SOFTWARE VERIFICATION, VALIDATION AND TESTING

TESTING DOCUMENTATION

CineStar Cinemas

Prepared by:

**Sanjin Biogradlić**

Proposed to:

Samed Jukić, Assist. Prof. Dr.

Aldin Kovačević, Teaching Assistant

Sarajevo, 22 januar 2023..

TABLE OF CONTENTS

[1. Introduction 3](#_Toc125322761)

[1.1. About the Project 3](#_Toc125322762)

[Project Functionalities and Screenshots 3](#_Toc125322763)

[ Rotator 3](#_Toc125322764)

[ Selection or filtering 4](#_Toc125322765)

[2. Test Plan 5](#_Toc125322766)

[2.1. Scope 5](#_Toc125322767)

[2.2. Testing Environment and Tools 5](#_Toc125322768)

[3. Test Execution 6](#_Toc125322769)

[3.1. Test Scenario: User profile 7](#_Toc125322770)

[3.2. Test Scenario: User login 8](#_Toc125322771)

[3.3. Test User log out 9](#_Toc125322772)

[3.4. Test “Odaberi kino” drop-down list 10](#_Toc125322773)

[3.5. Test “Odaberi film” drop-down list 11](#_Toc125322774)

[3.6. Test Test search box 12](#_Toc125322775)

[3.7. Test “Odaberi kino” + “Odaberi film” - filtering combination 13](#_Toc125322776)

[3.8. Test selection of cinema screenings 14](#_Toc125322777)

[1. Conclusion 15](#_Toc125322778)

[Testing Summary 15](#_Toc125322779)

[Final Thoughts 15](#_Toc125322780)

# Introduction

## 

## 1.1. About the Project

The name of the project undergoing testing is **CineStar Cinemas**. It is the website of one of the leading cinema chains in the region. I will test and focus only on the page created for BiH.

The site is great, that is, there are many different testing options. There is detailed information about CineStar including movie schedule, benefits, services, loyalty programs, subscriptions. In addition, it is very functional, easy to use, which is the goal for easier buying or reservation of tickets.

Its link to homepage: <https://www.blitz-cinestar-bh.ba/>.

## Project Functionalities and Screenshots

I will work on testing technique in which Graphical User Interface (GUI), functionality and usability of web applications are tested, that is front-end testing.

The goal of this Front-end testing is testing overall or as much as possible functionalities to ensure to have an insight into their accuracy.

The main parts the comprise this homepage at the front-end level are:

### Rotator

****

On the left are five cards that classify self-rotating ads. Each classification has five advertisements, the label for the same as the manual switch is marked with a red arrow in the picture. Also, each of them is changeable, the projections change every week.

### Selection or filtering



Every page has this section, not just the homepage, except for buying or booking a ticket. That is, changing the option of a movie or cinema is possible at any time, except when it comes to choosing a seat. Before that, you had to choose the cinema, the movie, the day and the time of the screening. It is possible to filter them only individually, ie the first filtering according to one and then the other criteria. After each filtering, this selective bar remains visible.

**Overall page layout.**

****

# Test Plan

## 2.1. Scope

What this test documentation will contain are:

* test scenarios: different use cases that an ordinary customer might encounter using the website. Also used as a classification name for test cases.
* test cases: a detailed tabular description of all test cases performed, along with pre-condition(s), steps, inputs, expected and actual results, and their final status.
* test executions: screenshots of all written tests.

The first cinema in Bosnia and Herzegovina was opened by this company in 2012. So that the website has been upgraded for a long time and is very extensive, I will try to cover all the essential functionalities, but not all.

The test plan will not cover the testing of static resources, such as images and element texts, except in the areas where they prove that a desired functionality has triggered.

## 2.2. Testing Environment and Tools

Front-end tests of this web application will be conducted via the Selenium framework for Java, and contained within JUnit test cases. The tests will be written and run in Eclipse Java IDE 2021-09 (4.21.0).

The web driver will use the Google Chrome executable.

# Test Execution

The beginning of each code:



## 3.1. Test Scenario: User profile

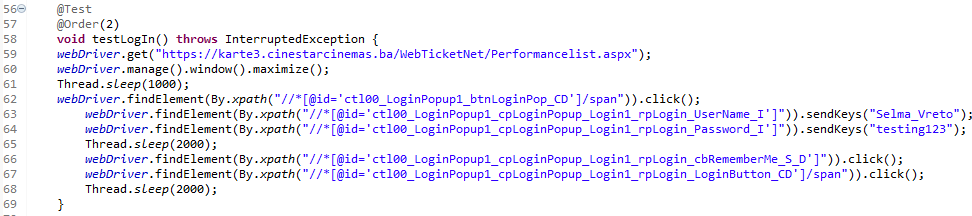
**Description:** A user who is not registered and logged in is not able to buy or reserve a ticket. The following tests will prove the functionality of both user registration and login.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **. Test Name: User registration** | | | | |
| **Description:** It is necessary to check the functionality of all steps during user registration. | | | | |
| **Pre-condition(s):**  Google Chrome as web driver is loaded. List of movies link is loaded. | | | | |
| **Test Steps:**   1. Go to the List of movies link 2. Click on “Prijava” button 3. Click on “Registracija” text link 4. Check if the URL of the opened page corresponds to the preferred option 5. Complete the form 6. Click on “Registracija” button | **Test Data:**   1. “Prijava” button 2. “Registracija” text link | **Expected Result:**  The user fills in the form correctly and creates a profile. | **Actual Result:**  The user fills in the form correctly and creates a profile. | **Status:**  PASS |
| **Notes:** It is necessary to confirm the profile by e-mail, but we will not do this to load the CineStar page and to make the code correct during other launches. | | | | |



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Name: User login** | | | | |
| **Description:** Check all the functionalities that we come across when user want to log in | | | | |
| **Pre-condition(s):**  Google Chrome as web driver is loaded. List of movies link is loaded.  The user must be registered. | | | | |
| **Test Steps:**   1. Go to the List of movies link 2. Click on “Prijava” button 3. Click on “User name” and fill in it 4. Click on “Password” and fill in it 5. Click on “Remember login” 6. Click on “Prijava” button | **Test Data:**   1. List of movies link 2. “Prijava” button | **Expected Result:**  The user fills in the form correctly and log in. | **Actual Result:**  The user fills in the form correctly and log in. | **Status:**  PASS |
| **Notes:** | | | | |

## 3.2. Test Scenario: User login



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Name:** **User log out** 3.3. Test User log out | | | | |
| **Description:** Check all the functionalities that we come across when user want to log out | | | | |
| **Pre-condition(s):**  Google Chrome as web driver is loaded. List of movies link is loaded.  The user must be registered. The user must be logged in. | | | | |
| **Test Steps:**   1. Go to the List of movies link 2. Click on “Prijava” button 3. Click on “User name” and fill in it 4. Click on “Password” and fill in it 5. Click on “Prijava” button 6. Click on “Odjava” button | **Test Data:**   1. List of movies link 2. “Prijava” button 3. “Odjava” button | **Expected Result:**  The user fills in the form correctly,  logged in and then log out. | **Actual Result:**  The user fills in the form correctly and logged in, then  when it is necessary to press the logout button the test crashes. | **Status:**  FAIL |
| **Notes:** Due to the overlap of two texts and buttons, the test fails. Because the manual button does not react on some parts. | | | | |

## 

## 3.4. Test “Odaberi kino” drop-down list

**Description:** Users want to be able to only see products from a category. There are different types of filtering, but it is possible to have several at once.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Name: Test “Odaberi kino” drop-down list** | | | | |
| **Description:** Check that the drop-down menu opens when clicked and that it leads to a selection of cinemas by city. | | | | |
| **Pre-condition(s):** Google Chrome as web driver is loaded. | | | | |
| **Test Steps:**   1. Go to the BaseUrl page 2. Check if all cities were dispayed. 3. Click on each city link individually 4. Check if the URLs of the opened pages correspond to the links. | **Test Data:**   1. homepage 2. the cities URLs | **Expected Result:**  The user is taken to the correct new  page when clicking on the selected  city. | **Actual Result:**  The user is taken to the correct new page when clicking on the selected city. | **Status:**  PASS |
| **Notes:** | | | | |



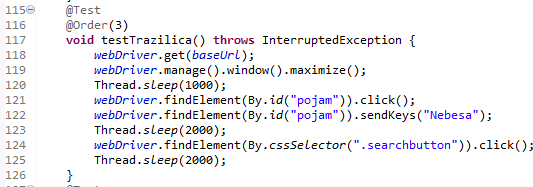
## 3.5. Test “Odaberi film” drop-down list

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Name: Test “Odaberi film” drop-down list** | | | | |
| **Description:** Check that the drop-down menu opens when clicked and that it leads to a selection of movies. | | | | |
| **Pre-condition(s):** Google Chrome as web driver is loaded. | | | | |
| **Test Steps:**   1. Go to the BaseUrl page 2. Check if all movies are dispayed. 3. Click on each city link individually 4. Check if the URL of the opened page corresponds to the link | **Test Data:**  1.homepage 2.the movies URLs | **Expected Result:**  The user is taken to the correct new  page when clicking on the movie. | **Actual Result:**  The user is taken to the correct new page when clicking on the movie. | **Status:**  PASS |
| Notes: Due to the huge number of films from all cinemas in BIH, we checked only a few. Movies in cinemas aren’t permanent, new movies are coming out, so if you try to run code in the future you might run into mistakes. | | | | |



## 3.6. Test Test search box

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Name: Test search box** | | | | |
| **Description:** By entering a keyword, the user wants to select movies. After entering the word, click on the icon leads to the recommended movies page. | | | | |
| **Pre-condition(s):** Google Chrome as web driver is loaded. | | | | |
| **Test Steps:**   1. Go to the BaseUrl page 2. Enter a keyword in the field (Nebesa) 3. Click on the browser icon 4. Check if the URL of the opened page corresponds to the preferred option | **Test Data:**  1.homepage link 2.” trazilica” URL | **Expected Result:**  The user has the ability to enter words. The user is taken to the correct  new page where the movies are selected. | **Actual Result:**  The user has the ability to enter words. The user is taken to the correct  new page where the movies are selected. | **Status:**  PASS |
| **Notes:** | | | | |



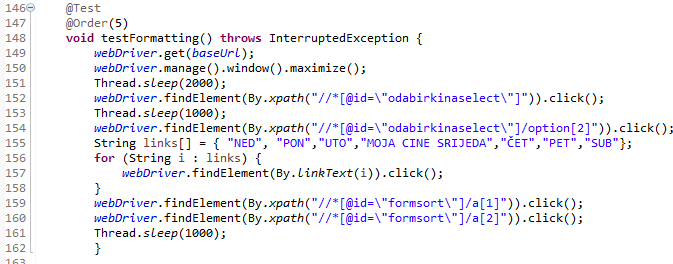
## 3.7. Test “Odaberi kino” + “Odaberi film” - filtering combination

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **. Test Name: Test “Odaberi kino” + “Odaberi film” - filtering combination** | | | | |
| **Description:** By clicking, the user can individually select both the movie and the cinema. | | | | |
| **Pre-condition(s):** Google Chrome as web driver is loaded. | | | | |
| **Test Steps:**   1. Go to the BaseUrl page 2. Click on the cinema you want (Sarajevo) 3. Click on the film you want (100% Vuk) 4. Check if the URL of the opened page corresponds to the preferred option | **Test Data:**   1. homepage link 2. movie URL 3. cinema URL | **Expected Result:**  The user chooses  the cinema and the movie of his choice, the selection takes him to the appropriate place. | **Actual Result:**  The user chooses the cinema and the  movie of his choice, the selection takes him to the appropriate place. | **Status:**  PASS |
| **Notes:** We took a random movie and a cinema because we checked the links individually above. | | | | |



## 3.8. Test selection of cinema screenings

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Name: Test selection of cinema screenings** | | | | |
| **Description:** By choosing a cinema, we will check the formats and days of screenings films. | | | | |
| **Pre-condition(s):** Google Chrome as web driver is loaded. | | | | |
| **Test Steps:**   1. Go to the BaseUrl page 2. Click on random city (Sarajevo) 3. Click and check each days link 4. Click and check select by format 5. Click and check screening time | **Test Data:**  1.homepage 2.the city URLs  3. the movie URLs | **Expected Result:**  The user can filter movies by days, format or screening time. | **Actual Result:**  The user can filter movies by days, format or screening time. | **Status:**  PASS |
| **Notes:** | | | | |



# Conclusion

## Testing Summary

|  |  |  |  |
| --- | --- | --- | --- |
| **Testing Tool** | **Total Tests** | **Passed Tests** | **Failed Tests** |
| the Selenium framework for Java, and contained within JUnit test cases | 8 test cases | 7 passed test | 1 failed test [LogOut](#_bookmark29) |

# Final Thoughts

During the execution of the testing plan, I concluded that this site is not very suitable for this project. It has great features that have been testing material for me. I would point out the registration, login. Although, as a flaw and why I think the site is not suitable, I would single out the daily updates. If some parts of the code are run after a certain period, errors may occur.

Since some functionalities are repeated, e.g. you can test that a link from every movie will lead you to buy or book tickets, and then testing every link to a movie in all cinemas in BiH, I think it was superfluous.

All in all, I am satisfied with my work and tests, but also with the website. Only one test fails and the only thing I found was manual testing. The greeting message after registration, the logout button and a short description of the film overlap. If you want to log out, the button accepts clicking only on the far right.

# 