Perfume's club

Beauty Lovers

Final Project

Alexandra Imre-Tamasanu

App description

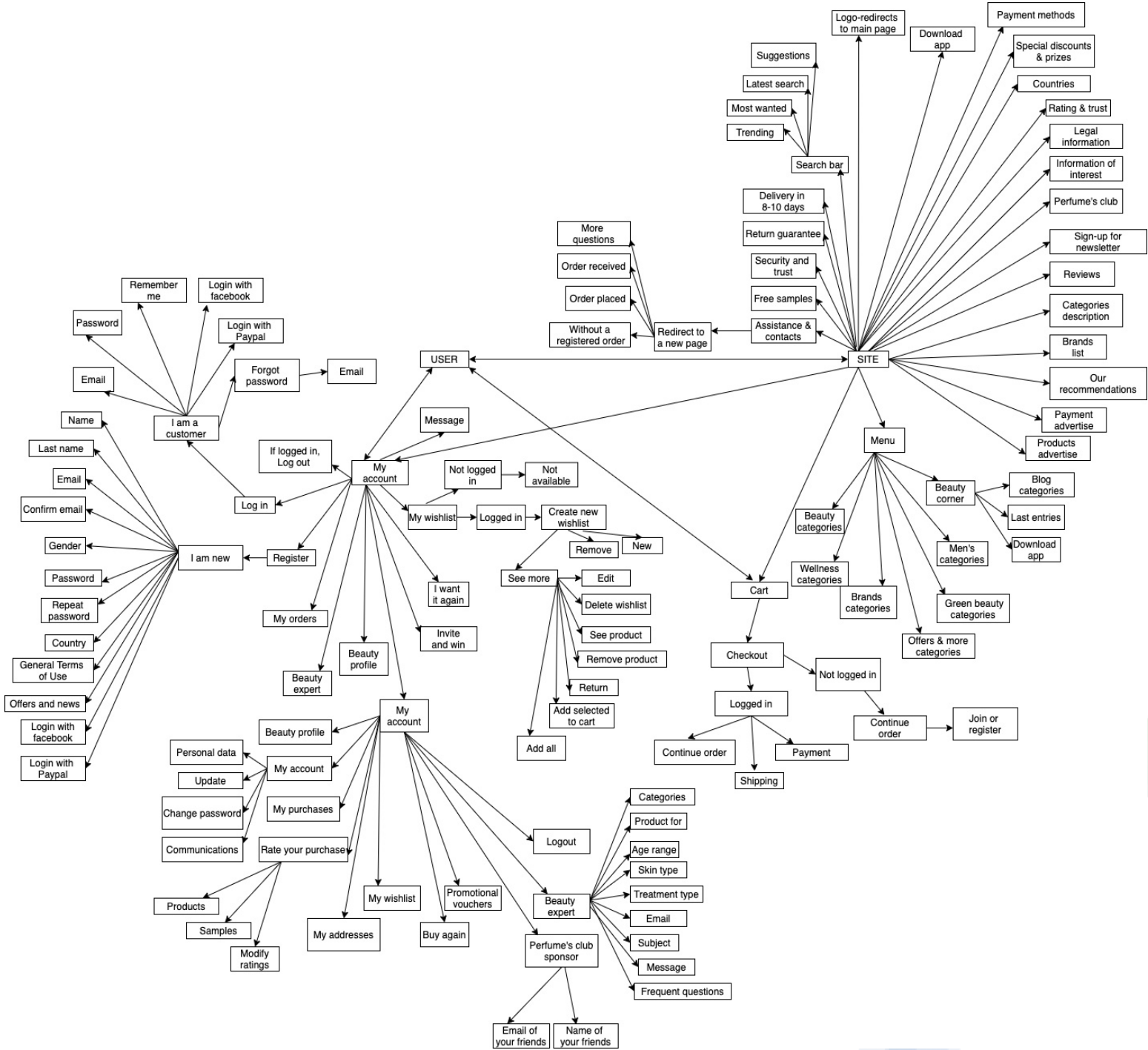
- www.perfumesclub.co.uk is an e-commerce website that offers products related to the world of beauty.
- It contains functional features such as register, login, search bar, and shopping cart, and non-functional features such as connectivity, localization, security, accessibility, usability, and performance.
- Users can use either their email and a password or PayPal or Facebook in the sign-up process.
- It has three payment gateways, like a credit card, PayPal, and bank transfer, which makes the website easy to use and offers a great checkout experience for customers.
- It is a cross-platform website compatible with multiple operating platforms (macOS and Windows) and a cross-browser website (Safari, Chrome, Edge).
- It is a website based on the location that displays information depending on the user's current location. The currency and language change when a specific country is selected.

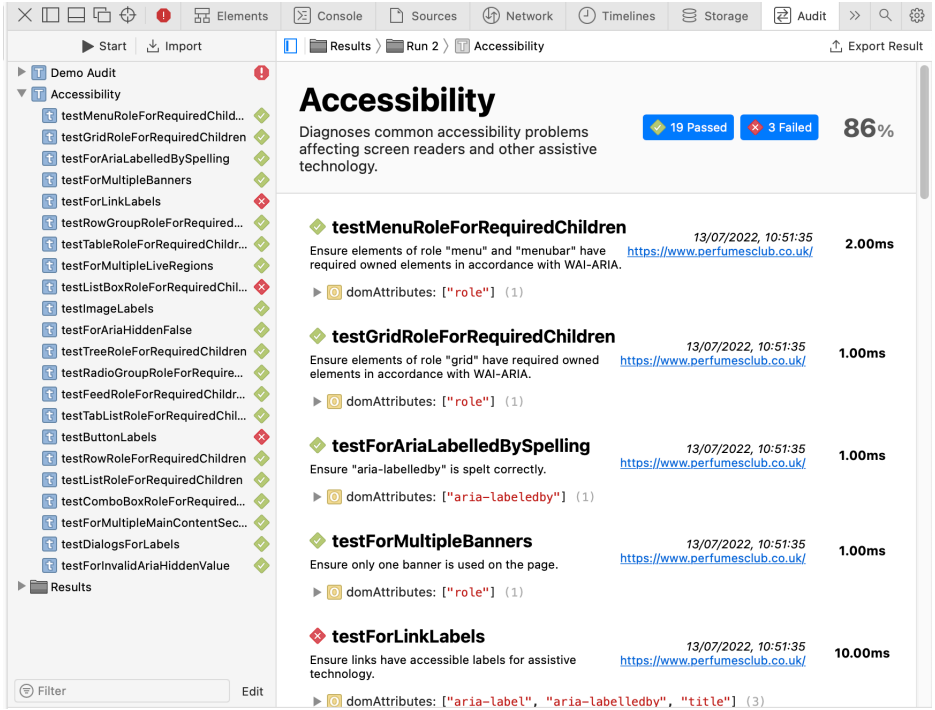
Testing approach

I approached testing using a mind map for a clear view of all functionalities and asking questions to clarify requirements to help create test cases.

Some of the questions asked are:

- The search bar should show history? The suggestions should be made by the most sought-after products? When the user starts typing a word in the text box it should suggest words that match the typed keyword?
- What language will be default for each country?
- Can products be added to the wishlist and cart without being logged in?
- When creating an account, can the user register with Facebook, or Instagram? What about login?
- The login section should have the option " Remember me"?
- With what OS or browser should the website be compatible?
- If an invalid email, name, last name, or password is entered should a message appear?
- If the connection is lost, the items from the cart and wish list will still be available?





Tools used

- Audit from Web Inspector on Safari and Lighthouse from DevTools on browsers Chrome and Edge for accessibility testing
- Lighthouse from DevTools for performance testing
- VM Parallels Desktop for compatibility testing
- TestLink for test cases
- Mantis for reporting bugs
- Microsoft Word for test report
- Microsoft Excel for test data
- MacBook Pro (13-inch, M1, 2020) as a device for testing

Testing types covered

•Smoke testing to verify that the most important functionalities are working

•Exploratory testing to find defects that might not be covered by other tests.

• Functional testing to tests each function of the software application, by providing appropriate input, verifying the output against the functional requirements

•Negative testing to ensure that application can handle the invalid input or unexpected user behavior.

•Localization testing to verify website's behavior, accuracy and suitability for specific countries.

•Performance testing to check the website's stability and response time.

•Security testing to check how the website is secure from internal and/or external threats and how secure& strong the authentication process is

•Accessibility testing to confirm that the application is usable for as many people as possible, including people with disabilities such as vision impairment, hearing problems and cognitive conditions.

•Usability testing to measure how easy and user-friendly application is.

•Compatibility testing to check whether the app is capable of running on different operating systems and browsers.

Test cases overview

420 test cases were planned regarding functional (login, register, shopping cart, search bar) and non-functional features (connectivity, localization, performance, security, accessibility, usability) using valid, invalid, and blank data for different operating systems and browsers.

Bugs overview

- The main defects found for each operating system and browser were in “login” feature(“forgot password” section) , accessibility (image elements do not have “alt” attributes, buttons do not have accessible labels, and list elements are not contained within parent elements), localization (language is not changing when selecting a certain country) and performance (short cache lifetime, excessive DOM size, image elements don't have explicit width and height)
- A total of 53 bugs were opened (49 bugs and 4 improvements) of which 3 had major severity, 4 minor severity, and the remaining ones had normal severity.

Test cases results

Test overview	Status					
	Pass %	No of tests	Passed	Failed	Blocked	Not tested
Functional testing-Register& login	90,32%	31	28	3	0	0
Functional Testing-Wishlist	100,00%	9	9	0	0	0
Functional testing-Shopping cart	100,00%	11	11	0	0	0
Functional testing-Delivery address	100,00%	10	10	0	0	0
Functional testing-Search bar	100,00%	8	8	0	0	0
Negative testing	92,86%	28	26	2	0	0
Nonfunctional features	86,11%	36	31	5	0	0
Compatibility testing mac&Chrome	88,00%	117	97	20	0	0
Compatibility testing Windows&Edge	87,00%	117	96	21	0	0
Smoke testing	94,44%	18	17	1	0	0
Exploratory testing	82,86%	35	29	6	0	0
Total	**92,87%**	**420**	**362**	**58**	**0**	**0**

Conclusion/ test report

The product was already released, so the tests needed to validate the implementation. Considering that all the test cases were complete and 92,87% passed, the product is validated.

The bugs regarding accessibility raised concerns and need fixing, so the navigation experience for screen reader users can improve. Also, the defects from performance functionality on browsers Chrome and Edge need fixing to help the page load faster.

Lesson learned

- For compatibility testing, Parallels Desktop VM had been installed on the access device, as no device with Windows OS was available.
- Consistently review and update test cases, bugs submitted, and data.
- Understand priority: tests high-impact functions first.