TRƯỜNG ĐẠI HỌC VĂN LANG

**KHOA CÔNG NGHỆ THÔNG TIN**

A red and black shield with white logo

Description automatically generated

**KIỂM THỬ TỰ ĐỘNG**

NGÀNH: CÔNG NGHỆ THÔNG TIN

***VER1.1 :***

**KIỂM THỬ TỰ ĐỘNG CHO CÁC CHỨC NĂNG TRONG HỆ THỐNG QUẢN LÝ PHÂN CÔNG GIẢNG DẠY**

**SVTH:**

**GVHD:**

**Trần Quốc Huy – 2174802010294**

**Lê Văn Học - 2174802010422**

**GVHD: Phan Thị Hồng**

TP. Hồ Chí Minh – năm 2025

Mục lục

[**1.1 Kiểm thử chức năng Thêm học kỳ mới (Trần Quốc Huy – 2174802010294 – CNTT11)** 3](#_Toc191851166)

[**1.2 Kiểm thử chức năng đăng nhập ( Lê Văn Học – 2174802010422)** 19](#_Toc191851167)

[**1.3 Kiểm thử chức năng xem danh sách người dùng( Lê Văn Học – 2174802010422)** 30](#_Toc191851168)

**ĐỒ ÁN KIỂM THỬ TỰ ĐỘNG**

# **1.1 Kiểm thử chức năng Thêm học kỳ mới (Trần Quốc Huy – 2174802010294 – CNTT11)**

***1.1.1 Mô tả chức năng:***

A screenshot of a computer

AI-generated content may be incorrect.

Mô tả hoạt động của chức năng Thêm học kỳ mới:

* Người dùng có quyền BCN Khoa mở giao diện "Quản lý học kỳ".
* Nhấn nút "Thêm học kỳ mới" một popup form hiển thị ngay giữa màn hình.
* Nhập thông tin học kỳ:
* Năm bắt đầu (mặc định là năm hiện tại)
* Năm kết thúc (mặc định là năm kế tiếp)
* Học kỳ (tối đa 3 chữ số)
* Tuần bắt đầu (giá trị tối đa 52)
* Ngày bắt đầu
* Tiết tối đa trong 1 ngày (giới hạn 15)
* Lớp tối đa trong 1 tuần (giới hạn 30)
* Nhấn nút "Lưu":
* Nếu nhập thiếu dữ liệu: Hiển thị thông báo lỗi dưới từng trường dữ liệu vi phạm.
* Nếu nhập sai định dạng (ví dụ: nhập chữ vào ô số): Hiển thị lỗi ngay bên dưới trường.
* Nếu nhập trùng học kỳ đã tồn tại: Hiển thị thông báo lỗi "Học kỳ này đã tồn tại."
* Nếu nhập đúng dữ liệu:
* Lưu dữ liệu vào hệ thống.
* Hiển thị thông báo "Lưu thành công!" ở góc phải màn hình và tự động ẩn sau 5 giây.
* Đóng popup và cập nhật danh sách học kỳ mới.

***1.1.2 TestCase:***

|  |  |  |  |
| --- | --- | --- | --- |
| TC’ID | Tiêu đề | Bước | Kết quả mong đợi |
| TC\_THKM\_01 | Kiểm thử chức năng Không nhập dữ liệu | 1.Vàotrang https://cntttest.v  anlanguni.edu.vn:18081/P  hancong02/Term  2. Chọn Thêm học kỳ mới  3. Không nhập thông tin  4. Nhấn Lưu | 1. Hiện trang lên  2. Cho phép ấn vào mục Thêm học kỳ mới  3. Không nhập dữ liệu  4. Ấn Lưu  5. Hiện thông báo lỗi ở các trường chưa nhập thông tin |
| TC\_THKM\_02 | Kiểm thử chức năng Nhập sai định dạng dữ liệu | 1.Vàotrang https://cntttest.v  anlanguni.edu.vn:18081/P  hancong02/Term  2. Chọn Thêm học kỳ mới  3. Nhập sai định dạng dữ liệu  4. Ấn nút Lưu | 1. Hiện trang lên  2. Cho phép ấn vào mục Thêm học kỳ mới  3. Nhập sai định dạng dữ liệu  4. Ấn Lưu  5. Hiện thông báo lỗi |
| TC\_THKM\_03 | Kiểm thử chức năng Giá trị mặc định | 1.Vàotrang https://cntttest.v  anlanguni.edu.vn:18081/P  hancong02/Term  2. Chọn Thêm học kỳ mới | 1. Hiện trang lên  2. Cho phép ấn vào mục Thêm học kỳ mới  3. So sánh các trường Giá trị mặc định  4. Các trường Giá trị mặc định hiển thị đúng |
| TC\_THKM\_04 | Kiểm thử chức năng Nhập đúng thông tin | 1.Vàotrang https://cntttest.v  anlanguni.edu.vn:18081/P  hancong02/Term  2. Chọn thêm học kỳ mới  3. Nhập đầy đủ dữ liệu  4. Ấn Lưu | 1. Hiện trang lên  2. Cho phép ấn vào mục Thêm học kỳ mới  3. Nhập thông tin đúng  4. Ấn Lưu và không xuất hiện lỗi |
| TC\_THKM\_05 | Kiểm tra chức năng Lưu học kỳ thành công | 1.Vàotrang https://cntttest.v  anlanguni.edu.vn:18081/P  hancong02/Term  2. Chọn thêm học kỳ mới  3. Nhập đầy đủ dữ liệu  4. Ấn Lưu | 1. Hiện trang lên  2. Cho phép ấn vào mục Thêm học kỳ mới  3. Nhập thông tin đúng  4. Ấn Lưu  5. Hiển thị thông báo Lưu thành công |

***1.1.3 TestScripts:***

**\* TC\_THKM\_01**

package ThemHocKyMoi;

import org.testng.annotations.Test;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.chrome.ChromeDriver;

import org.testng.annotations.AfterTest;

import org.testng.annotations.BeforeTest;

import java.io.\*;

public class TC\_THKM\_01 {

WebDriver driver;

Login login = new Login();

*@BeforeTest*

public void TruocTest() throws InterruptedException, IOException {

System.*setProperty*("webdriver.chrome.driver", "C:\\chromedriver2.exe");

driver = new ChromeDriver();

login.TruocTest(driver);

}

*@Test*

public void KhongNhapDuLieu() throws InterruptedException {

Thread.*sleep*(6000);

driver.findElement(By.*cssSelector*("a[href='/Phancong02/Term']")).click();

Thread.*sleep*(2000);

driver.findElement(By.*cssSelector*(".dt-button.createNew.btn.btn-primary")).click();

Thread.*sleep*(2000);

driver.findElement(By.*cssSelector*("button[type='submit']")).click();

KiemTraThongBaoLoi("id-error", "Hoc ky");

KiemTraThongBaoLoi("start\_date-error", "Ngay bat dau");

}

public void KiemTraThongBaoLoi(String IdLoi, String Truong) {

WebElement ThongBaoLoi = driver.findElement(By.*id*(IdLoi));

if (ThongBaoLoi.isDisplayed()) {

System.***out***.println(

"PASS: Hiển thị thông báo lỗi khi không nhập dữ liệu " + Truong + " - " + ThongBaoLoi.getText());

} else {

System.***out***.println("FAIL: Không hiển thị thông báo lỗi khi không nhập dữ liệu " + Truong);

}

}

*@AfterTest*

public void SauTest() {

driver.close(); }}

***Kết quả:***

***A screenshot of a computer

AI-generated content may be incorrect.***

***Giải thích Code:***

* **options.setExperimentalOption("debuggerAddress", "127.0.0.1:9222")**: kết nối Selenium với Chrome đã mở sẵn thay vì khởi chạy một trình duyệt Chrome mới.
* **String oldURL = driver.getCurrentUrl():** Lưu URL hiện tại của trang web vào biến oldURL
* **public void KiemTraThongBaoLoi(String IdLoi, String Truong):** kiểm tra thông báo lỗi xuất hiện trên trang web
* **WebElement ThongBaoLoi = driver.findElement(By.*id*(IdLoi)):** Tìm phần tử trên trang web có id là IdLoi, sau đó gán vào biến ThongBaoLoi

**\*TC\_THKM\_02**

package ThemHocKyMoi;

import org.testng.annotations.Test;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.chrome.ChromeDriver;

import org.testng.annotations.AfterTest;

import org.testng.annotations.BeforeTest;

import java.io.\*;

public class TC\_THKM\_02 {

WebDriver driver;

Login login = new Login();

*@BeforeTest*

public void TruocTest() throws InterruptedException, IOException {

System.*setProperty*("webdriver.chrome.driver", "C:\\chromedriver2.exe");

driver = new ChromeDriver();

login.TruocTest(driver);

}

*@Test*

public void KiemTraDinhDangDuLieu() throws InterruptedException {

Thread.*sleep*(6000);

driver.findElement(By.*cssSelector*("a[href='/Phancong02/Term']")).click();

Thread.*sleep*(2000);

driver.findElement(By.*cssSelector*(".dt-button.createNew.btn.btn-primary")).click();

Thread.*sleep*(2000);

WebElement id = driver.findElement(By.*id*("id"));

id.sendKeys("1234");

WebElement week = driver.findElement(By.*cssSelector*("input#start\_week[value='1']"));

week.clear();

week.sendKeys(" ");

WebElement lesson = driver.findElement(By.*cssSelector*("input#max\_lesson[value='3']"));

lesson.clear();

lesson.sendKeys(" ");

WebElement mclass = driver.findElement(By.*cssSelector*("input#max\_class[value='1']"));

mclass.clear();

mclass.sendKeys(" ");

driver.findElement(By.*cssSelector*("button[type='submit']")).click();

KiemTraThongBaoLoi("id-error", "Hoc ky");

KiemTraThongBaoLoi("start\_week-error", "Tuan bat dau");

KiemTraThongBaoLoi("max\_lesson-error", "Tiet toi da");

KiemTraThongBaoLoi("max\_class-error", "Lop toi da");

}

public void KiemTraThongBaoLoi(String IdLoi, String Truong) {

WebElement ThongBaoLoi = driver.findElement(By.*id*(IdLoi));

if (ThongBaoLoi.isDisplayed()) {

System.***out***.println("PASS: Hiển thị thông báo lỗi khi nhập sai " + Truong + " - " + ThongBaoLoi.getText());

} else {

System.***out***.println("FAIL: Không hiển thị thông báo lỗi cho " + Truong);

}

}

*@AfterTest*

public void SauTest() {

driver.quit();

}

}

***Kết quả:***

***A screenshot of a computer

AI-generated content may be incorrect.***

***Giải thích code:***

* **By.cssSelector:** tìm phần tử HTML trên trang web
* **public void KiemTraThongBaoLoi(String IdLoi, String Truong):** kiểm tra có thông báo lỗi hiển thị không khi người dùng nhập sai dữ liệu
* **WebElement ThongBaoLoi = driver.findElement(By.id(IdLoi)):** Tìm phần tử trên trang web bằng id và gán nó vào biến ThongBaoLoi

**\*TC\_THKM\_03**

package ThemHocKyMoi;

import org.testng.annotations.Test;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.chrome.ChromeDriver;

import org.testng.annotations.AfterTest;

import org.testng.annotations.BeforeTest;

import java.io.\*;

public class TC\_THKM\_03 {

WebDriver driver;

Login login = new Login();

*@BeforeTest*

public void TruocTest() throws InterruptedException, IOException {

System.*setProperty*("webdriver.chrome.driver", "C:\\chromedriver2.exe");

driver = new ChromeDriver();

login.TruocTest(driver);

}

*@Test*

public void GiaTriMacDinh() throws InterruptedException {

Thread.*sleep*(2000);

driver.findElement(By.*cssSelector*("a[href='/Phancong02/Term']")).click();

Thread.*sleep*(1000);

driver.findElement(By.*cssSelector*(".dt-button.createNew.btn.btn-primary")).click();

Thread.*sleep*(1000);

WebElement NamBatDau = driver.findElement(By.*cssSelector*(".select2-selection\_\_rendered[title='2025']"));

WebElement NamKetThuc = driver.findElement(By.*cssSelector*(".select2-selection\_\_rendered[title='2026']"));

String namBatDauText = NamBatDau.getText();

String namKetThucText = NamKetThuc.getText();

if (namBatDauText.equals("2025") && namKetThucText.equals("2026")) {

System.***out***.println("PASS: Hiện đúng giá trị mặc định");

} else {

System.***out***.println("FAIL: Hiện sai giá trị mặc định");

}

}

*@AfterTest*

public void SauTest() {

driver.close();

}

}

***Kết quả:***

**A screenshot of a computer

AI-generated content may be incorrect.**

**\*TC\_THKM\_04**

package ThemHocKyMoi;

import org.testng.annotations.Test;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.chrome.ChromeDriver;

import org.testng.annotations.AfterTest;

import org.testng.annotations.BeforeTest;

import java.io.\*;

public class TC\_THKM\_04 {

WebDriver driver;

Login login = new Login();

*@BeforeTest*

public void TruocTest() throws InterruptedException, IOException {

System.*setProperty*("webdriver.chrome.driver", "C:\\chromedriver2.exe");

driver = new ChromeDriver();

login.TruocTest(driver);

}

*@Test*

public void NhapDungThongTin() throws InterruptedException {

Thread.*sleep*(6000);

driver.findElement(By.*cssSelector*("a[href='/Phancong02/Term']")).click();

Thread.*sleep*(2000);

driver.findElement(By.*cssSelector*(".dt-button.createNew.btn.btn-primary")).click();

Thread.*sleep*(2000);

WebElement HocKy = driver.findElement(By.*id*("id"));

WebElement TuanBatDau = driver.findElement(By.*cssSelector*("input#start\_week[value='1']"));

WebElement TietToiDa = driver.findElement(By.*cssSelector*("input#max\_lesson[value='3']"));

WebElement LopToiDa = driver.findElement(By.*cssSelector*("input#max\_class[value='1']"));

HocKy.sendKeys("221");

TuanBatDau.clear();

TuanBatDau.sendKeys("10");

TietToiDa.clear();

TietToiDa.sendKeys("10");

LopToiDa.clear();

LopToiDa.sendKeys("10");

driver.findElement(By.*cssSelector*("button[type='submit']")).click();

if (!driver.findElements(By.*className*("error")).isEmpty()) {

if (driver.findElements(By.*cssSelector*("input#start\_week + .error")).size() > 0) {

System.***out***.println("FAIL: Tuần bắt đầu không hợp lệ.");

} else {

System.***out***.println("PASS: Tuần bắt đầu hợp lệ.");

}

if (driver.findElements(By.*cssSelector*("input#max\_lesson + .error")).size() > 0) {

System.***out***.println("FAIL: Tiết tối đa không hợp lệ.");

} else {

System.***out***.println("PASS: Tiết tối đa hợp lệ.");

}

if (driver.findElements(By.*cssSelector*("input#max\_class + .error")).size() > 0) {

System.***out***.println("FAIL: Lớp tối đa không hợp lệ.");

} else {

System.***out***.println("PASS: Lớp tối đa hợp lệ.");}

System.***out***.println("FAIL: Nhập không đúng các thông tin khác");

} else {

System.***out***.println("PASS: Nhập đúng thông tin");}}

*@AfterTest*

public void SauTest() {

driver.close();}}

***Kết quả:***

***A screenshot of a computer

AI-generated content may be incorrect.***

***Giải thích code:***

* **driver.findElements(By.className("error")) :** Tìm tất cả phần tử HTML có class là "error".
* **.isEmpty() :** Kiểm tra danh sách phần tử tìm được có rỗng hay không

**\*TC\_THKM\_05**

package ThemHocKyMoi;

import org.testng.annotations.Test;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.chrome.ChromeDriver;

import org.testng.annotations.AfterTest;

import org.testng.annotations.BeforeTest;

import java.io.\*;

public class TC\_THKM\_05 {

WebDriver driver;

Login login = new Login();

*@BeforeTest*

public void TruocTest() throws InterruptedException, IOException {

System.*setProperty*("webdriver.chrome.driver", "C:\\chromedriver2.exe");

driver = new ChromeDriver();

login.TruocTest(driver);

}

*@Test*

public void KiemTraLuuThanhCong() throws InterruptedException {

Thread.*sleep*(2000);

driver.findElement(By.*cssSelector*("a[href='/Phancong02/Term']")).click();

Thread.*sleep*(1000);

driver.findElement(By.*cssSelector*(".dt-button.createNew.btn.btn-primary")).click();

Thread.*sleep*(1000);

WebElement HocKy = driver.findElement(By.*id*("id"));

WebElement TuanBatDau = driver.findElement(By.*cssSelector*("input#start\_week[value='1']"));

WebElement TietToiDa = driver.findElement(By.*cssSelector*("input#max\_lesson[value='3']"));

WebElement LopToiDa = driver.findElement(By.*cssSelector*("input#max\_class[value='1']"));

HocKy.sendKeys("221");

TuanBatDau.clear();

TuanBatDau.sendKeys("10");

TietToiDa.clear();

TietToiDa.sendKeys("10");

LopToiDa.clear();

LopToiDa.sendKeys("10");

driver.findElement(By.*cssSelector*("button[type='submit']")).click();

if (driver.findElements(By.*className*("error")).isEmpty()) {

WebElement ThongBaoThanhCong = driver.findElement(By.*xpath*("//div[contains(text(),'Lưu thành công!')]"));

if (ThongBaoThanhCong.isDisplayed()) {

System.***out***.println("PASS: Hiển thị thông báo Lưu học kỳ thành công");

} else {

System.***out***.println("FAIL: Không hiển thị thông báo Lưu thành công");

}

} else {

System.***out***.println("FAIL: Có lỗi khi lưu học kỳ");

}

}

*@AfterTest*

public void SauTest() {

driver.close();

}

}

***Kết quả:***

***A screenshot of a computer

AI-generated content may be incorrect.***

***Giải thích code:***

* **if (ThongBaoThanhCong.isDisplayed()):** Kiểm tra xem một phần tử trên trang web có đang hiển thị hay không

# **1.2 Kiểm thử chức năng đăng nhập ( Lê Văn Học – 2174802010422)**

**1.2.1 Mô tả chức năng:**

A screenshot of a computer

AI-generated content may be incorrect.

**1.2.2 TestCase:**

|  |  |  |  |
| --- | --- | --- | --- |
| **TC’ID** | Tiêu đề | Bước thực hiện | Kết quả mong đợi |
| TC\_LOGIN\_01 | Kiểm thử đăng nhập thành công | 1. Mở trang đăng nhập.  2. Nhập tài khoản hợp lệ.  3. Nhấn Login. | 1. Hệ thống chuyển hướng đến trang chính sau khi đăng nhập thành công. |
| TC\_LOGIN\_02 | Kiểm thử đăng nhập với sai tài khoản | 1. Mở trang đăng nhập.  2. Nhập tài khoản không tồn tại.  3. Nhấn Login. | 1. Hiển thị thông báo lỗi "Tài khoản không tồn tại". |
| TC\_LOGIN\_03 | Kiểm thử đăng nhập với sai mật khẩu | 1. Mở trang đăng nhập.  2. Nhập tài khoản đúng nhưng sai mật khẩu.  3. Nhấn Login. | 1. Hiển thị thông báo lỗi "Sai mật khẩu". |

**1.2.1 TesScripts:**

**\*TC\_BaseTest:**

package LoginTest;

import org.openqa.selenium.\*;

import org.openqa.selenium.chrome.ChromeDriver;

import org.openqa.selenium.support.ui.ExpectedConditions;

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

import java.time.Duration;

public class BaseTest {

protected WebDriver driver;

protected WebDriverWait wait;

public BaseTest() {

System.setProperty("webdriver.chrome.driver", "D:\\java-2024-12\\chromedriver-win64\\chromedriver.exe");

driver = new ChromeDriver();

wait = new WebDriverWait(driver, Duration.ofSeconds(60));

}

public void openLoginPage() {

driver.get("https://cntttest.vanlanguni.edu.vn:18081/Phancong02/Account/Login");

WebElement loginButton = wait.until(ExpectedConditions.elementToBeClickable(By.id("OpenIdConnect")));

loginButton.click();

wait.until(ExpectedConditions.urlContains("login.microsoftonline.com"));

}

public void closeBrowser() {

driver.quit();

}

}

**Giải thích code:**

* **Lớp BaseTest** giúp thiết lập Selenium WebDriver.
* **openLoginPage()** tự động mở trang đăng nhập và nhấn vào OpenID.
* **closeBrowser()** để đóng trình duyệt.
* Có thể **kế thừa BaseTest** để viết các bài kiểm thử khác nhau mà không cần lặp lại mã.

**\*TC\_InvalidPasswordTest:**

package LoginTest;

import org.openqa.selenium.By;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.support.ui.ExpectedConditions;

public class InvalidPasswordTest extends BaseTest {

public static void main(String[] args) {

InvalidPasswordTest test = new InvalidPasswordTest();

test.runTest();

}

public void runTest() {

System.out.println("🔍 Kiểm thử trường hợp sai mật khẩu...");

openLoginPage();

WebElement emailInput = wait.until(ExpectedConditions.visibilityOfElementLocated(By.id("i0116")));

emailInput.clear();

emailInput.sendKeys("hoc.2174802010422@vanlanguni.vn");

WebElement nextButton = wait.until(ExpectedConditions.elementToBeClickable(By.id("idSIButton9")));

nextButton.click();

WebElement passwordInput = wait.until(ExpectedConditions.visibilityOfElementLocated(By.id("i0118")));

passwordInput.sendKeys("sai\_mat\_khau");

WebElement submitButton = wait.until(ExpectedConditions.elementToBeClickable(By.id("idSIButton9")));

submitButton.click();

try {

WebElement errorMessage = wait.until(ExpectedConditions.visibilityOfElementLocated(By.id("passwordError")));

System.out.println("⚠️ Thông báo lỗi xuất hiện: " + errorMessage.getText());

} catch (Exception e) {

System.err.println("⚠️ Không tìm thấy thông báo lỗi cho mật khẩu sai");

}

closeBrowser();}}

**Kết quả:**

A screenshot of a computer

AI-generated content may be incorrect.

**Giải thích code:**

Lớp **InvalidPasswordTest** kế thừa từ BaseTest và thực hiện kiểm thử tình huống nhập sai mật khẩu khi đăng nhập:

* **openLoginPage():** Mở trang đăng nhập.
* Nhập email hợp lệ và tiếp tục.
* Nhập mật khẩu sai và gửi đi.
* Kiểm tra thông báo lỗi xuất hiện khi nhập sai mật khẩu.
* Đóng trình duyệt sau khi kiểm thử.

Mục đích của test case này là xác minh hệ thống xử lý lỗi khi người dùng nhập sai mật khẩu.

**\*TC\_InvalidUsernameTest**

package LoginTest;

import org.openqa.selenium.By;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.support.ui.ExpectedConditions;

public class InvalidUsernameTest extends BaseTest {

public static void main(String[] args) {

InvalidUsernameTest test = new InvalidUsernameTest();

test.runTest();

}

public void runTest() {

System.out.println("🔍 Kiểm thử trường hợp sai tên đăng nhập...");

openLoginPage();

WebElement emailInput = wait.until(ExpectedConditions.visibilityOfElementLocated(By.id("i0116")));

emailInput.clear();

emailInput.sendKeys("sai\_email@vanlanguni.vn");

WebElement nextButton = wait.until(ExpectedConditions.elementToBeClickable(By.id("idSIButton9")));

nextButton.click();

try {

WebElement errorMessage = wait.until(ExpectedConditions.visibilityOfElementLocated(By.id("usernameError")));

System.out.println("⚠️ Thông báo lỗi xuất hiện: " + errorMessage.getText());

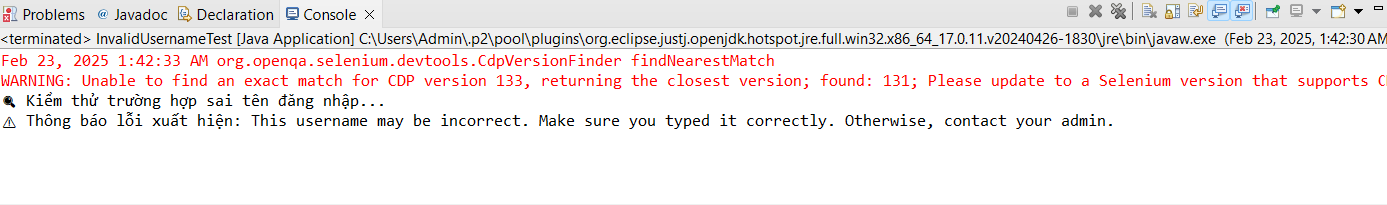
} catch (Exception e) {

System.err.println("⚠️ Không tìm thấy thông báo lỗi cho tên đăng nhập sai");

}

closeBrowser(); }}

**Kết quả:**



**Giải thích code:**

Lớp **InvalidUsernameTest** kế thừa từ BaseTest và thực hiện kiểm thử tình huống nhập sai tên đăng nhập khi đăng nhập:

* **openLoginPage()**: Mở trang đăng nhập.
* Nhập email không hợp lệ và tiếp tục.
* Kiểm tra thông báo lỗi xuất hiện khi nhập sai email.
* Đóng trình duyệt sau khi kiểm thử.

Mục đích của test case này là xác minh hệ thống xử lý lỗi khi người dùng nhập sai tên đăng nhập.

**\*TC\_ValidLoginTest:**

package LoginTest;

import org.openqa.selenium.By;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.support.ui.ExpectedConditions;

public class ValidLoginTest extends BaseTest {

public static void main(String[] args) {

ValidLoginTest test = new ValidLoginTest();

test.runTest();

}

public void runTest() {

System.out.println("✅ Kiểm thử đăng nhập đúng...");

openLoginPage();

WebElement emailInput = wait.until(ExpectedConditions.visibilityOfElementLocated(By.id("i0116")));

emailInput.clear();

emailInput.sendKeys("hoc.2174802010422@vanlanguni.vn");

WebElement nextButton = wait.until(ExpectedConditions.elementToBeClickable(By.id("idSIButton9")));

nextButton.click();

WebElement passwordInput = wait.until(ExpectedConditions.visibilityOfElementLocated(By.id("i0118")));

passwordInput.sendKeys("VLU@2174802010422");

WebElement submitButton = wait.until(ExpectedConditions.elementToBeClickable(By.id("idSIButton9")));

submitButton.click();

try {

wait.until(ExpectedConditions.urlContains("cntttest.vanlanguni.edu.vn"));

System.out.println("🎉 Đăng nhập thành công!");

} catch (Exception e) {

System.err.println("❌ Lỗi khi kiểm thử đăng nhập đúng: " + e.getMessage());}}}

**Kết quả:**

A screenshot of a computer

AI-generated content may be incorrect.

**Giải thích code:**

Lớp **ValidLoginTest** kế thừa từ BaseTest và kiểm thử trường hợp đăng nhập hợp lệ:

* Mở trang đăng nhập (openLoginPage()).
* Nhập đúng email và mật khẩu, sau đó nhấn tiếp tục.
* Xử lý màn hình "Stay signed in?" nếu xuất hiện.
* Kiểm tra đăng nhập thành công bằng cách xác minh URL.
* Thông báo kết quả kiểm thử.

Mục đích: Đảm bảo hệ thống cho phép đăng nhập với thông tin hợp lệ

**\*TC\_LoginTestSuite:**

package LoginTest;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.chrome.ChromeDriver;

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

import java.time.Duration;

public class LoginTestSuite {

public static void main(String[] args) {

System.out.println("🚀 Bắt đầu kiểm thử đăng nhập...");

// Khởi tạo WebDriver (Chrome)

System.setProperty("webdriver.chrome.driver", "path/to/chromedriver"); // Thay đổi đường dẫn đúng

WebDriver driver = new ChromeDriver();

// Thiết lập WebDriverWait

WebDriverWait wait = new WebDriverWait(driver, Duration.ofSeconds(10));

try {

new InvalidUsernameTest().runTest();

new InvalidPasswordTest().runTest();

new ValidLoginTest().runTest();

} finally {

// Đóng trình duyệt sau khi hoàn thành test

driver.quit();

}

}

}

**Giải thích code:**

Lớp **LoginTestSuite** tập hợp và chạy tất cả các test case liên quan đến đăng nhập:

* Khởi tạo WebDriver để chạy trình duyệt Chrome.
* Thiết lập WebDriverWait để chờ đợi phần tử xuất hiện.
* Chạy lần lượt các test case:
  + InvalidUsernameTest: Kiểm thử đăng nhập với tên đăng nhập sai.
  + InvalidPasswordTest: Kiểm thử đăng nhập với mật khẩu sai.
  + ValidLoginTest: Kiểm thử đăng nhập thành công.
* Đóng trình duyệt sau khi chạy xong.

Mục đích: Tự động hóa toàn bộ quá trình kiểm thử đăng nhập trong một lần chạy.

# **1.3 Kiểm thử chức năng xem danh sách người dùng( Lê Văn Học – 2174802010422)**

**1.3.1 Mô tả chức năng:**

**A screenshot of a computer

AI-generated content may be incorrect.1.3.2 TestCase:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TC’ID** | **Tiêu đề** | **Bước thực hiện** | |  | | --- | |  |  |  | | --- | | **Kết quả mong đợi** | |
| TC\_VIEW\_01 | Kiểm thử truy cập danh sách người dùng khi đã đăng nhập | 1. Đăng nhập hệ thống.  2. Điều hướng đến trang danh sách người dùng. | 1. Danh sách người dùng hiển thị đầy đủ. |
| TC\_VIEW\_02 | Kiểm thử truy cập danh sách người dùng khi chưa đăng nhập | 1. Truy cập trực tiếp vào trang danh sách người dùng mà không đăng nhập. | 1. Hệ thống chuyển hướng đến trang đăng nhập. |

**\*TC\_ViewUsersTest**

package UserManagermentTest;

import org.openqa.selenium.\*;

import org.openqa.selenium.chrome.ChromeDriver;

import org.openqa.selenium.support.ui.ExpectedConditions;

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

import java.time.Duration;

public class ViewUsersTest {

private WebDriver driver;

private WebDriverWait wait;

public ViewUsersTest() {

System.setProperty("webdriver.chrome.driver", "D:\\java-2024-12\\chromedriver-win64\\chromedriver.exe");

this.driver = new ChromeDriver();

this.wait = new WebDriverWait(driver, Duration.ofSeconds(60));

}

public void login() {

try {

System.out.println("🔑 Đang thực hiện đăng nhập...");

driver.get("https://cntttest.vanlanguni.edu.vn:18081/Phancong02/Account/Login");

// Nhấn vào nút đăng nhập OpenID

WebElement loginButton = wait.until(ExpectedConditions.elementToBeClickable(By.id("OpenIdConnect")));

loginButton.click();

// Chuyển sang trang Microsoft Login

wait.until(ExpectedConditions.urlContains("login.microsoftonline.com"));

// Nhập email

WebElement emailInput = wait.until(ExpectedConditions.visibilityOfElementLocated(By.id("i0116")));

emailInput.sendKeys("hoc.2174802010422@vanlanguni.vn");

driver.findElement(By.id("idSIButton9")).click();

// Đợi trang tải mật khẩu

Thread.sleep(2000);

// Nhập mật khẩu

WebElement passwordInput = wait.until(ExpectedConditions.visibilityOfElementLocated(By.id("i0118")));

passwordInput.sendKeys("VLU@2174802010422");

driver.findElement(By.id("idSIButton9")).click();

// Kiểm tra trang "Stay signed in?"

try {

WebElement staySignedInButton = wait.until(ExpectedConditions.elementToBeClickable(By.id("idSIButton9")));

staySignedInButton.click();

} catch (Exception e) {

System.out.println("ℹ Không có trang xác nhận 'Stay signed in?'. Tiếp tục...");

}

// Kiểm tra đăng nhập thành công

wait.until(ExpectedConditions.urlContains("cntttest.vanlanguni.edu.vn"));

System.out.println("✅ Đăng nhập thành công!");

} catch (Exception e) {

System.err.println("❌ Lỗi khi đăng nhập: " + e.getMessage());

e.printStackTrace();

}

}

public void viewUsers() {

try {

System.out.println("📋 Đang kiểm thử chức năng xem danh sách người dùng...");

// Chuyển đến trang danh sách người dùng

driver.get("https://cntttest.vanlanguni.edu.vn:18081/Phancong02/User");

System.out.println("✅ Trang danh sách người dùng đã tải thành công!");

} finally {

driver.quit(); // Đóng trình duyệt sau khi hoàn thành

}

}

public void run() {

login();

viewUsers();

} public static void main(String[] args) {

ViewUsersTest test = new ViewUsersTest();

test.run();

}}

Kết quả:

A screenshot of a computer

AI-generated content may be incorrect.

Giải thích code:

**Lớp ViewUsersTest** thực hiện kiểm thử chức năng xem danh sách người dùng trong hệ thống.

* **login()** – Kiểm thử quá trình đăng nhập:
  + Mở trang đăng nhập và chọn phương thức OpenID.
  + Nhập email và mật khẩu để đăng nhập.
  + Xử lý trang "Stay signed in?" nếu xuất hiện.
  + Kiểm tra đăng nhập thành công.
* **viewUsers()** – Kiểm thử chức năng xem danh sách người dùng:
  + Điều hướng đến trang danh sách người dùng.
  + Kiểm tra xem trang có tải thành công không.
* **run()** – Gọi lần lượt hai bước trên.
* **main()** – Khởi chạy toàn bộ kiểm thử.

Mục đích: Xác minh người dùng có thể đăng nhập thành công và truy cập danh sách người dùng trong hệ thống. 🚀