



# UNIVERSITÀ DEGLI STUDI DI SALERNO

Corso di Laurea in Informatica, a.a. 2021-22

Progetto del corso di Ingegneria del Software

prof. A. De Lucia, dott. F. Pecorelli

Repository GitHub: [/is-shodan-21-22/](#)

# SHODAN

- › Project Proposal
- › Problem Statement
- › Requirement Analysis Document
- › System Design Document
- › Object Design Document
- › Test Plan
- › Test Execution Report
- › Test Summary Report**
- › Teamwork Report

# Shodan [IS-2021/2022-DE LUCIA]

## Partecipanti del progetto

Nome	Matricola	E-mail
Antonio Gravino	05121 07161	a.gravino3@studenti.unisa.it
Dario Trinchese	05121 07479	d.trinchese2@studenti.unisa.it
Raffaele Zheng	05121 09015	r.zheng@studenti.unisa.it
Carmine Fabbri	05121 07353	c.fabbri@studenti.unisa.it
Carmine Napolitano	05121 06417	c.napolitano44@studenti.unisa.it

Questo documento tratta esclusivamente l'Unit Testing del progetto.

Per ulteriore documentazione valida ai fini dell'esame, consultare la repository [/deliverables/](#).

### Premessa

Il testing di unità consiste nel testare le singole unità del software del sistema suddivisi in sottosistemi individuali, con l'obiettivo di verificare che ogni sottosistema sia stato codificato correttamente. Il test verrà effettuato utilizzando *JUnit*, con l'ausilio di *Selenium*.

### Relazione con altri documenti

Il presente documento si basa sui documenti Test Plan e Test Case relativi al progetto Shodan.

## Librerie, driver e risorse esterne utilizzate

```
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;
```

```
src
├── main
└── test \ java
    ├── AggiornaTitoloTest.java
    ├── AggiungiArticoloTest.java
    ├── AggiungiCartaTest.java
    ├── AggiungiTitoloTest.java
    ├── LoginTest.java
    ├── ModificaEmailTest.java
    ├── ModificaPasswordTest.java
    ├── PagamentoTest.java
    ├── RegistrazioneTest.java
    ├── RicaricaSaldoTest.java
    ├── RimuoviArticoloTest.java
    └── RimuoviTitoloTest.java
```

```

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

    @Before
    public void setUp() {
        System.setProperty("webdriver.chrome.driver", "../commons/driver/chromedriver.exe");
        driver = new ChromeDriver();
        js = (JavascriptExecutor) driver;
        vars = new HashMap<String, Object> ();
    }

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

    @Test
    public void aggiornaTitoloFormatoIDErrato() {
        driver.get("http://localhost:8080/shodan_maven/");
        driver.manage().window().setSize(new Dimension(1898, 1018));
        driver.findElement(By.id("login-username")).click();
        driver.findElement(By.id("login-username")).sendKeys("admin");
        driver.findElement(By.id("login-password")).click();
        driver.findElement(By.id("login-password")).sendKeys("123");
        driver.findElement(By.id("login-submit")).click();
        {
            WebDriverWait wait = new WebDriverWait(driver, 30);
            wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
        }
        driver.findElement(By.id("update-game-id")).sendKeys("abc");
        driver.findElement(By.id("update-game-price")).click();
        driver.findElement(By.id("update-game-price")).sendKeys("55");
        driver.findElement(By.cssSelector(".button:nth-child(5)")).click();
        assertThat(driver.findElement(By.id("update-game-result")).getText(), is("Formato ID non valido"));
    }

    @Test
    public void aggiornaTitoloFormatoPrezzoErrato() {
        driver.get("http://localhost:8080/shodan_maven/");
        driver.manage().window().setSize(new Dimension(1898, 1018));
        driver.findElement(By.id("login-username")).click();
        driver.findElement(By.id("login-username")).sendKeys("admin");
        driver.findElement(By.id("login-password")).click();
        driver.findElement(By.id("login-password")).sendKeys("123");
    }
}

```

```

driver.findElement(By.id("login-submit")).click();
{
    WebDriverWait wait = new WebDriverWait(driver, 30);
    wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
}
driver.findElement(By.id("update-game-id")).sendKeys("3");
driver.findElement(By.id("update-game-price")).click();
driver.findElement(By.id("update-game-price")).sendKeys("abc");
driver.findElement(By.cssSelector(".button:nth-child(5)")).click();
assertThat(driver.findElement(By.id("update-game-result")).getText(), is("Formato prezzo non valido"));
}

@Test
public void aggiornaTitoloDMancante() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1898, 1018));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("admin");
    driver.findElement(By.id("login-password")).click();
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-submit")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
    }
    driver.findElement(By.id("update-game-price")).click();
    driver.findElement(By.id("update-game-price")).sendKeys("55");
    driver.findElement(By.cssSelector(".button:nth-child(5)")).click();
    assertThat(driver.findElement(By.id("update-game-result")).getText(), is("ID: campo obbligatorio"));
}

@Test
public void aggiornaTitoloOK() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1898, 1018));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("admin");
    driver.findElement(By.id("login-password")).click();
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-form")).click();
    driver.findElement(By.id("login-submit")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
    }
}

```

```

    }

    driver.findElement(By.id("update-game-id")).click();
    driver.findElement(By.id("update-game-id")).sendKeys("2");
    driver.findElement(By.id("update-game-price")).sendKeys("100");
    driver.findElement(By.cssSelector(".button:nth-child(5)")).click();
    assertThat(driver.findElement(By.id("update-game-result")).getText(), is("Gioco aggiornato con
    successo!"));
}

@Test
public void aggiornaTitoloPrezzoMancante() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1898, 1018));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("admin");
    driver.findElement(By.id("login-password")).click();
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-submit")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
    }
    driver.findElement(By.id("update-game-id")).click();
    driver.findElement(By.id("update-game-id")).sendKeys("3");
    driver.findElement(By.cssSelector(".button:nth-child(5)")).click();
    assertThat(driver.findElement(By.id("update-game-result")).getText(), is("Prezzo: campo obbligatorio"));
}

@Test
public void aggiornaTitoloPrezzoNegativo() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1898, 1018));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("admin");
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-submit")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
    }
    driver.findElement(By.id("update-game-id")).click();
    driver.findElement(By.id("update-game-id")).sendKeys("3");
    driver.findElement(By.id("update-game-price")).sendKeys("-10");
    driver.findElement(By.cssSelector(".button:nth-child(5)")).click();

```

```

        assertThat(driver.findElement(By.id("update-game-result")).getText(), is("Prezzo negativo"));
    }

    @Test
    public void aggiornaTitoloTitoloInesistente() {
        driver.get("http://localhost:8080/shodan_maven/");
        driver.manage().window().setSize(new Dimension(1898, 1018));
        driver.findElement(By.id("login-username")).click();
        driver.findElement(By.id("login-username")).sendKeys("admin");
        driver.findElement(By.id("login-password")).click();
        driver.findElement(By.id("login-password")).sendKeys("123");
        driver.findElement(By.id("login-submit")).click();
        {
            WebDriverWait wait = new WebDriverWait(driver, 30);
            wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
        }
        driver.findElement(By.id("update-game-id")).click();
        driver.findElement(By.id("update-game-id")).sendKeys("999");
        driver.findElement(By.id("update-game-price")).sendKeys("55");
        driver.findElement(By.cssSelector(".button:nth-child(5)")).click();
        assertThat(driver.findElement(By.id("update-game-result")).getText(), is("Titolo non presente"));
    }
}

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

    @Before
    public void setUp() {
        System.setProperty("webdriver.chrome.driver", "../commons/driver/chromedriver.exe");
        driver = new ChromeDriver();
        js = (JavascriptExecutor) driver;
        vars = new HashMap<String, Object>();
    }

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

    @Test
    public void aggiungiArticoloContenutoMancante() {
        driver.get("http://localhost:8080/shodan_maven/");
        driver.manage().window().setSize(new Dimension(1898, 1018));
    }
}

```



```

driver.findElement(By.id("login-username")).click();
driver.findElement(By.id("login-username")).sendKeys("admin");
driver.findElement(By.id("login-password")).click();
driver.findElement(By.id("login-password")).sendKeys("123");
driver.findElement(By.id("login-submit")).click();
{
    WebDriverWait wait = new WebDriverWait(driver, 30);
    wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
}
driver.findElement(By.id("roles-link")).click();
driver.findElement(By.cssSelector(".role-item:nth-child(3)")).click();
driver.findElement(By.id("add-article-title")).sendKeys("Titolo");
driver.findElement(By.id("add-article-shortTitle")).sendKeys("Sottotitolo");
driver.findElement(By.id("add-article-button")).click();
assertThat(driver.findElement(By.id("add-article-result")).getText(), is("Contenuto: campo obbligatorio"));
}

@Test
public void aggiungiArticoloSottotitoloMancante() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1898, 1018));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("admin");
    driver.findElement(By.id("login-password")).click();
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-submit")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
    }
    driver.findElement(By.id("roles-link")).click();
    driver.findElement(By.cssSelector(".role-item:nth-child(3)")).click();
    driver.findElement(By.id("add-article-title")).sendKeys("Titolo");
    driver.findElement(By.id("article-description")).sendKeys("Contenuto");
    driver.findElement(By.id("add-article-button")).click();
    assertThat(driver.findElement(By.id("add-article-result")).getText(), is("Sottotitolo: campo obbligatorio"));
}

@Test
public void aggiungiArticoloTitoloMancante() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1898, 1018));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("admin");

```

```

driver.findElement(By.id("login-password")).click();
driver.findElement(By.id("login-password")).sendKeys("123");
driver.findElement(By.id("login-submit")).click();
{
    WebDriverWait wait = new WebDriverWait(driver, 30);
    wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
}
driver.findElement(By.id("roles-link")).click();
driver.findElement(By.cssSelector(".role-item:nth-child(3)")).click();
driver.findElement(By.id("add-article-shortTitle")).click();
driver.findElement(By.id("add-article-shortTitle")).sendKeys("Sottotitolo");
driver.findElement(By.id("article-description")).sendKeys("Contenuto");
driver.findElement(By.id("add-article-button")).click();
assertThat(driver.findElement(By.id("add-article-result")).getText(), is("Titolo: campo obbligatorio"));
}

@Test
public void aggiungiArticoloOK() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1898, 1018));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("admin");
    driver.findElement(By.id("login-password")).click();
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-submit")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
    }
    driver.findElement(By.id("roles-link")).click();
    driver.findElement(By.cssSelector(".role-item:nth-child(3)")).click();
    driver.findElement(By.id("add-article-title")).click();
    driver.findElement(By.id("add-article-title")).sendKeys("Titolo");
    driver.findElement(By.id("add-article-shortTitle")).sendKeys("Sottotitolo");
    driver.findElement(By.id("article-description")).sendKeys("Contenuto");
    driver.findElement(By.id("add-article-button")).click();
    {
        WebElement element = driver.findElement(By.id("add-article-button"));
        Actions builder = new Actions(driver);
        builder.moveToElement(element).perform();
    }
    assertThat(driver.findElement(By.id("add-article-result")).getText(), is("Articolo aggiunto con successo!"));
}

```

```
}
```

```
public class AggiungiCartaTest {  
    private WebDriver driver;  
    private Map<String, Object> vars;  
    JavascriptExecutor js;  
    @Before  
    public void setUp() {  
        System.setProperty("webdriver.chrome.driver", "../commons/driver/chromedriver.exe");  
        driver = new ChromeDriver();  
        js = (JavascriptExecutor) driver;  
        vars = new HashMap<String, Object>();  
    }  
    @After  
    public void tearDown() {  
        driver.quit();  
    }  
    @Test  
    public void aggiungiCartaCVVMancante() {  
        driver.get("http://localhost:8080/shodan_maven/");  
        driver.manage().window().setSize(new Dimension(1898, 1018));  
        driver.findElement(By.id("login-username")).click();  
        driver.findElement(By.id("login-username")).sendKeys("slave");  
        driver.findElement(By.id("login-password")).sendKeys("123");  
        driver.findElement(By.id("login-submit")).sendKeys(Keys.ENTER);  
        {  
            WebDriverWait wait = new WebDriverWait(driver, 30);  
            wait.until(ExpectedConditions.presenceOfElementLocated(By.id("settings-link")));  
        }  
        driver.findElement(By.id("settings-link")).click();  
        {  
            WebDriverWait wait = new WebDriverWait(driver, 30);  
            wait.until(ExpectedConditions.presenceOfElementLocated(By.id("add-card")));  
        }  
        driver.findElement(By.id("add-card")).click();  
        driver.findElement(By.cssSelector(".credit-card-add")).click();  
        driver.findElement(By.cssSelector(".card-type")).click();  
        driver.findElement(By.cssSelector(".card-type")).sendKeys("VISA");  
        driver.findElement(By.cssSelector(".card-number")).sendKeys("1111222233334444");  
        driver.findElement(By.cssSelector(".card-owner")).sendKeys("Slaver Test");  
        driver.findElement(By.cssSelector(".card-due")).sendKeys("2024-01-01");  
        driver.findElement(By.id("new-card-form-submit")).click();  
    }  
}
```

```

    assertThat(driver.findElement(By.cssSelector(".card-tip")).getText(), is("CVV: campo obbligatorio"));
}

@Test
public void aggiungiCartaCartaEsistente() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1898, 1018));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("slave");
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-password")).sendKeys(Keys.ENTER);
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("settings-link")));
    }
    driver.findElement(By.id("settings-link")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("add-card")));
    }
    driver.findElement(By.id("add-card")).click();
    driver.findElement(By.cssSelector(".credit-card-add")).click();
    driver.findElement(By.cssSelector(".card-type")).click();
    driver.findElement(By.cssSelector(".card-type")).sendKeys("MASTERCARD");
    driver.findElement(By.cssSelector(".card-number")).sendKeys("8219671281263413");
    driver.findElement(By.cssSelector(".card-owner")).sendKeys("Slave Tester");
    driver.findElement(By.cssSelector(".card-cvv")).sendKeys("123");
    driver.findElement(By.cssSelector(".card-due")).sendKeys("2025-01-01");
    driver.findElement(By.id("new-card-form-submit")).click();
    assertThat(driver.findElement(By.cssSelector(".card-tip")).getText(), is("Carta gia\\\\" esistente"));
}

@Test
public void aggiungiCartaCartaScaduta() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1898, 1018));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("slave");
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-password")).sendKeys(Keys.ENTER);
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("settings-link")));
    }
}

```

```

driver.findElement(By.id("settings-link")).click();
{
    WebDriverWait wait = new WebDriverWait(driver, 30);
    wait.until(ExpectedConditions.presenceOfElementLocated(By.id("add-card")));
}
driver.findElement(By.id("add-card")).click();
driver.findElement(By.cssSelector(".credit-card-add")).click();
driver.findElement(By.cssSelector(".card-type")).click();
driver.findElement(By.cssSelector(".card-type")).sendKeys("VISA");
driver.findElement(By.cssSelector(".card-number")).sendKeys("1111222233334444");
driver.findElement(By.cssSelector(".card-owner")).sendKeys("Slaver Test");
driver.findElement(By.cssSelector(".card-cvv")).sendKeys("123");
driver.findElement(By.cssSelector(".card-due")).sendKeys("2020-01-01");
driver.findElement(By.id("new-card-form-submit")).click();
assertThat(driver.findElement(By.cssSelector(".card-tip")).getText(), is("Carta scaduta"));
}
@Test
public void aggiungiCartaFormatoCVVNonValido() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1898, 1018));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("slave");
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-password")).sendKeys(Keys.ENTER);
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("settings-link")));
    }
    driver.findElement(By.id("settings-link")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("add-card")));
    }
    driver.findElement(By.id("add-card")).click();
    driver.findElement(By.cssSelector(".credit-card-add")).click();
    driver.findElement(By.cssSelector(".card-type")).click();
    driver.findElement(By.cssSelector(".card-type")).sendKeys("VISA");
    driver.findElement(By.cssSelector(".card-number")).sendKeys("1111222233334444");
    driver.findElement(By.cssSelector(".card-owner")).sendKeys("Slaver Test");
    driver.findElement(By.cssSelector(".card-cvv")).sendKeys("abc");
    driver.findElement(By.cssSelector(".card-due")).sendKeys("2024-01-01");
    driver.findElement(By.id("new-card-form-submit")).click();
}

```

```

        assertThat(driver.findElement(By.cssSelector(".card-tip")).getText(), is("Formato CVV non valido"));
    }

    @Test
    public void aggiungiCartaFormatoDataNonValido() {
        driver.get("http://localhost:8080/shodan_maven/");
        driver.manage().window().setSize(new Dimension(1898, 1018));
        driver.findElement(By.id("login-username")).click();
        driver.findElement(By.id("login-username")).sendKeys("slave");
        driver.findElement(By.id("login-password")).sendKeys("123");
        driver.findElement(By.id("login-password")).sendKeys(Keys.ENTER);
        {
            WebDriverWait wait = new WebDriverWait(driver, 30);
            wait.until(ExpectedConditions.presenceOfElementLocated(By.id("settings-link")));
        }
        driver.findElement(By.id("settings-link")).click();
        {
            WebDriverWait wait = new WebDriverWait(driver, 30);
            wait.until(ExpectedConditions.presenceOfElementLocated(By.id("add-card")));
        }
        driver.findElement(By.id("add-card")).click();
        driver.findElement(By.cssSelector(".credit-card-add")).click();
        driver.findElement(By.cssSelector(".card-type")).click();
        driver.findElement(By.cssSelector(".card-type")).sendKeys("VISA");
        driver.findElement(By.cssSelector(".card-number")).sendKeys("1111222233334444");
        driver.findElement(By.cssSelector(".card-owner")).sendKeys("Slaver Test");
        driver.findElement(By.cssSelector(".card-cvv")).sendKeys("123");
        driver.findElement(By.cssSelector(".card-due")).sendKeys("2024/01/01");
        driver.findElement(By.id("new-card-form-submit")).click();
        assertThat(driver.findElement(By.cssSelector(".card-tip")).getText(), is("Formato data non valido"));
    }

    @Test
    public void aggiungiCartaFormatoNumeroNonValido() {
        driver.get("http://localhost:8080/shodan_maven/");
        driver.manage().window().setSize(new Dimension(1898, 1018));
        driver.findElement(By.id("login-username")).click();
        driver.findElement(By.id("login-username")).sendKeys("slave");
        driver.findElement(By.id("login-password")).sendKeys("123");
        driver.findElement(By.id("login-password")).sendKeys(Keys.ENTER);
        {
            WebDriverWait wait = new WebDriverWait(driver, 30);
            wait.until(ExpectedConditions.presenceOfElementLocated(By.id("settings-link")));
        }
    }

```

```

driver.findElement(By.id("settings-link")).click();
{
    WebDriverWait wait = new WebDriverWait(driver, 30);
    wait.until(ExpectedConditions.presenceOfElementLocated(By.id("add-card")));
}
driver.findElement(By.id("add-card")).click();
driver.findElement(By.cssSelector(".credit-card-add")).click();
driver.findElement(By.cssSelector(".card-type")).click();
driver.findElement(By.cssSelector(".card-type")).sendKeys("VISA");
driver.findElement(By.cssSelector(".card-number")).sendKeys("1234");
driver.findElement(By.cssSelector(".card-owner")).sendKeys("Slaver Test");
driver.findElement(By.cssSelector(".card-cvv")).sendKeys("123");
driver.findElement(By.cssSelector(".card-due")).sendKeys("2024-01-01");
driver.findElement(By.id("new-card-form-submit")).click();
assertThat(driver.findElement(By.cssSelector(".card-tip")).getText(), is("Formato numero carta non
valido"));
}
@Test
public void aggiungiCartaFormatoTitolareNonValido() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1898, 1018));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("slave");
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-password")).sendKeys(Keys.ENTER);
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("settings-link")));
    }
    driver.findElement(By.id("settings-link")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("add-card")));
    }
    driver.findElement(By.id("add-card")).click();
    driver.findElement(By.cssSelector(".credit-card-add")).click();
    driver.findElement(By.cssSelector(".card-type")).click();
    driver.findElement(By.cssSelector(".card-type")).sendKeys("VISA");
    driver.findElement(By.cssSelector(".card-number")).sendKeys("1111222233334444");
    driver.findElement(By.cssSelector(".card-owner")).sendKeys("test");
    driver.findElement(By.cssSelector(".card-cvv")).sendKeys("123");
    driver.findElement(By.cssSelector(".card-due")).sendKeys("2024-01-01");

```



```

    driver.findElement(By.id("new-card-form-submit")).click();
    assertThat(driver.findElement(By.cssSelector(".card-tip")).getText(), is("Formato titolare non valido"));
}

@Test
public void aggiungiCartaNumeroMancante() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1898, 1018));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("slave");
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-password")).sendKeys(Keys.ENTER);
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("settings-link")));
    }
    driver.findElement(By.id("settings-link")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("add-card")));
    }
    driver.findElement(By.id("add-card")).click();
    driver.findElement(By.cssSelector(".credit-card-add")).click();
    driver.findElement(By.cssSelector(".card-type")).click();
    driver.findElement(By.cssSelector(".card-type")).sendKeys("VISA");
    driver.findElement(By.cssSelector(".card-owner")).sendKeys("Slaver Test");
    driver.findElement(By.cssSelector(".card-cvv")).sendKeys("123");
    driver.findElement(By.cssSelector(".card-due")).sendKeys("2024-01-01");
    driver.findElement(By.id("new-card-form-submit")).click();
    assertThat(driver.findElement(By.cssSelector(".card-tip")).getText(), is("Numero carta: campo obbligatorio"));
}

@Test
public void aggiungiCartaOK() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1898, 1018));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("admin");
    driver.findElement(By.id("login-password")).click();
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-submit")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);

```



```

        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
    }
    driver.findElement(By.id("roles-link")).click();
    driver.findElement(By.cssSelector(".role-item:nth-child(2)")).click();
    driver.findElement(By.id("settings-link")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("add-card")));
    }
    driver.findElement(By.id("add-card")).click();
    driver.findElement(By.cssSelector(".credit-card-add")).click();
    driver.findElement(By.cssSelector(".card-type")).click();
    driver.findElement(By.cssSelector(".card-type")).click();
    driver.findElement(By.cssSelector(".card-type")).sendKeys("NUOVA");
    driver.findElement(By.cssSelector(".card-number")).sendKeys("1111222233334444");
    driver.findElement(By.cssSelector(".card-owner")).sendKeys("Admin Test");
    driver.findElement(By.cssSelector(".card-cvv")).sendKeys("111");
    driver.findElement(By.cssSelector(".card-due")).sendKeys("2022-05-05");
    driver.findElement(By.cssSelector("#new-card-form-submit")).click();
    {
        List<WebElement> elements = driver.findElements(By.cssSelector(".credit-card-add"));
        assert(elements.size() > 0);
    }
}

@Test
public void aggiungiCartaTitolareMancante() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1898, 1018));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("slave");
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-password")).sendKeys(Keys.ENTER);
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("settings-link")));
    }
    driver.findElement(By.id("settings-link")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("add-card")));
    }
    driver.findElement(By.id("add-card")).click();
}

```

```

        driver.findElement(By.cssSelector(".credit-card-add")).click();
        driver.findElement(By.cssSelector(".card-type")).click();
        driver.findElement(By.cssSelector(".card-type")).sendKeys("VISA");
        driver.findElement(By.cssSelector(".card-number")).sendKeys("1111222233334567");
        driver.findElement(By.cssSelector(".card-cvv")).sendKeys("123");
        driver.findElement(By.cssSelector(".card-due")).sendKeys("2024-01-01");
        driver.findElement(By.id("new-card-form-submit")).click();
        assertThat(driver.findElement(By.cssSelector(".card-tip")).getText(), is("Titolare: campo obbligatorio"));
    }
}

```

```

public class AggiungiTitoloTest {
    private WebDriver driver;
    private Map<String, Object> vars;
    private final String UT_NAMESPACE = "AggiungiTitolo_";
    JavascriptExecutor js;

    @Before
    public void setUp() {
        System.setProperty("webdriver.chrome.driver", "../commons/driver/chromedriver.exe");
        driver = new ChromeDriver();
        js = (JavascriptExecutor) driver;
        vars = new HashMap<String, Object>();
    }

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

    @Test
    public void aggiungiTitoloNomeMancante() {
        String testName = UT_NAMESPACE + "NomeMancante";

        File cover = new File("example/Cover.png");
        File sfondo = new File("example/Sfondo.png");

        driver.get("http://localhost:8080/shodan_maven/");
        driver.manage().window().setSize(new Dimension(1294, 1400));
        driver.findElement(By.id("login-username")).click();
        driver.findElement(By.id("login-username")).sendKeys("admin");
        driver.findElement(By.id("login-password")).click();
    }
}

```

```

driver.findElement(By.id("login-password")).sendKeys("123");
driver.findElement(By.id("login-submit")).click();
{
    WebDriverWait wait = new WebDriverWait(driver, 30);
    wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
}
driver.findElement(By.name("game-price")).sendKeys("55");
driver.findElement(By.name("game-date")).sendKeys("1995-01-01");
driver.findElement(By.id("textarea-game")).sendKeys("Descrizione");
driver.findElement(By.id("image-loader")).sendKeys(cover.getAbsolutePath());
driver.findElement(By.id("landscape-loader")).sendKeys(sfondo.getAbsolutePath());
driver.findElement(By.cssSelector(".button:nth-child(9)")).click();
try {
    assertThat(driver.findElement(By.id("add-game-result")).getText(), is("Nome: campo obbligatorio"));
    System.out.println(testName + " PASS");
} catch (AssertionError e) {
    System.out.println(testName + " FAIL");
}
}

```

@Test

```

public void aggiungiTitoloPrezzoMancante() {
    String testName = UT_NAMESPACE + "PrezzoMancante";

    File cover = new File("example/Cover.png");
    File sfondo = new File("example/Sfondo.png");

    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1294, 1400));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("admin");
    driver.findElement(By.id("login-password")).click();
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-submit")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
    }
    driver.findElement(By.name("game-name")).sendKeys("Test");
    // driver.findElement(By.name("game-price")).sendKeys("55");
    driver.findElement(By.name("game-date")).sendKeys("1995-01-01");
    driver.findElement(By.id("textarea-game")).sendKeys("Descrizione");
}

```

```

driver.findElement(By.id("image-loader")).sendKeys(cover.getAbsolutePath());
driver.findElement(By.id("landscape-loader")).sendKeys(sfondo.getAbsolutePath());
driver.findElement(By.cssSelector(".button:nth-child(9)")).click();
try {
    assertThat(driver.findElement(By.id("add-game-result")).getText(), is("Prezzo: campo obbligatorio"));
    System.out.println(testName + " PASS");
} catch (AssertionError e) {
    System.out.println(testName + " FAIL");
}
}

@Test
public void aggiungiTitoloDataMancante() {
    String testName = UT_NAMESPACE + "DataMancante";

    File cover = new File("example/Cover.png");
    File sfondo = new File("example/Sfondo.png");

    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1294, 1400));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("admin");
    driver.findElement(By.id("login-password")).click();
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-submit")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
    }
    driver.findElement(By.name("game-name")).sendKeys("Test");
    driver.findElement(By.name("game-price")).sendKeys("55");
    // driver.findElement(By.name("game-date")).sendKeys("1995-01-01");
    driver.findElement(By.id("textarea-game")).sendKeys("Descrizione");
    driver.findElement(By.id("image-loader")).sendKeys(cover.getAbsolutePath());
    driver.findElement(By.id("landscape-loader")).sendKeys(sfondo.getAbsolutePath());
    driver.findElement(By.cssSelector(".button:nth-child(9)")).click();
    try {
        assertThat(driver.findElement(By.id("add-game-result")).getText(), is("Data: campo obbligatorio"));
        System.out.println(testName + " PASS");
    } catch (AssertionError e) {
        System.out.println(testName + " FAIL");
    }
}

```

```
}
```

```
@Test
```

```
public void aggiungiTitoloDescrizioneMancante() {
```

```
    String testName = UT_NAMESPACE + "DescrizioneMancante";
```

```
    File cover = new File("example/Cover.png");
```

```
    File sfondo = new File("example/Sfondo.png");
```

```
    driver.get("http://localhost:8080/shodan_maven/");
```

```
    driver.manage().window().setSize(new Dimension(1294, 1400));
```

```
    driver.findElement(By.id("login-username")).click();
```

```
    driver.findElement(By.id("login-username")).sendKeys("admin");
```

```
    driver.findElement(By.id("login-password")).click();
```

```
    driver.findElement(By.id("login-password")).sendKeys("123");
```

```
    driver.findElement(By.id("login-submit")).click();
```

```
{
```

```
    WebDriverWait wait = new WebDriverWait(driver, 30);
```

```
    wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
```

```
}
```

```
    driver.findElement(By.name("game-name")).sendKeys("Test");
```

```
    driver.findElement(By.name("game-price")).sendKeys("55");
```

```
    driver.findElement(By.name("game-date")).sendKeys("1995-01-01");
```

```
    // driver.findElement(By.id("textarea-game")).sendKeys("Descrizione");
```

```
    driver.findElement(By.id("image-loader")).sendKeys(cover.getAbsolutePath());
```

```
    driver.findElement(By.id("landscape-loader")).sendKeys(sfondo.getAbsolutePath());
```

```
    driver.findElement(By.cssSelector(".button:nth-child(9)")).click();
```

```
    try {
```

```
        assertThat(driver.findElement(By.id("add-game-result")).getText(), is("Descrizione: campo obbligatorio"));
```

```
        System.out.println(testName + " PASS");
```

```
    } catch (AssertionError e) {
```

```
        System.out.println(testName + " FAIL");
```

```
    }
```

```
}
```

```
@Test
```

```
public void aggiungiTitoloCoverMancante() {
```

```
    String testName = UT_NAMESPACE + "CoverMancante";
```

```
    File cover = new File("example/Cover.png");
```

```
    File sfondo = new File("example/Sfondo.png");
```

```

driver.get("http://localhost:8080/shodan_maven/");
driver.manage().window().setSize(new Dimension(1294, 1400));
driver.findElement(By.id("login-username")).click();
driver.findElement(By.id("login-username")).sendKeys("admin");
driver.findElement(By.id("login-password")).click();
driver.findElement(By.id("login-password")).sendKeys("123");
driver.findElement(By.id("login-submit")).click();
{
    WebDriverWait wait = new WebDriverWait(driver, 30);
    wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
}
driver.findElement(By.name("game-name")).sendKeys("Test");
driver.findElement(By.name("game-price")).sendKeys("55");
driver.findElement(By.name("game-date")).sendKeys("1995-01-01");
driver.findElement(By.id("textarea-game")).sendKeys("Descrizione");
// driver.findElement(By.id("image-loader")).sendKeys(cover.getAbsolutePath());
driver.findElement(By.id("landscape-loader")).sendKeys(sfondo.getAbsolutePath());
driver.findElement(By.cssSelector(".button:nth-child(9)")).click();
try {
    assertThat(driver.findElement(By.id("add-game-result")).getText(), is("Cover: campo obbligatorio"));
    System.out.println(testName + " PASS");
} catch (AssertionError e) {
    System.out.println(testName + " FAIL");
}
}

```

@Test

```

public void aggiungiTitoloSfondoMncante() {
    String testName = UT_NAMESPACE + "SfondoMancante";

    File cover = new File("example/Cover.png");
    File sfondo = new File("example/Sfondo.png");

    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1294, 1400));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("admin");
    driver.findElement(By.id("login-password")).click();
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-submit")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);

```

```

        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
    }

    driver.findElement(By.name("game-name")).sendKeys("Test");
    driver.findElement(By.name("game-price")).sendKeys("55");
    driver.findElement(By.name("game-date")).sendKeys("1995-01-01");
    driver.findElement(By.id("textarea-game")).sendKeys("Descrizione");
    driver.findElement(By.id("image-loader")).sendKeys(cover.getAbsolutePath());
    // driver.findElement(By.id("landscape-loader")).sendKeys(sfondo.getAbsolutePath());
    driver.findElement(By.cssSelector(".button:nth-child(9)")).click();
    try {
        assertThat(driver.findElement(By.id("add-game-result")).getText(), is("Sfondo: campo obbligatorio"));
        System.out.println(testName + " PASS");
    } catch (AssertionError e) {
        System.out.println(testName + " FAIL");
    }
}

@Test
public void aggiungiTitoloFormatoPrezzoNonValido() {
    String testName = UT_NAMESPACE + "FormatoPrezzoNonValido";

    File cover = new File("example/Cover.png");
    File sfondo = new File("example/Sfondo.png");

    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1294, 1400));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("admin");
    driver.findElement(By.id("login-password")).click();
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-submit")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
    }

    driver.findElement(By.name("game-name")).sendKeys("Test");
    driver.findElement(By.name("game-price")).sendKeys("abc");
    driver.findElement(By.name("game-date")).sendKeys("1995-01-01");
    driver.findElement(By.id("textarea-game")).sendKeys("Descrizione");
    driver.findElement(By.id("image-loader")).sendKeys(cover.getAbsolutePath());
    driver.findElement(By.id("landscape-loader")).sendKeys(sfondo.getAbsolutePath());
    driver.findElement(By.cssSelector(".button:nth-child(9)")).click();

```

```

    try {
        assertThat(driver.findElement(By.id("add-game-result")).getText(), is("Formato prezzo non valido"));
        System.out.println(testName + " PASS");
    } catch (AssertionError e) {
        System.out.println(testName + " FAIL");
    }
}

```

@Test

```

public void aggiungiTitoloDataFuoriIntervallo() {
    String testName = UT_NAMESPACE + "DataFuoriIntervalloValido";

    File cover = new File("example/Cover.png");
    File sfondo = new File("example/Sfondo.png");

    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1294, 1400));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("admin");
    driver.findElement(By.id("login-password")).click();
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-submit")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
    }
    driver.findElement(By.name("game-name")).sendKeys("Test");
    driver.findElement(By.name("game-price")).sendKeys("50");
    driver.findElement(By.name("game-date")).sendKeys("2020-01-01");
    driver.findElement(By.id("textarea-game")).sendKeys("Descrizione");
    driver.findElement(By.id("image-loader")).sendKeys(cover.getAbsolutePath());
    driver.findElement(By.id("landscape-loader")).sendKeys(sfondo.getAbsolutePath());
    driver.findElement(By.cssSelector(".button:nth-child(9)")).click();
    try {
        assertThat(driver.findElement(By.id("add-game-result")).getText(), is("Data fuori intervallo"));
        System.out.println(testName + " PASS");
    } catch (AssertionError e) {
        System.out.println(testName + " FAIL");
    }
}

```

@Test



```

public void aggiungiTitoloFormatoDataNonValido() {
    String testName = UT_NAMESPACE + "FormatoDataNonValido";

    File cover = new File("example/Cover.png");
    File sfondo = new File("example/Sfondo.png");

    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1294, 1400));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("admin");
    driver.findElement(By.id("login-password")).click();
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-submit")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
    }
    driver.findElement(By.name("game-name")).sendKeys("Test");
    driver.findElement(By.name("game-price")).sendKeys("50");
    driver.findElement(By.name("game-date")).sendKeys("1995/01/01");
    driver.findElement(By.id("textarea-game")).sendKeys("Descrizione");
    driver.findElement(By.id("image-loader")).sendKeys(cover.getAbsolutePath());
    driver.findElement(By.id("landscape-loader")).sendKeys(sfondo.getAbsolutePath());
    driver.findElement(By.cssSelector(".button:nth-child(9)")).click();
    try {
        assertThat(driver.findElement(By.id("add-game-result")).getText(), is("Formato data non valido"));
        System.out.println(testName + " PASS");
    } catch (AssertionError e) {
        System.out.println(testName + " FAIL");
    }
}

@Test
public void aggiungiTitoloTitoloEsistente() {
    String testName = UT_NAMESPACE + "TitoloEsistente";

    File cover = new File("example/Cover.png");
    File sfondo = new File("example/Sfondo.png");

    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1294, 1400));
    driver.findElement(By.id("login-username")).click();

```

```

driver.findElement(By.id("login-username")).sendKeys("admin");
driver.findElement(By.id("login-password")).click();
driver.findElement(By.id("login-password")).sendKeys("123");
driver.findElement(By.id("login-submit")).click();
{
    WebDriverWait wait = new WebDriverWait(driver, 30);
    wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
}
driver.findElement(By.name("game-name")).sendKeys("System Shock");
driver.findElement(By.name("game-price")).sendKeys("50");
driver.findElement(By.name("game-date")).sendKeys("1995-01-01");
driver.findElement(By.id("textarea-game")).sendKeys("Descrizione");
driver.findElement(By.id("image-loader")).sendKeys(cover.getAbsolutePath());
driver.findElement(By.id("landscape-loader")).sendKeys(sfondo.getAbsolutePath());
driver.findElement(By.cssSelector(".button:nth-child(9)")).click();
try {
    assertThat(driver.findElement(By.id("add-game-result")).getText(), is("Titolo gia' presente"));
    System.out.println(testName + " PASS");
} catch (AssertionError e) {
    System.out.println(testName + " FAIL");
}
}

@Test
public void aggiungiTitoloPrezzoNegativo() {
    String testName = UT_NAMESPACE + "PrezzoNegativo";

    File cover = new File("example/Cover.png");
    File sfondo = new File("example/Sfondo.png");

    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1294, 1400));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("admin");
    driver.findElement(By.id("login-password")).click();
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-submit")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
    }
    driver.findElement(By.name("game-name")).sendKeys("Test");

```

```

driver.findElement(By.name("game-price")).sendKeys("-50");
driver.findElement(By.name("game-date")).sendKeys("1995-01-01");
driver.findElement(By.id("textarea-game")).sendKeys("Descrizione");
driver.findElement(By.id("image-loader")).sendKeys(cover.getAbsolutePath());
driver.findElement(By.id("landscape-loader")).sendKeys(sfondo.getAbsolutePath());
driver.findElement(By.cssSelector(".button:nth-child(9)")).click();
try {
    assertThat(driver.findElement(By.id("add-game-result")).getText(), is("Prezzo negativo"));
    System.out.println(testName + " PASS");
} catch (AssertionError e) {
    System.out.println(testName + " FAIL");
}
}

@Test
public void aggiungiTitoloFormatoNomeNonValido() {
    String testName = UT_NAMESPACE + "FormatoNomeNonValido";

    File cover = new File("example/Cover.png");
    File sfondo = new File("example/Sfondo.png");

    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1294, 1400));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("admin");
    driver.findElement(By.id("login-password")).click();
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-submit")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
    }
    driver.findElement(By.name("game-name")).sendKeys("Test__:^");
    driver.findElement(By.name("game-price")).sendKeys("50");
    driver.findElement(By.name("game-date")).sendKeys("1995-01-01");
    driver.findElement(By.id("textarea-game")).sendKeys("Descrizione");
    driver.findElement(By.id("image-loader")).sendKeys(cover.getAbsolutePath());
    driver.findElement(By.id("landscape-loader")).sendKeys(sfondo.getAbsolutePath());
    driver.findElement(By.cssSelector(".button:nth-child(9)")).click();
    try {
        assertThat(driver.findElement(By.id("add-game-result")).getText(), is("Formato nome non valido"));
        System.out.println(testName + " PASS");
    }
}

```

```

    } catch (AssertionError e) {
        System.out.println(testName + " FAIL");
    }
}

@Test
public void aggiungiTitoloFormatoCoverNonValido() {
    String testName = UT_NAMESPACE + "FormatoCoverNonValido";

    File cover = new File("example/dummy.txt");
    File sfondo = new File("example/Sfondo.png");

    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1294, 1400));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("admin");
    driver.findElement(By.id("login-password")).click();
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-submit")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
    }
    driver.findElement(By.name("game-name")).sendKeys("Test");
    driver.findElement(By.name("game-price")).sendKeys("50");
    driver.findElement(By.name("game-date")).sendKeys("1995-01-01");
    driver.findElement(By.id("textarea-game")).sendKeys("Descrizione");
    driver.findElement(By.id("image-loader")).sendKeys(cover.getAbsolutePath());
    driver.findElement(By.id("landscape-loader")).sendKeys(sfondo.getAbsolutePath());
    driver.findElement(By.cssSelector(".button:nth-child(9)")).click();
    try {
        assertThat(driver.findElement(By.id("add-game-result")).getText(), is("Formato cover non valido"));
        System.out.println(testName + " PASS");
    } catch (AssertionError e) {
        System.out.println(testName + " FAIL");
    }
}

@Test
public void aggiungiTitoloFormatoSfondoNonValido() {
    String testName = UT_NAMESPACE + "FormatoSfondoNonValido";

```

```

File cover = new File("example/Cover.png");
File sfondo = new File("example/dummy.txt");

driver.get("http://localhost:8080/shodan_maven/");
driver.manage().window().setSize(new Dimension(1294, 1400));
driver.findElement(By.id("login-username")).click();
driver.findElement(By.id("login-username")).sendKeys("admin");
driver.findElement(By.id("login-password")).click();
driver.findElement(By.id("login-password")).sendKeys("123");
driver.findElement(By.id("login-submit")).click();
{
    WebDriverWait wait = new WebDriverWait(driver, 30);
    wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
}
driver.findElement(By.name("game-name")).sendKeys("Test");
driver.findElement(By.name("game-price")).sendKeys("50");
driver.findElement(By.name("game-date")).sendKeys("1995-01-01");
driver.findElement(By.id("textarea-game")).sendKeys("Descrizione");
driver.findElement(By.id("image-loader")).sendKeys(cover.getAbsolutePath());
driver.findElement(By.id("landscape-loader")).sendKeys(sfondo.getAbsolutePath());
driver.findElement(By.cssSelector(".button:nth-child(9)")).click();
try {
    assertThat(driver.findElement(By.id("add-game-result")).getText(), is("Formato sfondo non valido"));
    System.out.println(testName + " PASS");
} catch (AssertionError e) {
    System.out.println(testName + " FAIL");
}
}

@Test
public void aggiungiTitoloOK() {
    String testName = UT_NAMESPACE + "OK";

    File cover = new File("example/Cover.png");
    File sfondo = new File("example/Sfondo.png");

    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1294, 1400));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("admin");
    driver.findElement(By.id("login-password")).click();
    driver.findElement(By.id("login-password")).sendKeys("123");

```

```

driver.findElement(By.id("login-submit")).click();
{
    WebDriverWait wait = new WebDriverWait(driver, 30);
    wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
}
driver.findElement(By.name("game-name")).sendKeys("Test");
driver.findElement(By.name("game-price")).sendKeys("50");
driver.findElement(By.name("game-date")).sendKeys("1995-01-01");
driver.findElement(By.id("textarea-game")).sendKeys("Descrizione");
driver.findElement(By.id("image-loader")).sendKeys(cover.getAbsolutePath());
driver.findElement(By.id("landscape-loader")).sendKeys(sfondo.getAbsolutePath());
driver.findElement(By.cssSelector(".button:nth-child(9)")).click();
try {
    assertThat(driver.findElement(By.id("add-game-result")).getText(), is("Gioco aggiunto"));
    System.out.println(testName + " PASS");
} catch (AssertionError e) {
    System.out.println(testName + " FAIL");
}
}
}
}

```

```

public class LoginTest {
    private WebDriver driver;
    private Map<String, Object> vars;
    JavascriptExecutor js;
    @Before
    public void setUp() {
        System.setProperty("webdriver.chrome.driver", "../commons/driver/chromedriver.exe");
        driver = new ChromeDriver();
        js = (JavascriptExecutor) driver;
        vars = new HashMap<String, Object>();
    }
    @After
    public void tearDown() {
        driver.quit();
    }
    @Test
    public void loginOK() {
        driver.get("http://localhost:8080/shodan_maven/");
        driver.manage().window().setSize(new Dimension(1900, 1020));
        driver.findElement(By.id("login-username")).click();
        driver.findElement(By.id("login-username")).sendKeys("admin");
    }
}

```

```

driver.findElement(By.id("login-password")).click();
driver.findElement(By.id("login-password")).sendKeys("123");
driver.findElement(By.id("login-submit")).click();
{
    WebDriverWait wait = new WebDriverWait(driver, 30);
    wait.until(ExpectedConditions.presenceOfElementLocated(By.id("app")));
}
}

@Test
public void loginPasswordErrata() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1900, 1020));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("admin");
    driver.findElement(By.id("login-password")).click();
    driver.findElement(By.id("login-password")).sendKeys("1");
    driver.findElement(By.id("login-submit")).click();
    assertThat(driver.findElement(By.id("login-message")).getText(), is("Password errata"));
}

@Test
public void loginUtenteInesistente() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1900, 1020));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("utenteNonEsistente");
    driver.findElement(By.id("login-password")).sendKeys("12345a");
    driver.findElement(By.id("login-submit")).click();
    assertThat(driver.findElement(By.id("login-message")).getText(), is("Utente non esistente"));
}
}

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

    @Before
    public void setUp() {
        System.setProperty("webdriver.chrome.driver", "../commons/driver/chromedriver.exe");
        driver = new ChromeDriver();
        js = (JavascriptExecutor) driver;
        vars = new HashMap<String, Object>();
    }
}

```

```

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

@Test
public void modificaEmailOK() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1900, 1020));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("slave");
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-password")).sendKeys(Keys.ENTER);
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("settings-link")));
    }
    driver.findElement(By.id("settings-link")).click();
    driver.findElement(By.id("settings-input-last-email")).click();
    driver.findElement(By.id("settings-input-last-email")).sendKeys("slave@shodan.it");
    driver.findElement(By.id("settings-input-email")).sendKeys("slave2@shodan.it");
    driver.findElement(By.id("settings-submit-email")).click();
    assertThat(driver.findElement(By.id("email-change-message")).getText(), is("Email modificata con
successo!"));
}

@Test
public void modificaEmailNuovaEmailMancante() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1900, 1020));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("dummy");
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-submit")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("settings-link")));
    }
    driver.findElement(By.id("settings-link")).click();
    driver.findElement(By.id("settings-input-last-email")).click();
    driver.findElement(By.id("settings-input-last-email")).sendKeys("slave@shodan.it");
    driver.findElement(By.id("settings-input-last-email")).click();
    driver.findElement(By.id("settings-input-last-email")).sendKeys("dummy@shodan.it");
    driver.findElement(By.id("settings-submit-email")).click();
}

```



```

{
    List<WebElement> elements = driver.findElements(By.xpath("//div[contains(@email-change-
message,\'\'')]]"));
    assert(elements.size() > 0);
}
}

@Test
public void modificaEmailFormatoNonValido() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1900, 1020));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("dummy");
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-password")).sendKeys(Keys.ENTER);
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("settings-link")));
    }
    driver.findElement(By.id("settings-link")).click();
    driver.findElement(By.id("settings-input-last-email")).click();
    driver.findElement(By.id("settings-input-last-email")).sendKeys("dummy@shodan.it");
    driver.findElement(By.id("settings-input-email")).sendKeys("dd@@dd.it");
    driver.findElement(By.id("settings-submit-email")).click();
    {
        List<WebElement> elements = driver.findElements(By.xpath("//div[contains(@password-change-
message,\'\'')]]"));
        assert(elements.size() > 0);
    }
}

@Test
public void modificaEmailEmailInUso() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1900, 1020));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("dummy");
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-password")).sendKeys(Keys.ENTER);
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("settings-link")));
    }
    driver.findElement(By.id("settings-link")).click();

```

```

        driver.findElement(By.id("settings-input-last-email")).click();
        driver.findElement(By.id("settings-input-last-email")).sendKeys("dummy@shodan.it");
        driver.findElement(By.id("settings-input-email")).sendKeys("antonio@shodan.it");
        driver.findElement(By.id("settings-submit-email")).click();
        assertThat(driver.findElement(By.id("email-change-message")).getText(), is("Impossibile usare questa
email."));
    }

    @Test
    public void modificaEmailEmailAttualeErrata() {
        driver.get("http://localhost:8080/shodan_maven/");
        driver.manage().window().setSize(new Dimension(1900, 1020));
        driver.findElement(By.id("login-username")).click();
        driver.findElement(By.id("login-username")).sendKeys("dummy");
        driver.findElement(By.id("login-password")).click();
        driver.findElement(By.id("login-password")).sendKeys("123");
        driver.findElement(By.id("login-submit")).click();
        {
            WebDriverWait wait = new WebDriverWait(driver, 30);
            wait.until(ExpectedConditions.presenceOfElementLocated(By.id("settings-link")));
        }
        driver.findElement(By.id("settings-link")).click();
        driver.findElement(By.id("settings-input-last-email")).click();
        driver.findElement(By.id("settings-input-last-email")).sendKeys("duummy@shodan.it");
        driver.findElement(By.id("settings-input-email")).sendKeys("nuova@shodan.it");
        driver.findElement(By.id("settings-input-email")).sendKeys(Keys.ENTER);
        assertThat(driver.findElement(By.id("email-change-message")).getText(), is("Non hai inserito correttamente
la tua email attuale."));
    }
}

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

    @Before
    public void setUp() {
        System.setProperty("webdriver.chrome.driver", "../commons/driver/chromedriver.exe");
        driver = new ChromeDriver();
        js = (JavascriptExecutor) driver;
        vars = new HashMap<String, Object>();
    }

    @After

```

```

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

@Test
public void modificaPasswordPasswordAttualeErrata() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1900, 1020));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("dummy");
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-password")).sendKeys(Keys.ENTER);
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("settings-link")));
    }
    driver.findElement(By.id("settings-link")).click();
    {
        WebElement element = driver.findElement(By.id("settings-submit-password"));
        Actions builder = new Actions(driver);
        builder.moveToElement(element).perform();
    }
    {
        WebElement element = driver.findElement(By.tagName("body"));
        Actions builder = new Actions(driver);
        builder.moveToElement(element, 0, 0).perform();
    }
    driver.findElement(By.id("settings-input-old-password")).click();
    driver.findElement(By.id("settings-input-old-password")).sendKeys("1");
    driver.findElement(By.id("settings-input-new-password")).sendKeys("a1234");
    driver.findElement(By.id("settings-input-new-password-again")).sendKeys("a1234");
    driver.findElement(By.id("settings-submit-password")).click();
    assertThat(driver.findElement(By.id("password-change-message")).getText(), is("Password corrente errata."));
}

@Test
public void modificaPasswordOK() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1900, 1020));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("puppet");
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-password")).sendKeys(Keys.ENTER);

```

```

{
    WebDriverWait wait = new WebDriverWait(driver, 30);
    wait.until(ExpectedConditions.presenceOfElementLocated(By.id("settings-link")));
}

driver.findElement(By.id("settings-link")).click();
driver.findElement(By.id("settings-input-old-password")).click();
driver.findElement(By.id("settings-input-old-password")).sendKeys("123");
driver.findElement(By.id("settings-input-new-password")).sendKeys("a1234");
driver.findElement(By.id("settings-input-new-password-again")).sendKeys("a1234");
driver.findElement(By.id("settings-submit-password")).click();
assertThat(driver.findElement(By.id("password-change-message")).getText(), is("Password modificata con
successo!"));
}

@Test
public void modificaPasswordNuovaPasswordMancante() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1900, 1020));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("dummy");
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-submit")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("settings-link")));
    }
    driver.findElement(By.id("settings-link")).click();
    driver.findElement(By.id("settings-input-old-password")).click();
    driver.findElement(By.id("settings-input-old-password")).sendKeys("123");
    driver.findElement(By.id("settings-input-new-password")).click();
    driver.findElement(By.id("settings-input-new-password")).sendKeys("a1234");
    driver.findElement(By.id("settings-submit-password")).click();
    {
        List<WebElement> elements = driver.findElements(By.xpath("//div[contains(@password-change-
message, '\\'')]]"));
        assert(elements.size() > 0);
    }
}

@Test
public void modificaPasswordNonCoincidenti() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1900, 1020));
    driver.findElement(By.id("login-username")).click();

```

```

driver.findElement(By.id("login-username")).sendKeys("dummy");
driver.findElement(By.id("login-password")).sendKeys("123");
driver.findElement(By.id("login-password")).sendKeys(Keys.ENTER);
{
    WebDriverWait wait = new WebDriverWait(driver, 30);
    wait.until(ExpectedConditions.presenceOfElementLocated(By.id("settings-link")));
}
driver.findElement(By.id("settings-link")).click();
driver.findElement(By.id("settings-input-old-password")).click();
driver.findElement(By.id("settings-input-old-password")).sendKeys("123");
driver.findElement(By.id("settings-input-new-password")).sendKeys("a1234");
driver.findElement(By.id("settings-input-new-password-again")).sendKeys("a1235");
driver.findElement(By.id("settings-submit-password")).click();
assertThat(driver.findElement(By.id("password-change-message")).getText(), is("Le password non
coincidono."));
}
@Test
public void modificaPasswordFormatoNuovaPasswordInvalido() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1900, 1020));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("dummy");
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-submit")).click();
    driver.findElement(By.id("settings-link")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("settings-link")));
    }
    driver.findElement(By.id("settings-link")).click();
    driver.findElement(By.id("settings-input-old-password")).sendKeys("123");
    driver.findElement(By.id("settings-input-new-password")).sendKeys("a123");
    driver.findElement(By.id("settings-input-new-password-again")).sendKeys("a123");
    driver.findElement(By.id("settings-submit-password")).click();
    driver.findElement(By.id("settings-submit-password")).click();
    assertThat(driver.findElement(By.id("password-change-message")).getText(), is("La password deve avere
almeno cinque caratteri, di cui almeno una lettera e un numero."));
}
}

public class PagamentoTest {
    private WebDriver driver;

```

```

private Map<String, Object> vars;
JavascriptExecutor js;
@Before
public void setUp() {
    System.setProperty("webdriver.chrome.driver", "../commons/driver/chromedriver.exe");
    driver = new ChromeDriver();
    js = (JavascriptExecutor) driver;
    vars = new HashMap<String, Object>();
}
@After
public void tearDown() {
    driver.quit();
}
@Test
public void pagamentoSaldoInsufficiente() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1898, 1018));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("slave");
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-password")).sendKeys(Keys.ENTER);
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.cssSelector(".game-container:nth-child(1) > .game-add")));
    }
    driver.findElement(By.cssSelector(".game-container:nth-child(1) > .game-add")).click();
    driver.findElement(By.id("cart-link")).click();
    driver.findElement(By.id("cart-pay")).click();
    assertThat(driver.findElement(By.cssSelector(".last-row-info")).getText(), is("Non hai abbastanza soldi."));
}
@Test
public void pagamentoOK() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1898, 1018));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("dummy");
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-password")).sendKeys(Keys.ENTER);
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
    }
}

```

```

        wait.until(ExpectedConditions.presenceOfElementLocated(By.cssSelector(".game-container:nth-child(1) >
.game-add"))));
    }
    driver.findElement(By.cssSelector(".game-container:nth-child(1) > .game-add")).click();
    driver.findElement(By.id("cart-link")).click();
    driver.findElement(By.id("cart-pay")).click();
}

@Test
public void pagamentoCarrelloVuoto() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1898, 1018));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("dummy");
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-password")).sendKeys(Keys.ENTER);
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("cart-link")));
    }
    driver.findElement(By.id("cart-link")).click();
    assertThat(driver.findElement(By.cssSelector(".empty-view > div")).getText(), is("Non c'è ancora nulla
qui..."));
}
}

public class RegistrazioneTest {
    private WebDriver driver;
    private Map<String, Object> vars;
    JavascriptExecutor js;
    @Before
    public void setUp() {
        System.setProperty("webdriver.chrome.driver", "../commons/driver/chromedriver.exe");
        driver = new ChromeDriver();
        js = (JavascriptExecutor) driver;
        vars = new HashMap<String, Object>();
    }
    @After
    public void tearDown() {
        driver.quit();
    }
    @Test
    public void registrazioneOK() {

```

```

driver.get("http://localhost:8080/shodan_maven/");
driver.manage().window().setSize(new Dimension(1822, 1054));
driver.findElement(By.id("flip-login-card")).click();
driver.findElement(By.id("signin-username")).click();
driver.findElement(By.id("signin-username")).sendKeys("Francesco");
driver.findElement(By.id("signin-password")).sendKeys("a1234");
driver.findElement(By.id("signin-password-again")).click();
driver.findElement(By.id("signin-password-again")).sendKeys("a1234");
driver.findElement(By.id("signin-email")).sendKeys("Francesco@Shodan.it");
driver.findElement(By.id("signin-submit")).click();
{
    List<WebElement> elements = driver.findElements(By.id("signin-message"));
    assert(elements.size() > 0);
}
}

@Test
public void registrazioneEmailNonValida() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1822, 1040));
    driver.findElement(By.id("flip-login-card")).click();
    driver.findElement(By.id("signin-username")).click();
    driver.findElement(By.id("signin-username")).sendKeys("test");
    driver.findElement(By.id("signin-password")).click();
    driver.findElement(By.id("signin-password")).sendKeys("a1234");
    driver.findElement(By.id("signin-password-again")).click();
    driver.findElement(By.id("signin-password-again")).sendKeys("a1234");
    driver.findElement(By.id("signin-email")).click();
    driver.findElement(By.id("signin-email")).sendKeys("test@@a.it");
    driver.findElement(By.id("signin-submit")).click();
    {
        List<WebElement> elements = driver.findElements(By.xpath("//div[contains(@signin-message, '\')]"));
        assert(elements.size() > 0);
    }
}

@Test
public void registrazioneEmailInUso() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1822, 1040));
    driver.findElement(By.id("flip-login-card")).click();
    driver.findElement(By.id("signin-username")).click();
    driver.findElement(By.id("signin-username")).sendKeys("test");
    driver.findElement(By.id("signin-password")).click();

```



```

driver.findElement(By.id("signin-password")).sendKeys("a1234");
driver.findElement(By.id("signin-password-again")).click();
driver.findElement(By.id("signin-password-again")).sendKeys("a1234");
driver.findElement(By.id("signin-email")).click();
driver.findElement(By.id("signin-email")).sendKeys("antonio@shodan.it");
driver.findElement(By.id("signin-submit")).click();
{
    List<WebElement> elements = driver.findElements(By.id("signin-message"));
    assert(elements.size() > 0);
}
}

@Test
public void registrazioneEmailMancante() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1900, 1020));
    driver.findElement(By.id("flip-login-card")).click();
    driver.findElement(By.id("signin-username")).click();
    driver.findElement(By.id("signin-username")).sendKeys("test");
    driver.findElement(By.id("signin-username")).sendKeys("test5");
    driver.findElement(By.id("signin-password")).click();
    driver.findElement(By.id("signin-password")).sendKeys("a1234");
    driver.findElement(By.id("signin-password-again")).click();
    driver.findElement(By.id("signin-password-again")).sendKeys("a1234");
    driver.findElement(By.id("signin-submit")).click();
    {
        List<WebElement> elements = driver.findElements(By.xpath("//div[contains(@signin-message, '\\'')]]"));
        assert(elements.size() > 0);
    }
}

@Test
public void registrazionePasswordDiverse() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1900, 1020));
    driver.findElement(By.id("flip-login-card")).click();
    driver.findElement(By.id("signin-username")).click();
    driver.findElement(By.id("signin-username")).sendKeys("test_nuovo");
    driver.findElement(By.id("signin-password")).sendKeys("a1234");
    driver.findElement(By.id("signin-password-again")).click();
    driver.findElement(By.id("signin-password-again")).sendKeys("a1235");
    driver.findElement(By.id("signin-email")).click();
    driver.findElement(By.id("signin-email")).sendKeys("tes3t@test.it");
    driver.findElement(By.id("signin-submit")).click();
}

```

```

    assertThat(driver.findElement(By.id("signin-message")).getText(), is("Le password non coincidono"));
}

@Test
public void registrazionePasswordMancante() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1900, 1020));
    driver.findElement(By.id("flip-login-card")).click();
    driver.findElement(By.id("signin-username")).click();
    driver.findElement(By.id("signin-username")).sendKeys("test3");
    driver.findElement(By.id("signin-password-again")).click();
    driver.findElement(By.id("signin-password-again")).sendKeys("a1234");
    driver.findElement(By.id("signin-email")).click();
    driver.findElement(By.id("signin-email")).sendKeys("test2@test.it");
    driver.findElement(By.id("signin-submit")).click();
    {
        List<WebElement> elements = driver.findElements(By.xpath("//div[contains(@signin-message,\'\\\'\\')]]"));
        assertThat(elements.size() > 0);
    }
}

@Test
public void registrazionePasswordNonValida() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1900, 1020));
    driver.findElement(By.id("flip-login-card")).click();
    driver.findElement(By.id("signin-username")).click();
    driver.findElement(By.id("signin-username")).sendKeys("test");
    driver.findElement(By.id("signin-username")).sendKeys("test5");
    driver.findElement(By.id("signin-password")).click();
    driver.findElement(By.id("signin-password")).sendKeys("a123");
    driver.findElement(By.id("signin-password-again")).click();
    driver.findElement(By.id("signin-password-again")).sendKeys("a123");
    driver.findElement(By.id("signin-email")).click();
    driver.findElement(By.id("signin-email")).sendKeys("test2@test.it");
    driver.findElement(By.id("signin-submit")).click();
    {
        WebElement element = driver.findElement(By.id("signin-submit"));
        Actions builder = new Actions(driver);
        builder.moveToElement(element).perform();
    }
    {
        WebElement element = driver.findElement(By.tagName("body"));
        Actions builder = new Actions(driver);
    }
}

```

```

        builder.moveToElement(element, 0, 0).perform();
    }

    assertThat(driver.findElement(By.id("signin-message")).getText(), is("La password non rispetta i
criteri.\\n\\nSono necessari almeno cinque caratteri,\\n\\nndi cui per lo meno un numero e una lettera."));
}

@Test
public void registrazioneRipetiPasswordMancante() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1900, 1020));
    driver.findElement(By.id("flip-login-card")).click();
    driver.findElement(By.id("signin-username")).click();
    driver.findElement(By.id("signin-username")).sendKeys("test5");
    driver.findElement(By.id("signin-password")).click();
    driver.findElement(By.id("signin-password")).sendKeys("test1");
    driver.findElement(By.id("signin-email")).click();
    driver.findElement(By.id("signin-email")).sendKeys("test2@test.it");
    driver.findElement(By.id("signin-submit")).click();
    {
        List<WebElement> elements = driver.findElements(By.xpath("//div[contains(@signin-message, '\\n')]"));
        assertThat(elements.size(), > 0);
    }
}

@Test
public void registrazioneUsernameInUso() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1900, 1020));
    driver.findElement(By.id("flip-login-card")).click();
    driver.findElement(By.id("signin-username")).click();
    driver.findElement(By.id("signin-username")).sendKeys("admin");
    driver.findElement(By.id("signin-password")).sendKeys("a1234");
    driver.findElement(By.id("signin-password-again")).click();
    driver.findElement(By.id("signin-password-again")).sendKeys("a1234");
    driver.findElement(By.id("signin-email")).click();
    driver.findElement(By.id("signin-email")).sendKeys("antonio@shodan.it");
    driver.findElement(By.id("signin-submit")).click();
    assertThat(driver.findElement(By.id("signin-message")).getText(), is("Username esistente"));
}

@Test
public void registrazioneUsernameMancante() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1900, 1020));
    driver.findElement(By.id("flip-login-card")).click();

```

```

        driver.findElement(By.id("signin-password")).click();
        driver.findElement(By.id("signin-password")).sendKeys("a1234");
        driver.findElement(By.id("signin-password-again")).click();
        driver.findElement(By.id("signin-password-again")).sendKeys("a1234");
        driver.findElement(By.id("signin-email")).click();
        driver.findElement(By.id("signin-email")).sendKeys("test@test.it");
        driver.findElement(By.id("signin-submit")).click();
    }
    List<WebElement> elements = driver.findElements(By.xpath("//div[contains(@signin-message,\'\\\'\\')]]"));
    assert(elements.size() > 0);
}
}

@Test
public void registrazioneUsernameNonValido() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1900, 1020));
    driver.findElement(By.id("flip-login-card")).click();
    driver.findElement(By.id("signin-username")).click();
    driver.findElement(By.id("signin-username")).sendKeys("a");
    driver.findElement(By.id("signin-password")).click();
    driver.findElement(By.id("signin-password")).sendKeys("1234a");
    driver.findElement(By.id("signin-password-again")).click();
    driver.findElement(By.id("signin-password-again")).sendKeys("1234a");
    driver.findElement(By.id("signin-email")).click();
    driver.findElement(By.id("signin-email")).sendKeys("antonio@shodan.it");
    driver.findElement(By.id("signin-submit")).click();
    assertThat(driver.findElement(By.id("signin-message")).getText(), is("Formato username non valido"));
}
}

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

    @Before
    public void setUp() {
        System.setProperty("webdriver.chrome.driver", "../commons/driver/chromedriver.exe");
        driver = new ChromeDriver();
        js = (JavascriptExecutor) driver;
        vars = new HashMap<String, Object>();
    }

    @After

```

```

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

@Test
public void ricaricaSaldoQuantitaNonValida() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1898, 1018));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("slave");
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-password")).sendKeys(Keys.ENTER);
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("settings-link")));
    }
    driver.findElement(By.id("settings-link")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("add-card")));
    }
    driver.findElement(By.id("add-card")).click();
    driver.findElement(By.id("amount-input")).click();
    driver.findElement(By.id("amount-input")).sendKeys("0");
    driver.findElement(By.cssSelector(".valid-credit-card")).click();
    assertThat(driver.findElement(By.id("cards-message")).getText(), is("Quantita\\\\" non valido."));
}

@Test
public void ricaricaSaldoOK() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1898, 1018));
    driver.findElement(By.id("login-username")).sendKeys("admin");
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-password")).sendKeys(Keys.ENTER);
    driver.findElement(By.id("roles-link")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
    }
    driver.findElement(By.cssSelector(".role-item:nth-child(2)")).click();
    driver.findElement(By.id("settings-link")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
    }
}

```

```

        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("add-card")));
    }
    driver.findElement(By.id("add-card")).click();
    driver.findElement(By.cssSelector(".VISA h1")).click();
    assertThat(driver.findElement(By.id("cards-message")).getText(), is("Ricarica effettuata con successo!"));
}

@Test
public void ricaricaSaldoFormatoQuantitaNonValido() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1898, 1018));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("slave");
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-password")).sendKeys(Keys.ENTER);
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("settings-link")));
    }
    driver.findElement(By.id("settings-link")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("add-card")));
    }
    driver.findElement(By.id("add-card")).click();
    driver.findElement(By.id("amount-input")).click();
    driver.findElement(By.id("amount-input")).sendKeys("uno");
    driver.findElement(By.cssSelector(".valid-credit-card")).click();
    assertThat(driver.findElement(By.id("cards-message")).getText(), is("Formato quantita\\\\" non valido"));
}

@Test
public void ricaricaSaldoCartaScaduta() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1898, 1018));
    driver.findElement(By.id("login-username")).sendKeys("slave");
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-password")).sendKeys(Keys.ENTER);
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("settings-link")));
    }
    driver.findElement(By.id("settings-link")).click();
    {

```

```

        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("add-card")));
    }
    driver.findElement(By.id("add-card")).click();
    driver.findElement(By.cssSelector(".MASTERCARD h1")).click();
    assertThat(driver.findElement(By.id("cards-message")).getText(), is("Carta scaduta"));
}
}

public class RimuoviArticoloTest {
    private WebDriver driver;
    private Map<String, Object> vars;
    JavascriptExecutor js;
    @Before
    public void setUp() {
        System.setProperty("webdriver.chrome.driver", "../commons/driver/chromedriver.exe");
        driver = new ChromeDriver();
        js = (JavascriptExecutor) driver;
        vars = new HashMap<String, Object>();
    }
    @After
    public void tearDown() {
        driver.quit();
    }
    @Test
    public void rimuoviArticoloOK() {
        driver.get("http://localhost:8080/shodan_maven/");
        driver.manage().window().setSize(new Dimension(1898, 1018));
        driver.findElement(By.id("login-username")).click();
        driver.findElement(By.id("login-username")).sendKeys("admin");
        driver.findElement(By.id("login-password")).click();
        driver.findElement(By.id("login-password")).sendKeys("123");
        driver.findElement(By.id("login-submit")).click();
        {
            WebDriverWait wait = new WebDriverWait(driver, 30);
            wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
        }
        driver.findElement(By.id("roles-link")).click();
        driver.findElement(By.cssSelector(".role-item:nth-child(3)")).click();
        driver.findElement(By.id("remove-article-id")).click();
        driver.findElement(By.id("remove-article-id")).sendKeys("2");
        driver.findElement(By.cssSelector(".button:nth-child(2)")).click();
    }
}

```

```

        assertThat(driver.findElement(By.id("remove-article-result")).getText(), is("Articolo rimosso con
successo!"));
    }

    @Test
    public void rimuoviArticoloIDMancante() {
        driver.get("http://localhost:8080/shodan_maven/");
        driver.manage().window().setSize(new Dimension(1898, 1018));
        driver.findElement(By.id("login-username")).click();
        driver.findElement(By.id("login-username")).sendKeys("admin");
        driver.findElement(By.id("login-password")).click();
        driver.findElement(By.id("login-password")).sendKeys("123");
        driver.findElement(By.id("login-submit")).click();
        {
            WebDriverWait wait = new WebDriverWait(driver, 30);
            wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
        }
        driver.findElement(By.id("roles-link")).click();
        driver.findElement(By.cssSelector(".role-item:nth-child(3)")).click();
        driver.findElement(By.cssSelector(".button:nth-child(2)")).click();
        assertThat(driver.findElement(By.id("remove-article-result")).getText(), is("ID: campo obbligatorio."));
    }

    @Test
    public void rimuoviArticoloFormatoIDErrato() {
        driver.get("http://localhost:8080/shodan_maven/");
        driver.manage().window().setSize(new Dimension(1898, 1018));
        driver.findElement(By.id("login-username")).click();
        driver.findElement(By.id("login-username")).sendKeys("admin");
        driver.findElement(By.id("login-password")).click();
        driver.findElement(By.id("login-password")).sendKeys("123");
        driver.findElement(By.id("login-submit")).click();
        {
            WebDriverWait wait = new WebDriverWait(driver, 30);
            wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
        }
        driver.findElement(By.id("roles-link")).click();
        driver.findElement(By.cssSelector(".role-item:nth-child(3)")).click();
        driver.findElement(By.id("remove-article-id")).click();
        driver.findElement(By.id("remove-article-id")).sendKeys("uno");
        driver.findElement(By.id("remove-article-id")).sendKeys(Keys.ENTER);
        assertThat(driver.findElement(By.id("remove-article-result")).getText(), is("Formato ID errato."));
    }

    @Test

```



```

public void rimuoviArticoloArticoloInesistente() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1898, 1018));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("admin");
    driver.findElement(By.id("login-password")).click();
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-submit")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
    }
    driver.findElement(By.id("roles-link")).click();
    driver.findElement(By.cssSelector(".role-item:nth-child(3)")).click();
    driver.findElement(By.id("remove-article-id")).click();
    driver.findElement(By.id("remove-article-id")).sendKeys("10");
    driver.findElement(By.cssSelector(".button:nth-child(2)")).click();
    assertThat(driver.findElement(By.id("remove-article-result")).getText(), is("Articolo non esistente"));
}
}

```

```

public class RimuoviTitoloTest {
    private WebDriver driver;
    private Map<String, Object> vars;
    JavascriptExecutor js;
    @Before
    public void setUp() {
        System.setProperty("webdriver.chrome.driver", "../commons/driver/chromedriver.exe");
        driver = new ChromeDriver();
        js = (JavascriptExecutor) driver;
        vars = new HashMap<String, Object>();
    }
    @After
    public void tearDown() {
        driver.quit();
    }
    @Test
    public void rimuoviTitoloFormatoIDErrato() {
        driver.get("http://localhost:8080/shodan_maven/");
        driver.manage().window().setSize(new Dimension(1898, 1018));
        driver.findElement(By.id("login-username")).click();
        driver.findElement(By.id("login-username")).sendKeys("admin");
    }
}

```

```

driver.findElement(By.id("login-password")).click();
driver.findElement(By.id("login-password")).sendKeys("123");
driver.findElement(By.id("login-submit")).click();
{
    WebDriverWait wait = new WebDriverWait(driver, 30);
    wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
}
driver.findElement(By.id("delete-game-id")).click();
driver.findElement(By.id("delete-game-id")).sendKeys("abc");
driver.findElement(By.cssSelector(".button:nth-child(4)")).click();
assertThat(driver.findElement(By.id("delete-game-result")).getText(), is("Formato ID errato"));
}

@Test
public void rimuoviTitoloIDMancante() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1898, 1018));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("admin");
    driver.findElement(By.id("login-password")).click();
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-submit")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
    }
    driver.findElement(By.id("delete-game-id")).click();
    driver.findElement(By.cssSelector(".button:nth-child(4)")).click();
    assertThat(driver.findElement(By.id("delete-game-result")).getText(), is("ID Gioco: campo obbligatorio"));
}

@Test
public void rimuoviTitoloOK() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1898, 1018));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("admin");
    driver.findElement(By.id("login-password")).click();
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-submit")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
    }
}

```

```

driver.findElement(By.id("delete-game-id")).click();
driver.findElement(By.id("delete-game-id")).sendKeys("1");
driver.findElement(By.cssSelector(".button:nth-child(4)")).click();
assertThat(driver.findElement(By.id("delete-game-result")).getText(), is("Gioco eliminato con successo!"));
}

@Test
public void rimuoviTitoloTitoloInesistente() {
    driver.get("http://localhost:8080/shodan_maven/");
    driver.manage().window().setSize(new Dimension(1898, 1018));
    driver.findElement(By.id("login-username")).click();
    driver.findElement(By.id("login-username")).sendKeys("admin");
    driver.findElement(By.id("login-password")).click();
    driver.findElement(By.id("login-password")).sendKeys("123");
    driver.findElement(By.id("login-submit")).click();
    {
        WebDriverWait wait = new WebDriverWait(driver, 30);
        wait.until(ExpectedConditions.presenceOfElementLocated(By.id("roles-link")));
    }
    driver.findElement(By.id("delete-game-id")).click();
    driver.findElement(By.id("delete-game-id")).sendKeys("999");
    driver.findElement(By.cssSelector(".button:nth-child(4)")).click();
    assertThat(driver.findElement(By.id("delete-game-result")).getText(), is("Il titolo non e\\\' presente"));
}
}

```