**SOFTWARE VERIFICATION, VALIDATION AND TESTING**

TESTING DOCUMENTATION

Lunatik.io

Prepared by:

**Naida Fatić**

**Mirza Krupić**

Proposed to:

**Samed Jukić, Assist. Prof. Dr.**

**Aldin Kovačević, Teaching Assistant**

15.01.2022

TABLE OF CONTENTS

Contents

[1. Introduction 3](#_Toc91440922)

[1.1. About the Project 3](#_Toc91440923)

[1.2. Project Functionalities and Screenshots 3](#_Toc91440924)

[2. Test Plan 3](#_Toc91440925)

[2.1. Scope 3](#_Toc91440926)

[2.2. Testing Environment and Tools 3](#_Toc91440927)

[3. Test Execution 3](#_Toc91440928)

[3.1. Test Scenario Name 3](#_Toc91440929)

[3.2. Example Scenario: Visit the Home Page 4](#_Toc91440930)

[9. Conclusion 5](#_Toc91440931)

[9.1. Testing Summary 5](#_Toc91440932)

[9.2. Final Thoughts 5](#_Toc91440933)

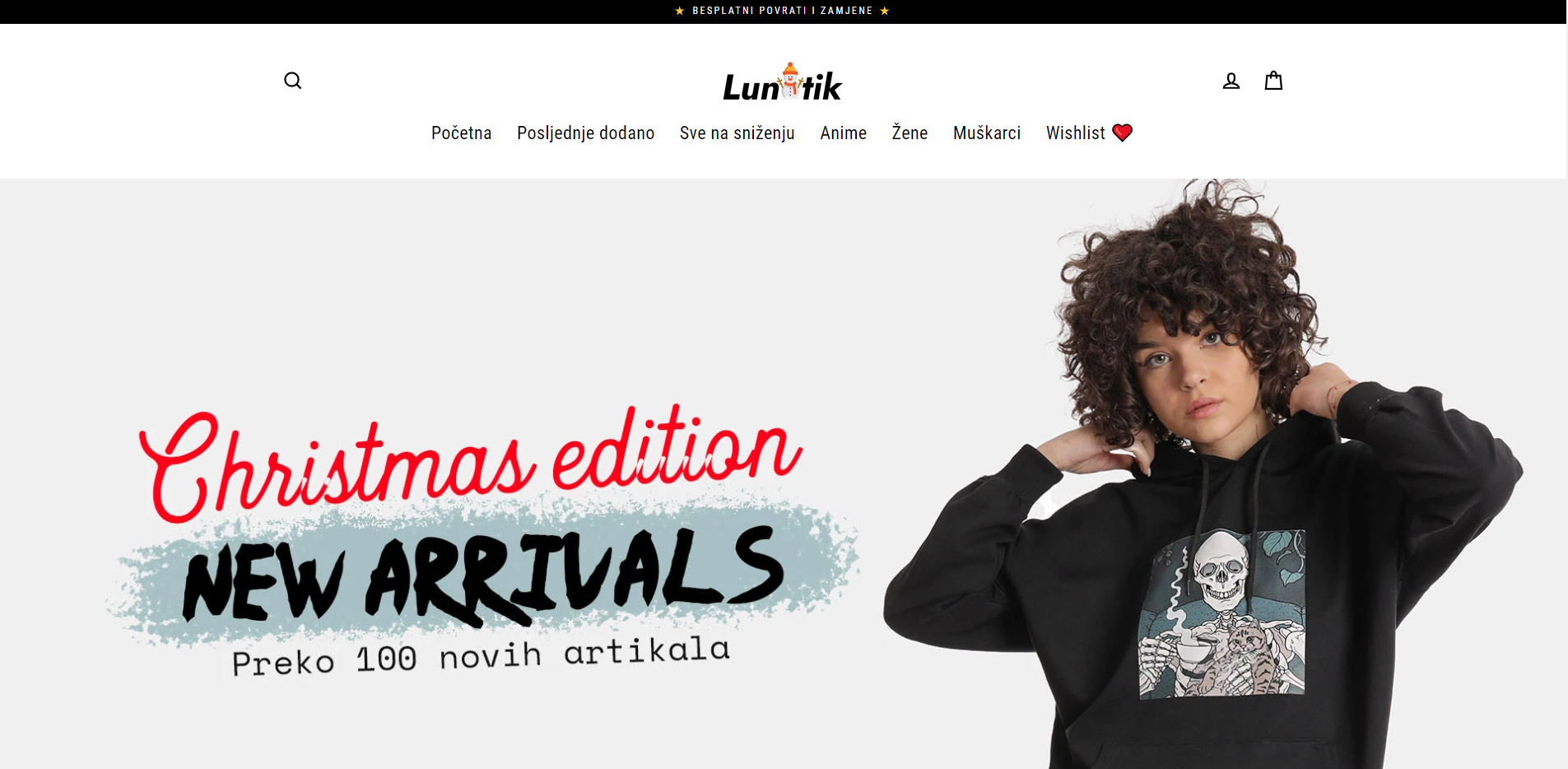
# 1. Introduction

## 1.1. About the Project

On this project we will work on [Lunatik.io](https://lunatik.io/) website. We got this idea from the previous shopping experience with this website, Naida made some orders via this site before. We will test most of the features just like we would do in real life.

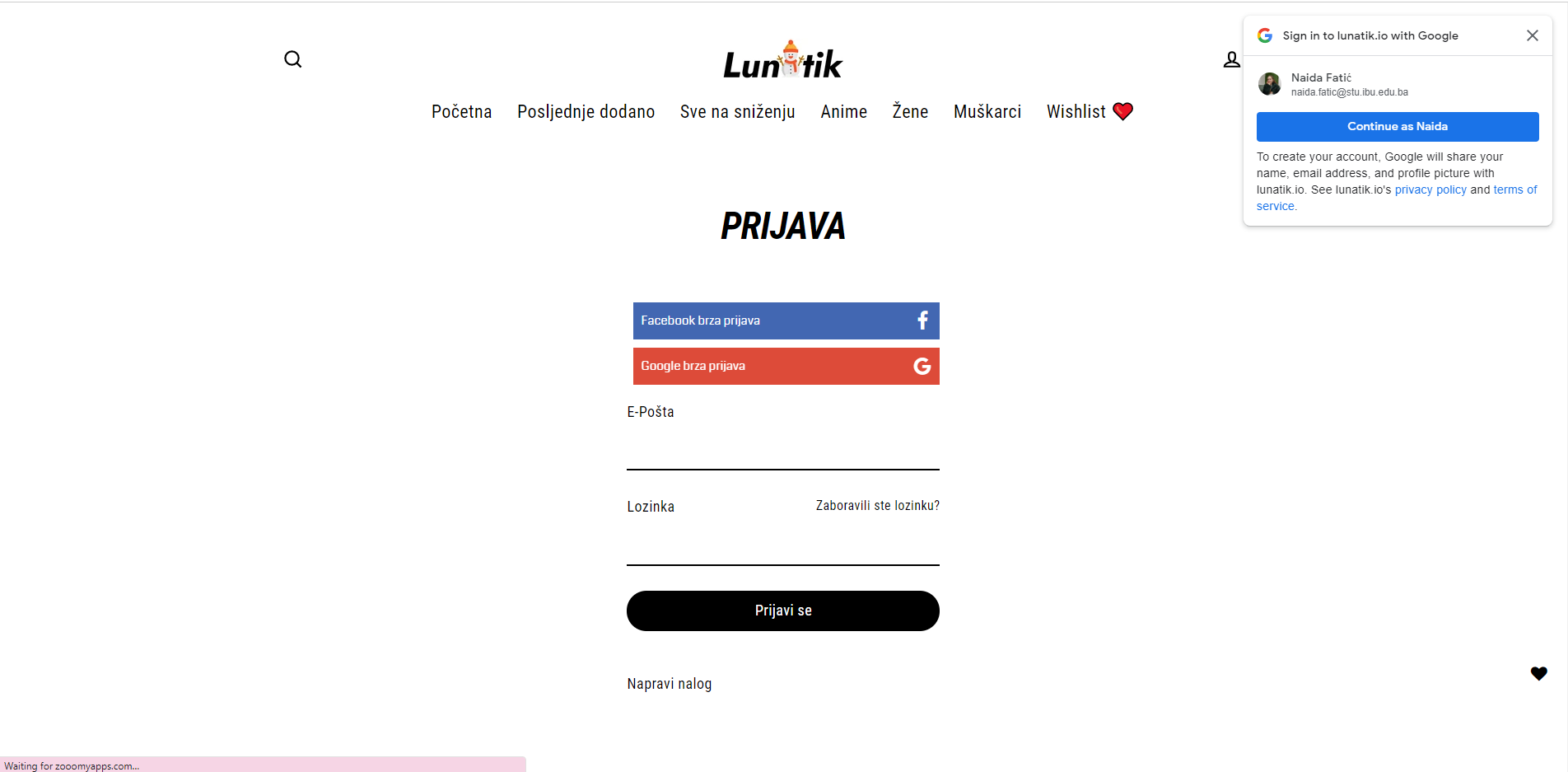
## 1.2. Project Functionalities and Screenshots

Search of some items or categories

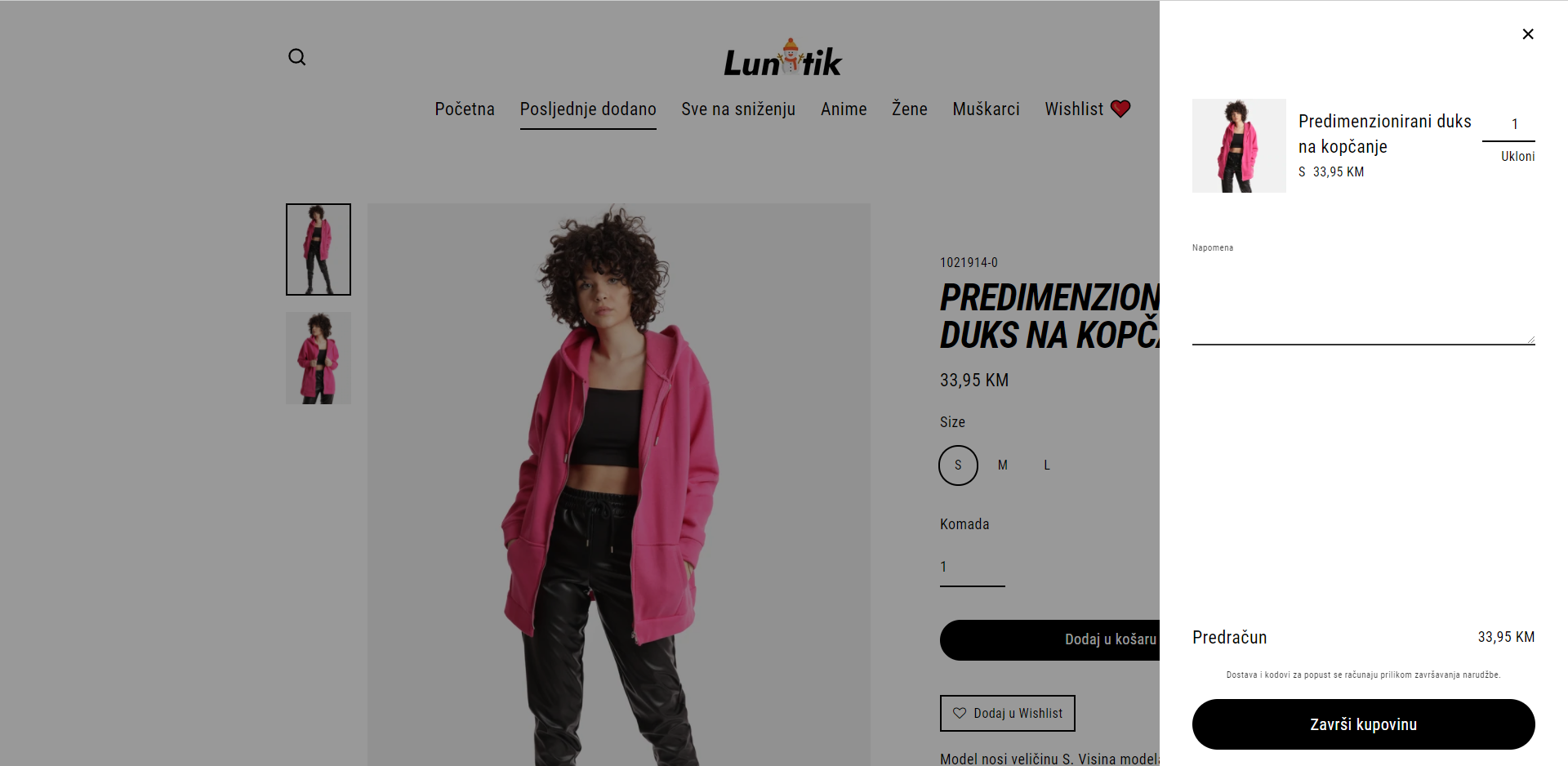




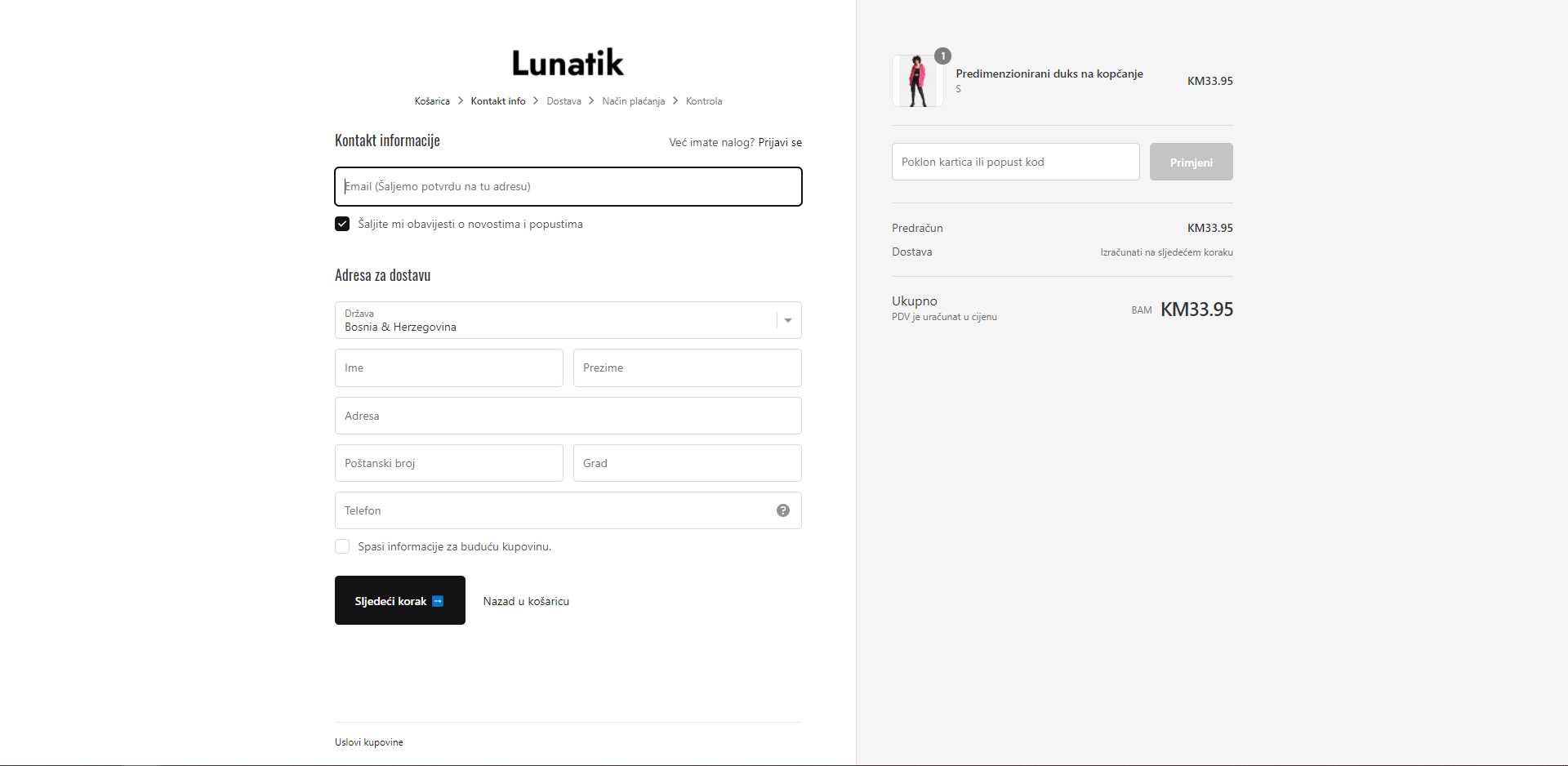
Create an account



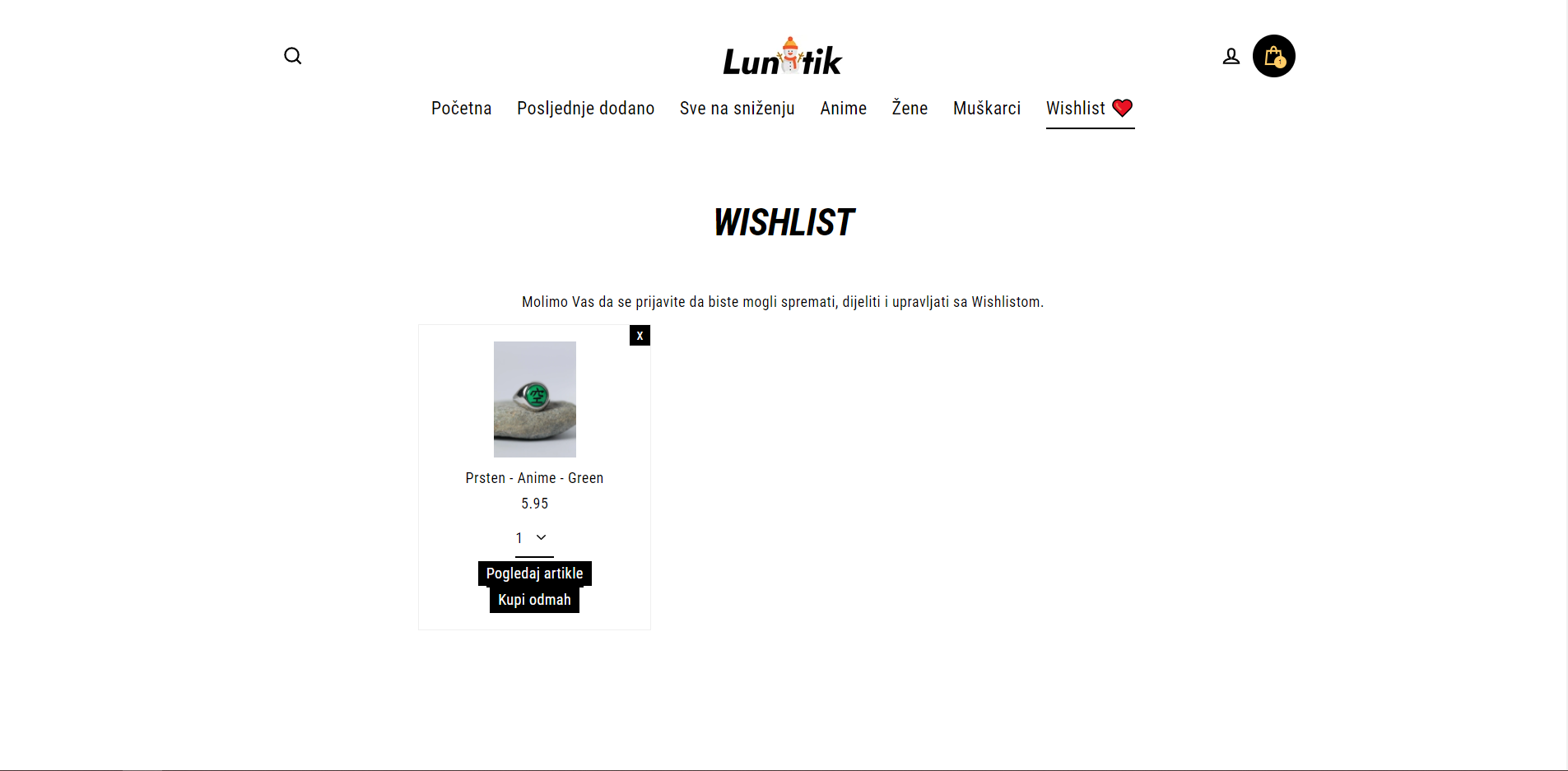
Cart



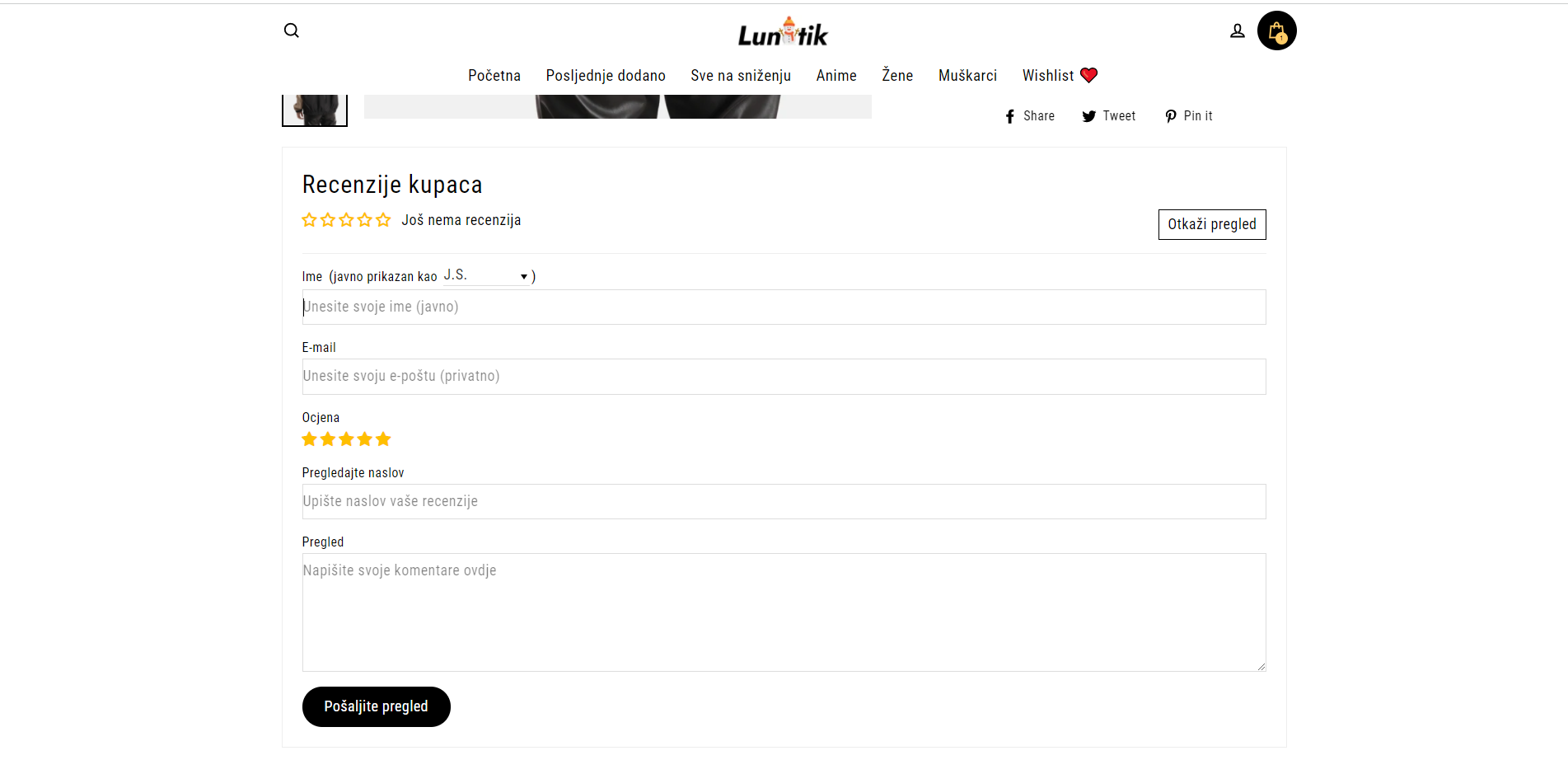
Checkout



Wishlist



Review of the product



# 2. Test Plan

## 2.1. Scope

Focus on this test will be on the shopping experience. We will test if the website is working properly: if we can go to all of the category links, if the prices are correctly added to the cart, we will also proceed to the checkout but we will not test the checkout feature. Also, we will test the Wishlist and add to cart feature. We are planning just to test desktop version (maximize window).

## 2.2. Testing Environment and Tools

We will be using Selenium tool for the test and write the tests in Java language.

# 3. Test Execution

## 3.1. Visit the Home Page

Users want to be able to visit the home page, and get a quick glance at the most important features it has to offer.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Name:** Test top bar links | | | | |
| **Description:**  Check if all top bar items are clickable and lead to their respective page. Click links one after another. | | | | |
| **Pre-condition(s):** | | | | |
| **Test Steps:**  1. Go to the landing page  2. Check title of page  3. Check if all links have been rendered  4. Click on each link individually  5. Check if the URL of the opened page is correct | **Test Data:** | **Expected Result:**  The user is taken to the correct new page when clicking on the link. | **Actual Result:**  The user is taken to the correct new page when clicking on the link. | **Status:**  PASS |
| **Notes:** | | | | |

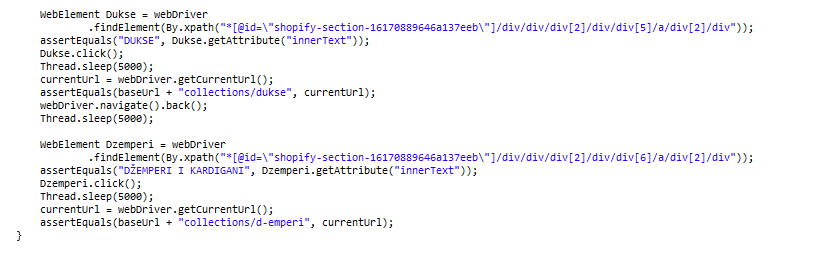


## 3.2. Test “Collections” section links

Users want to be able to visit the home page and to click on the links that home page offer in the collection section that is at the bottom of the page.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Name:** Test “Collections” section links | | | | |
| **Description:**  Check if link on “Collections” section are clickable and if they take user to the correct page. | | | | |
| **Pre-condition(s):** | | | | |
| **Test Steps:**  1. Go to the landing page  2. Check title of page  3. Scroll down to the “Collections” section  4. Check if all links have been rendered  5. Click on each link individually  6. Check if the URL of the opened page is correct | **Test Data:** | **Expected Result:**  “Collections” section is rendered properly and user is taken on the correct page when clicking each of the links in this section. | **Actual Result:**  “Collections” section is rendered properly and user is taken on the correct page when clicking each of the links in this section. | **Status:**  PASS |
| **Notes:** | | | | |





## 3.3. Search

Users want to be able to search some items in order to see faster what site offers.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Name:** Test search feature | | | | |
| **Description:**  Check if search feature works both for items which exist and items which doesn’t exist in the database. | | | | |
| **Pre-condition(s):** | | | | |
| **Test Steps:**  1. Go to the landing page  2. Click on the search button in the upper left corner  3. Input search parameter for item which exists in the database  4. Click on the search button  5. Check if system returned any results  6. Click on the search button in the upper left corner  7. Input search parameter for item which doesn’t exists in the database  8. Click the enter button  9. Check if system returned any results | **Test Data:**  “majice” and “aass” text we used for the search input. | **Expected Result:**  Search should return certain amount of items if item’s name contains phrase we entered in the search field or it should not return any results if there is no item which contains in its name phrase we searched for. | **Actual Result:**  Search returns certain amount of items if item’s name contains phrase we entered in the search field or it doesn’t not return any results if there is no item which contains in its name phrase we searched for. | **Status:**  PASS |
| **Notes:** | | | | |



## 3.4. Item sort

Users want to be able to sort items by some value in order to search see items.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Name:** Test item sorting | | | | |
| **Description:**  Check if sorting on shop pages works properly. | | | | |
| **Pre-condition(s):** | | | | |
| **Test Steps:**  1. Go to the “Posljednje dodano” page  2. Check current items  3. Select next option in dropdown menu  4. Check if items in the item list changed  5. Repeat the process for every menu item | **Test Data:** | **Expected Result:**  Whenever we change option in the dropdown menu, item list should change order of items rendered. | **Actual Result:**  Whenever we change option in the dropdown menu, item list changes order of items rendered. | **Status:**  PASS |
| **Notes:** | | | | |



## 3.5. Item page

Users want to see items in order to get their description, add them to cart, add them to Wishlist or to leave review.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Name:** Test item page | | | | |
| **Description:**  Check if item page works properly. | | | | |
| **Pre-condition(s):** | | | | |
| **Test Steps:**  1. Go to the home page.  2. Navigate to the “Muškarci” link on the top bar.  3. Hover over it and hover over “Nakit” link  4. Click on the “Prstenje” link  5. Check if the title of the page says “Muško prstenje”  6. Check if the price of first item is 15.95  7. Click on the first item and check the price of the item if it is 15.95  8. Add item to the cart  9.Close cart asset that cart number is 1  10. Add item to the Wishlist and check if the button says “Ukloni iz Wishlist”  11. At the bottom of the page we can go back to the previous page we are going to test that button | **Test Data:** | **Expected Result:**  The “Muško prstenje” page should appear, we should be able to click on the item and add it to the cart or Wishlist. After all of that we want to be able to go back to the “Muško prstenje” page. | **Actual Result:**  The page loaded, we are able to click on the item and add it to the cart or Wishlist. The cart number incremented. We were able to go back to the “Muško prstenje” page. | **Status:**  PASS |
| **Notes:** | | | | |



## 3.6. Successful registration

In order for user to be able to add items to Wishlist, to save addresses and to track down history of his purchases, he has to register so he can use that account for keeping and accessing all of that information.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Name:** Test successful registration | | | | |
| **Description:**  Check if registration system passes when user enters valid data. | | | | |
| **Pre-condition(s): We have to input email that has not been used before.** | | | | |
| **Test Steps:**  1. Go to the login page by clicking account button in upper right corner.  2. Go to registration page by clicking on “Napravi nalog” located bellow login form  3. Fill in form with valid data  4. Click on “Napravi” button located bellow the form  5. Get redirected to home page while being logged in in the same time. | **Test Data:**  Email: [(randomly generated string)@veb65.com](mailto:dadirel637@veb65.com),  Password: “dummypassword”  Ime: “ime”,  Prezime: “prezime” | **Expected Result:**  User should be able to create new account on the webpage after entering valid data into the registration form and clicking “Napravi” button. | **Actual Result:**  User is able to create new account on the webpage after entering valid data into the registration form and clicking “Napravi” button. | **Status:**  PASS |
| **Notes:** | | | | |



## 3.7. Email already used during registration process

During the registration process, user might enter the email which is already used by some other account. In that case we want to test if the user gets error prompt which says that email is taken with link to go to password recovery page and we also need to test if that link works.

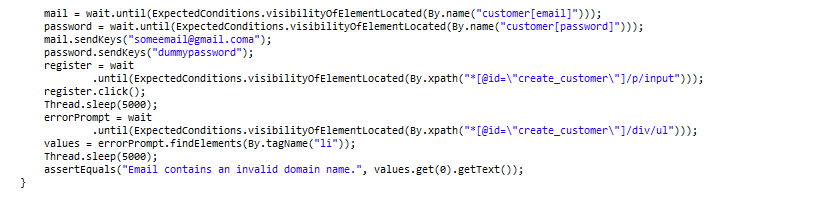
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Name:** Test email already used during registration | | | | |
| **Description:**  Check if system of detecting if user entered email which is already in used and test if link which sends user to password recovery page works. | | | | |
| **Pre-condition(s): We should have account already created with this email.** | | | | |
| **Test Steps:**  1. Go to the login page by clicking account button in upper right corner.  2. Go to registration page by clicking on “Napravi nalog” located bellow login form  3. Fill in form including email which is already used previously to create some other account  4. Click on “Napravi” button located bellow the form  5. Get error prompt which says that this email is in use and prompt includes link which should lead user to password recovery page  6. Click on password recovery link inside error prompt | **Test Data:**  Email : [dadirel637@veb65.com](mailto:dadirel637@veb65.com),  Password : “dummypassword”  Ime: “ime”,  Prezime: “prezime” | **Expected Result:**  User should be prompted with error that email is already in use when he clicks on “Napravi” button. When user clicks on “Resetuj lozinku” he should be redirected to password recovery page. | **Actual Result:**  User is prompted with error that email is already in use when he clicks on “Napravi” button. When user clicks on “Resetuj lozinku” he is not redirected to password recovery page. He is redirected to the URL which resembles password reset but that URL leads user to the homepage rather than password recovery page. | **Status:**  FAIL |
| **Notes:** | | | | |



## 3.8. Registration form validation

In order for user to be able to register successfully, developer should implement form validation for registration form in order to guide user to enter valid data inside the form.

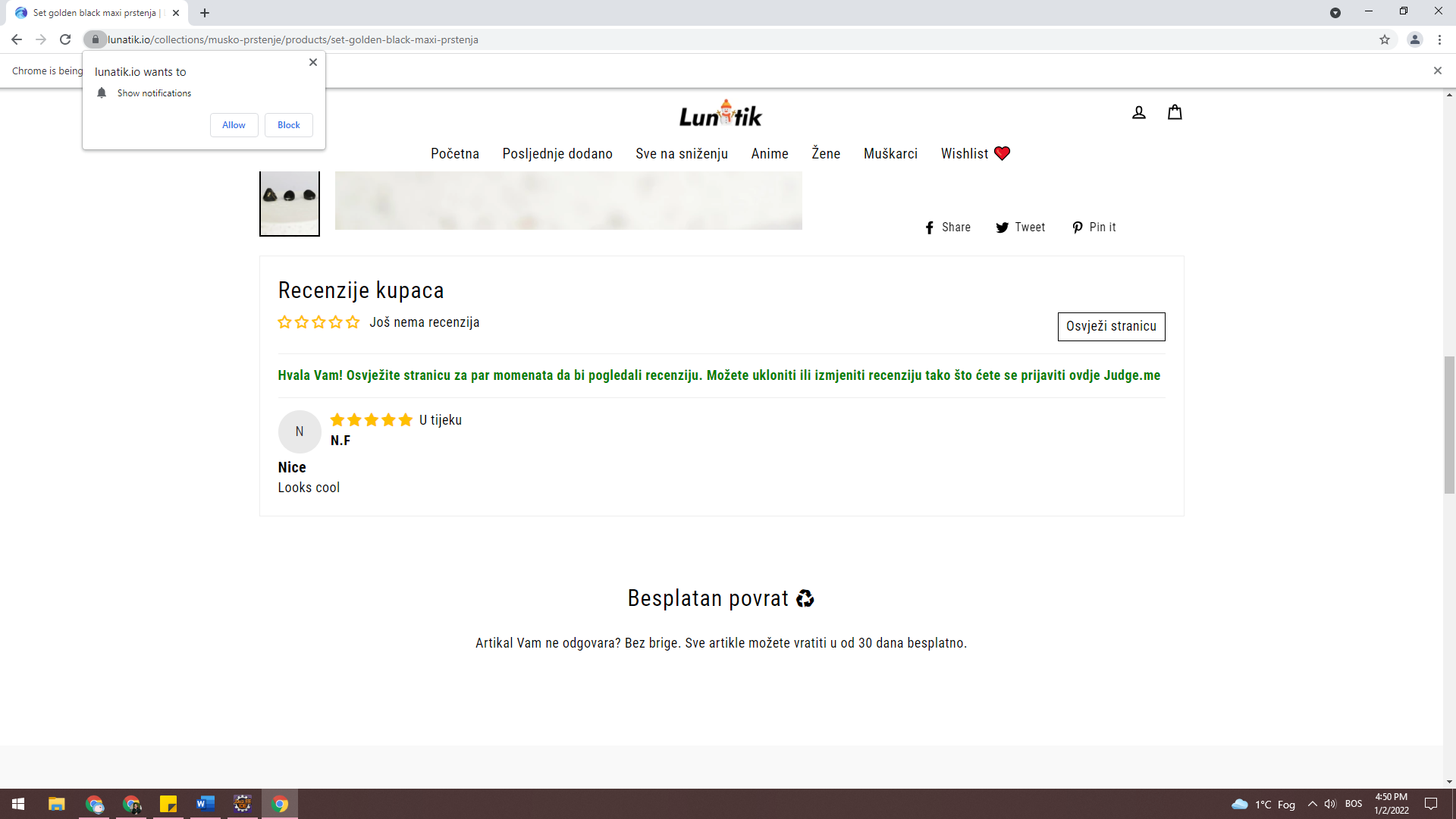
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Name:** Registration form validation test | | | | |
| **Description:**  Checking all possible invalid data scenarios inside registration form validation. | | | | |
| **Pre-condition(s):** | | | | |
| **Test Steps:**  1. Go to the login page by clicking account button in upper right corner.  2. Go to registration page by clicking on “Napravi nalog” located bellow login form  3. Fill in form without inputting any data  4. Click on “Napravi” button located bellow the form  5. Get redirected to the same page  6. Check error prompt  7. Fill in form without inputting email  8. Click on “Napravi” button  9. Get redirected to the same page  10. Check error prompt  11. Fill in form without inputting password  12. Click on “Napravi” button  13. Get redirected to the same page  14. Check error prompt  15. Fill in form by entering invalid email  16. Click on “Napravi” button  17. Get redirected to the same page  18. Check error prompt | **Test Data:**  Email: [dadirel637@veb65.com](mailto:dadirel637@veb65.com),  Password: “dummypassword”  Ime: “ime”,  Prezime: “prezime” | **Expected Result:**  When user tries to register with invalid data inside the form, he should not be allowed to do it and he should be prompted with the appropriate error prompts. | **Actual Result:**  When user tries to register with invalid data inside the form, he is not allowed to do it and he is prompted with the appropriate error prompts. | **Status:**  PASS |
| **Notes: This test works but when user enters invalid form of email, he is prompted with error message on English language while every other error prompt is in Bosnian which is not a good practice since this is supposed to be website for the Bosnian people.** | | | | |



## 3.9. Review comment

Users want to be able to leave review of item in order to state his/her opinion.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Name:** Test review of the item | | | | |
| **Description:**  Check if user can leave review on item. | | | | |
| **Pre-condition(s):** | | | | |
| **Test Steps:**  1. Go to item that we used in previous task.  2. Go to the bottom and click on the “Napisite svoje misljenje”  3. Populate all of the fields and press “Pošaljite pregled”  4. On purpose left the name part empty and assert that “Ovo polje je obavezno.” text appear.  5. After that we input name and send review | **Test Data:**  We user “N.F” for the name input, “[example@example.com](mailto:example@example.com)” for the email input, “Nice” text for the “Naslov” input and “Very cool” for the “Pregled” input. | **Expected Result:**  We should be able to left review and see our review after we refresh. | **Actual Result:**  The review is there we can see it on this [link](https://lunatik.io/collections/musko-prstenje/products/set-golden-black-maxi-prstenja). | **Status:**  PASS |
| **Notes: We are not going to write test that this review pass because we don’t want to spam page. The prof that this test passes is below and also on the link. If we want to write this test, we can do that with the assertEquals on the green text below.** | | | | |





## 3.10. Login successful

Users want to be able to login to their account in order to see the history or cart history.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Name:** Test login | | | | |
| **Description:**  Check if user can login into his account with correct information. | | | | |
| **Pre-condition(s):** | | | | |
| **Test Steps:**  1. Go to the login page  2. Populate all fields: email and password with an information that we used for the registration  4. Press login | **Test Data:**  Email: [dadirel637@veb65.com](mailto:dadirel637@veb65.com),  Password: “dummypassword” | **Expected Result:**  We should be able to login with this data and to be redirected to the account page. | **Actual Result:**  This test passed; we are able to login. | **Status:**  PASS |
| **Notes: We put a sleep, in order to click on captcha because it will appear after login.** | | | | |



## 3.11. Login with wrong credentials

Users can’t login until he enters right information.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Name:** Test login inputs | | | | |
| **Description:**  Check if user can login into his account with incorrect information. | | | | |
| **Pre-condition(s):** | | | | |
| **Test Steps:**  1. Go to the login page  2. Populate all fields: email and password with an information that we used for the registration  4. Press login | **Test Data:**  Email: “someemail@gmail.coma”,  Password: “dummypassword” | **Expected Result:**  We should not be able to login with this data and error message should appear. | **Actual Result:**  This test passed; we are not able to login. | **Status:**  PASS |
| **Notes: We put a sleep, in order to click on captcha because it will appear after login.** | | | | |



## 3.12. Wishlist page

Users wants to be able to see all of the items he added an to buy from Wishlist.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Name:** Test Wishlist page | | | | |
| **Description:**  Check if user can see Wishlist item he added. | | | | |
| **Pre-condition(s): User has to add the item to the Wishlist first.** | | | | |
| **Test Steps:**  1. Go to the Anime page  2. Go to the item on the next page  4. Press that item  5. Press to add to the Wishlist  6. Go to the Wishlist  7. Assert that item’s price is correct  8. Test the no. of items selected to buy | **Test Data:** | **Expected Result:**  We should be able to add item to Wishlist and to buy it from there | **Actual Result:**  We are able to add the item to the Wishlist but we can’t buy it from there | **Status:**  FAIL |
| **Notes: We tried to select the no. of items in the Wishlist but after we select it the “Kupi odmah” button don’t work. The test passes when we select another more expensive item. This may be some functionality of the page but user should be informed if he cannot purchase some item.** | | | | |



## 3.13. Save address

Users wants to be able to save address for later use when he is ordering his items.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Name:** Test save address feature | | | | |
| **Description:**  Check if user can see save address to his profile. | | | | |
| **Pre-condition(s): We have to have an account.** | | | | |
| **Test Steps:**  1. Go to the login page  2. Populate all fields: email and password with an information that we used for the registration  4. Press login  5. Press “Pogledaj adrese” on the right side of page  6. Press “Dodaj adresu” button in the top right corner of page  7. Fill in form  8. Save address by clicking on “Dodaj adresu” button | **Test Data:**  Login data:  Email: [dadirel637@veb65.com](mailto:dadirel637@veb65.com),  Password: “dummypassword”  Add address form data:  Ime prezime  IBU  Dummy Address 18  71000 Sarajevo  Bosnia & Herzegovina | **Expected Result:**  We should be able to add new address to our profile which we will be able to use later during purchases. | **Actual Result:**  We should be able to add new address to our profile which we will be able to use later during purchases. | **Status:**  PASS |
| **Notes: Dropdown menu where user should select Canton he lives in has only one option and it is Bosnia and Herzegovina which is country not Canton.** | | | | |



## 3.14. Edit address

Users wants to be able to edit one of his addresses for later use when he is ordering his items.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Name:** Test edit address feature | | | | |
| **Description:**  Check if user can edit address to his profile. | | | | |
| **Pre-condition(s): User has to add the address first and have an account.** | | | | |
| **Test Steps:**  1. Go to the login page  2. Populate all fields: email and password with an information that we used for the registration  4. Press login  5. Press “Pogledaj adrese” on the right side of page  6. Press “Uredi” button bellow the address  7. Fill in form  8. Save address by clicking on “Ažuriraj adresu” button | **Test Data:**  In this case we already have address entered and we are only changing number in address by removing last character and adding 12 to the end. | **Expected Result:**  We should be able to edit address on our profile which we will be able to use later during purchases. | **Actual Result:**  We are able to edit address on our profile which we will be able to use later during purchases. | **Status:**  PASS |
| **Notes:** | | | | |



## 3.15. Delete address

Users wants to be able to delete one of his addresses.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Name:** Test delete address feature | | | | |
| **Description:**  Check if user can delete address on his profile. | | | | |
| **Pre-condition(s): User has to be logged in and have the address added first.** | | | | |
| **Test Steps:**  1. Go to the account page  2. Press the “Pogledaj adrese (1)” button  3. Assert it takes you to next page  4. Press “Obriši” button  5. Press “OK” in alert message  6. Assert that address is deleted | **Test Data:** | **Expected Result:**  We should be able to delete address on our profile. | **Actual Result:**  We are able to delete address on our profile. | **Status:**  PASS |
| **Notes:** | | | | |



## 3.16. Cart price calculator

When user adds multiple items to his cart, he wants to see total price of those items at the bottom of the cart.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Name:** Test cart price calculator | | | | |
| **Description:**  Check if application calculates right price when we add multiple items to the cart. | | | | |
| **Pre-condition(s):** | | | | |
| **Test Steps:**  1. Go to the shop page  2. Go to item page of the first item in the list  3. Add item to cart  4. Go back one page  5. Go to item page of the second item in the list  6. Add item to cart  7. Check if calculated price is correct | **Test Data:** | **Expected Result:**  When we add more than one item to the cart, we should be presented with the right total price. | **Actual Result:**  When we add more than one item to the cart, we are presented with the correct total price. | **Status:**  PASS |
| **Notes:** | | | | |

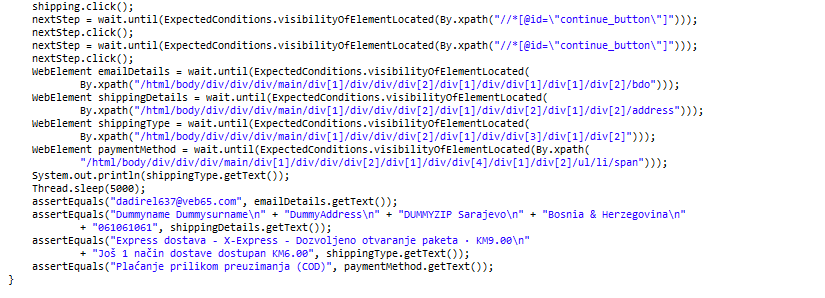


## 3.17. Checkout

After user adds every item, he desires to buy into the cart, he wants to be able to checkout and order those items.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Name:** Test checkout | | | | |
| **Description:**  Check if checkout feature works when user wants to buy selected items inside the cart | | | | |
| **Pre-condition(s):** | | | | |
| **Test Steps:**  1. Go to the shop page  2. Go to item page of the first item in the list  3. Add item to cart  4. Go back one page  5. Go to item page of the second item in the list  6. Add item to cart  7. Click “Završi kupovinu” button  8. Fill in form  9. Click “Sljedeći korak” button  10. Select second option  11. Click “Sljedeći korak” button  12. Click “Pregledaj narudžbu” button | **Test Data:**  Email: [dadirel637@veb65.com](mailto:dadirel637@veb65.com),  First name:  “Dummyname”,  Last name:  “Dummysurname”,  Address:  “DummyAddress”,  Zip:  “DUMMYZIP”,  City: “Sarajevo,  Country: “Bosnia & Herzegovina”,  Phone:  “061061061” | **Expected Result:**  When we add all items, we want to buy to the cart, we should be able to proceed to checkout and finish whole process. | **Actual Result:**  When we add all items, we want to buy to the cart, we are able to proceed to checkout and finish whole process. | **Status:**  PASS |
| **Notes: We are not able to complete whole purchase process since we would have to buy some items for real. We completed this process as far as we were possible to.** | | | | |





# 9. Conclusion

Test will fail sometimes because some unknown reason. It may fail because items may be added to the site or removed.

The xpath may change because items are added or removed or because of unknown reasons and test will fail.

Also, the price of the items may change, there may be a sale and the test may fail to assert correct value of the price.

All of the information that we have provided in the document are correct for the moment we have tested them.

## 9.1. Testing Summary

|  |  |  |  |
| --- | --- | --- | --- |
| **Testing Tool** | **Total Tests** | **Passed Tests** | **Failed Tests** |
| 1. Selenium, Java language | 17 | 15 | 2 |

Test failed:

1. [Wishlist page](#_3.12._Wishlist_page)
2. [Email already used during registration process](#_3.7._Email_already)

## 9.2. Final Thoughts

In conclusion the page looks good and it works but the structure of the page is not good. All of the items are loaded every time page is loaded and sometimes the xpath of same elements change for unknown reason.

To fix this problem we may introduce the single page functionality.

Also, page has some bugs like redirect or some text is in Bosnian and other is in English. These mistakes are easily fixable but they also very easy to make.

Page looks good and it is easy to navigate through the pages but the wait to load everything takes too long.

Application may need some changes when it comes to the UX because user doesn’t know some of the functionalities like in the Wishlist test. In that test “Kupi odmah” works if the purchase is a bit expensive but there is no text to indicate that.