchMaker() {
        driver.get("http://localhost:3000/");
        driver.manage().window().setSize(new Dimension(1552, 832));
        driver.findElement(By.linkText("Features")).click();
        driver.findElement(By.linkText("Match Makers")).click();
        driver.findElement(By.linkText("Add +")).click();
        driver.findElement(By.name("mname")).click();
        driver.findElement(By.name("mname")).sendKeys("Kanmani");
        driver.findElement(By.name("addr")).click();
        driver.findElement(By.name("addr")).sendKeys("Chinnalampati,Dgl");
        driver.findElement(By.name("cmobile")).click();
        driver.findElement(By.name("cmobile")).sendKeys("835915813");
        driver.findElement(By.id("submit")).click();
    }
}
```

Selenium IDE - AutomationTestingLab*

Project: AutomationTestingLab*

Tests + ▶ ⏪ ⏴ ⏵ ⏷

Search tests... Q

http://localhost:3000/

	Command	Target	Value
1	✓ open	http://localhost:3000/	
2	✓ set window size	1552x832	
3	✓ click	linkText=Features	
4	✓ click	linkText=Match Makers	
5	✓ click	linkText=Add +	
6	✓ click	name=mname	
7	✓ type	name=mname	Kanmani
8	✓ click	name=addr	
9	✓ type	name=addr	Chinnalampati,Dgl
10	✓ click	name=cmobile	
11	✓ type	name=cmobile	835915813
12	✓ click	id=submit	

Command: click ▼ // ✖

Target: id=submit ▼ ✖ 🔍

Value:

Description:

Log Reference

ed by Selenium IDE

```

import org.junit.Test;
import org.junit.Before;
import org.junit.After;
import static org.junit.Assert.*;
import static org.hamcrest.CoreMatchers.is;
import static org.hamcrest.core.IsNot.not;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.firefox.FirefoxDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.remote.RemoteWebDriver;
import org.openqa.selenium.remote.DesiredCapabilities;
import org.openqa.selenium.Dimension;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.interactions.Actions;
import org.openqa.selenium.support.ui.ExpectedConditions;
import org.openqa.selenium.support.ui.WebDriverWait;
import org.openqa.selenium.JavascriptExecutor;
import org.openqa.selenium.Alert;
import org.openqa.selenium.Keys;
import java.util.*;
import java.net.MalformedURLException;
import java.net.URL;

public class AddMatchMakerIfinputNullTest {
    private WebDriver driver;
    private Map<String, Object> vars;
    JavascriptExecutor js;
    @Before

```

```

public void setUp() {
    driver = new ChromeDriver();
    js = (JavascriptExecutor) driver;
    vars = new HashMap<String, Object>();
}

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

@Test
public void addMatchMakerInputNull() {
    driver.get("http://localhost:3000/");
    driver.manage().window().setSize(new Dimension(1552, 832));
    driver.findElement(By.linkText("Features")).click();
    driver.findElement(By.linkText("Match Makers")).click();
    driver.findElement(By.linkText("Add +")).click();
    driver.findElement(By.name("mname")).click();
    driver.findElement(By.name("addr")).click();
    driver.findElement(By.name("addr")).sendKeys("chennai");
    driver.findElement(By.name("cmobile")).click();
    driver.findElement(By.name("cmobile")).sendKeys("908765423");
    driver.findElement(By.id("submit")).click();
    assertThat(driver.switchTo().alert().getText(), is("Please Enter the Valid Inputs!!!"));
}
}

```

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

- Title Bar:** Selenium IDE - AutomationTestingLab*
- Toolbar:** Includes icons for executing, recording, and other functions.
- Project:** AutomationTestingLab*
- Test Name:** AddMatchMakerInputNull (highlighted in red)
- Run Current Test:** Run current test Ctrl+R
- Table of Recorded Steps:**

	Command	Target	Value
1	✓ open	http://localhost:3000/	
2	✓ set window size	1552x832	
3	✓ click	linkText=Features	
4	✓ click	linkText=Match Makers	
5	✓ click	linkText=Add +	
6	✓ click	name=mname	
7	✓ click	name=addr	
8	✓ type	name=addr	chennai
9	✓ click	name=cmobile	
10	✓ type	name=cmobile	908765423
11	✓ click	id=submit	
12	✓ assert alert	Please Enter the Valid Inputs!!!	
- Bottom Panel:**
 - Command, Target, Value, Description input fields.
 - Run status: Runs: 1 Failures: 0
 - Log tab is selected.
 - System tray icons: ENG IN, 16:45, 23-07-2024, PRB.

```

import org.openqa.selenium.Alert;
import org.openqa.selenium.Keys;

```

```
import java.util.*;
import java.net.MalformedURLException;
import java.net.URL;
public class EditUserTest {
    private WebDriver driver;
    private Map<String, Object> vars;
    JavascriptExecutor js;
    @Before
    public void setUp() {
        driver = new ChromeDriver();
        js = (JavascriptExecutor) driver;
        vars = new HashMap<String, Object>();
    }
    @After
    public void tearDown() {
        driver.quit();
    }
    @Test
    public void editUser() {
        driver.get("http://localhost:3000/");
        driver.manage().window().setSize(new Dimension(1552, 832));
        driver.findElement(By.linkText("Features")).click();
        driver.findElement(By.linkText("Our Customers")).click();
        driver.findElement(By.name("update37")).click();
        js.executeScript("window.scrollTo(0,44.79999923706055)");
        driver.findElement(By.name("uname")).click();
        driver.findElement(By.name("uname")).sendKeys("KaranKumar");
        driver.findElement(By.name("age")).click();
        driver.findElement(By.name("age")).sendKeys("30");
        driver.findElement(By.cssSelector("form")).click();
        driver.findElement(By.id("selectId")).click();
        {
            WebElement dropdown = driver.findElement(By.id("selectId"));
            dropdown.findElement(By.xpath("//option[. = '17']")).click();
        }
        driver.findElement(By.id("submit")).click();
        assertThat(driver.switchTo().alert().getText(), is(" Updated Successfully"));
    }
}
```

Selenium IDE - AutomationTestingLab*

Project: AutomationTestingLab*

Tests + ▶ ⌂ ⌂ ⌂ ⌂ ⌂

Search tests...

http://localhost:3000/

	Command	Target	Value
1	✓ open	http://localhost:3000/	
2	✓ set window size	1552x832	
3	✓ click	linkText=Features	
4	✓ click	linkText=Our Customers	
5	✓ click	name=update37	
6	✓ run script	window.scrollTo(0,44.79999923706055)	
7	✓ click	name=username	
8	✓ type	name=username	KaranKumar
9	✓ click	name=age	
10	✓ type	name=age	30
11	✓ click	css=form	
12	✓ click	id=selectId	
13	✓ select	id=selectId	label=17
14	✓ click	id=submit	
15	assert alert	Updated Successfully	

Command: assert alert

Target: Updated Successfully

Value:

Description:

Log Reference

The screenshot shows the Selenium IDE interface with a recorded test case. The test case consists of 15 commands, including opening the URL, setting window size, navigating to specific links, interacting with form fields (username, age), selecting dropdown values, and asserting an alert message. The 'assert alert' command at step 15 has a value of 'Updated Successfully'. Below the test list, there is a preview section for the last command, showing the command, target, value, and description fields.

```
// Generated by Selenium IDE
import org.junit.Test;
import org.junit.Before;
import org.junit.After;
import static org.junit.Assert.*;
import static org.hamcrest.CoreMatchers.is;
import static org.hamcrest.core.IsNot.not;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.firefox.FirefoxDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.remote.RemoteWebDriver;
import org.openqa.selenium.remote.DesiredCapabilities;
import org.openqa.selenium.Dimension;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.interactions.Actions;
import org.openqa.selenium.support.ui.ExpectedConditions;
import org.openqa.selenium.support.ui.WebDriverWait;
import org.openqa.selenium.JavascriptExecutor;
import org.openqa.selenium.Alert;
import org.openqa.selenium.Keys;
import java.util.*;
import java.net.MalformedURLException;
import java.net.URL;
public class EditMatchMAkerTest {
    private WebDriver driver;
    private Map<String, Object> vars;
    JavascriptExecutor js;
```

```
@Before  
  
public void setUp() {  
  
    driver = new ChromeDriver();  
  
    js = (JavascriptExecutor) driver;  
  
    vars = new HashMap<String, Object>();  
  
}  
  
@After  
  
public void tearDown() {  
  
    driver.quit();  
  
}  
  
@Test  
  
public void editMAatchMAker() {  
  
    driver.get("http://localhost:3001");  
  
    driver.manage().window().setSize(new Dimension(1000, 600));  
  
    driver.findElement(By.cssSelector("a[href='http://localhost:3001/edit']")).click();  
  
    driver.findElement(By.cssSelector("a[href='http://localhost:3001/edit']")).click();  
  
    driver.findElement(By.linkText("Edit Match")).click();  
  
    driver.findElement(By.linkText("Edit Match")).click();  
  
    driver.findElement(By.name("username")).sendKeys("user1");  
  
    driver.findElement(By.name("age")).sendKeys("25");  
  
    driver.findElement(By.name("age")).sendKeys("25");  
  
    driver.findElement(By.id("submit")).click();  
  
    assertEquals("Match updated successfully!", driver.switchTo().alert().getText());  
  
}  
  
}
```

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

- Title Bar:** Selenium IDE - AutomationTestingLab*
- Project:** AutomationTestingLab*
- Status Bar:** Paused in debugger
- Toolbar:** Tests, +, D, >, □, II, ⏹, REC
- Search Bar:** Search tests... (with a magnifying glass icon)
- URL Bar:** http://localhost:3000/
- Test List:** A list of test cases with green checkmarks:
 - AddMatchMaker*
 - AddMatchMakerIfInputNull
 - AddUser*
 - AddUser>IfFieldsareNull)*
 - AddUser>InvalidInput)*
 - EditMatchMAker*** (highlighted in blue)
 - EditUser*
 - LoginFailedInvalidDetails*
 - LoginSuccess*
 - SignUp*
 - SignupFailureCase*
- Test Details Table:** Displays the steps for the selected test case "EditMatchMAker*".

	Command	Target	Value
1	✓ open	http://localhost:3000/	
2	✓ set window size	802x816	
3	✓ click	css=.w-7	
4	✓ click	css=.max-lg\3A border-b:nth-child(4)	
5	✓ click	linkText=Features	
6	✓ click	linkText=Match Makers	
7	✓ click	name=update17	
8	✓ click	name=addr	
9	✓ type	name=addr	Chinnalampati,Dgl-620097
10	✓ click	id=submit	
11	assert alert	Updated Successfully	
- Bottom Panel:** Command, Target, Value, Description input fields with associated icons.
- Taskbar:** chrome-extension://mooikfkahbdckldjjndioackbalphokd/index.html#
- System Icons:** Search, Task View, File Explorer, VS Code, Mail, OneDrive, Google Chrome, Microsoft Edge, Task Manager, Task View, System Tray (ENG IN, Battery, Volume, Network, Date/Time).

Delete user

```

// Generated by Selenium IDE
import org.junit.Test;
import org.junit.Before;
import org.junit.After;
import static org.junit.Assert.*;
import static org.hamcrest.CoreMatchers.is;
import static org.hamcrest.core.IsNot.not;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.firefox.FirefoxDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.remote.RemoteWebDriver;
import org.openqa.selenium.remote.DesiredCapabilities;
import org.openqa.selenium.Dimension;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.interactions.Actions;
import org.openqa.selenium.support.ui.ExpectedConditions;
import org.openqa.selenium.support.ui.WebDriverWait;
import org.openqa.selenium.JavascriptExecutor;
import org.openqa.selenium.Alert;
import org.openqa.selenium.Keys;
import java.util.*;
import java.net.MalformedURLException;
import java.net.URL;
public class DltusrTest {
    private WebDriver driver;
    private Map<String, Object> vars;
    JavascriptExecutor js;
    @Before
    public void setUp() {
        driver = new ChromeDriver();
        js = (JavascriptExecutor) driver;
        vars = new HashMap<String, Object>();
    }
    @After
    public void tearDown() {
        driver.quit();
    }
    @Test
    public void dltusr() {
        driver.get("http://localhost:3000/landing");
        driver.manage().window().setSize(new Dimension(1552, 832));
        driver.findElement(By.cssSelector(".space-y-6")).click();
        driver.findElement(By.linkText("Match Makers")).click();
        driver.findElement(By.name("delete18")).click();
    }
}

```

Delete MAtchmaker

```

// Generated by Selenium IDE
import org.junit.Test;
import org.junit.Before;
import org.junit.After;
import static org.junit.Assert.*;
import static org.hamcrest.CoreMatchers.is;
import static org.hamcrest.core.IsNot.not;

```

```

import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.firefox.FirefoxDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.remote.RemoteWebDriver;
import org.openqa.selenium.remote.DesiredCapabilities;
import org.openqa.selenium.Dimension;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.interactions.Actions;
import org.openqa.selenium.support.ui.ExpectedConditions;
import org.openqa.selenium.support.ui.WebDriverWait;
import org.openqa.selenium.JavascriptExecutor;
import org.openqa.selenium.Alert;
import org.openqa.selenium.Keys;
import java.util.*;
import java.net.MalformedURLException;
import java.net.URL;

public class DltCustomerTest {
    private WebDriver driver;
    private Map<String, Object> vars;
    JavascriptExecutor js;

    @Before
    public void setUp() {
        driver = new ChromeDriver();
        js = (JavascriptExecutor) driver;
        vars = new HashMap<String, Object>();
    }

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

    @Test
    public void dltCustomer() {
        driver.get("http://localhost:3000/");
        driver.manage().window().setSize(new Dimension(1552, 832));
        driver.findElement(By.linkText("Features")).click();
        driver.findElement(By.linkText("Our Customers")).click();
        driver.findElement(By.name("delete26")).click();
        assertEquals(driver.switchTo().alert().getText(), "Do you want to delete");
        driver.switchTo().alert().accept();
        assertEquals(driver.switchTo().alert().getText(), "record has deleted");
    }
}

```

```

// Generated by Selenium IDE
import org.junit.Test;
import org.junit.Before;
import org.junit.After;
import static org.junit.Assert.*;
import static org.hamcrest.CoreMatchers.is;
import static org.hamcrest.core.IsNot.not;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.firefox.FirefoxDriver;
import org.openqa.selenium.chrome.ChromeDriver;

```

```
import org.openqa.selenium.remote.RemoteWebDriver;
import org.openqa.selenium.remote.DesiredCapabilities;
import org.openqa.selenium.Dimension;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.interactions.Actions;
import org.openqa.selenium.support.ui.ExpectedConditions;
import org.openqa.selenium.support.ui.WebDriverWait;
import org.openqa.selenium.JavascriptExecutor;
import org.openqa.selenium.Alert;
import org.openqa.selenium.Keys;
import java.util.*;
import java.net.MalformedURLException;
import java.net.URL;

public class ViewMatchTest {

    private WebDriver driver;
    private Map<String, Object> vars;
    JavascriptExecutor js;

    @Before
    public void setUp() {
        driver = new ChromeDriver();
        js = (JavascriptExecutor) driver;
        vars = new HashMap<String, Object>();
    }

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

    @Test
    public void viewMatch() {
        driver.get("http://localhost:3000/");
        driver.manage().window().setSize(new Dimension(798, 816));
        driver.findElement(By.cssSelector(".max-lg\\3A border-b:nth-child(4)").click());
        driver.findElement(By.linkText("Features")).click();
        driver.findElement(By.linkText("Match Makers")).click();
    }
}
```

http://localhost:3000/

	Command	Target	Value
1	open	http://localhost:3000/	
2	set window size	798x816	
3	click	css=.max-lg\3A border-b:nth-child(4)	
4	click	linkText=Features	
5	click	linkText=Match Makers	

http://localhost:3000/

	Command	Target
1	✓ open	http://localhost:3000/
2	✓ set window size	1552x832
3	✓ click	linkText=Features
4	✓ click	linkText=Our Customers