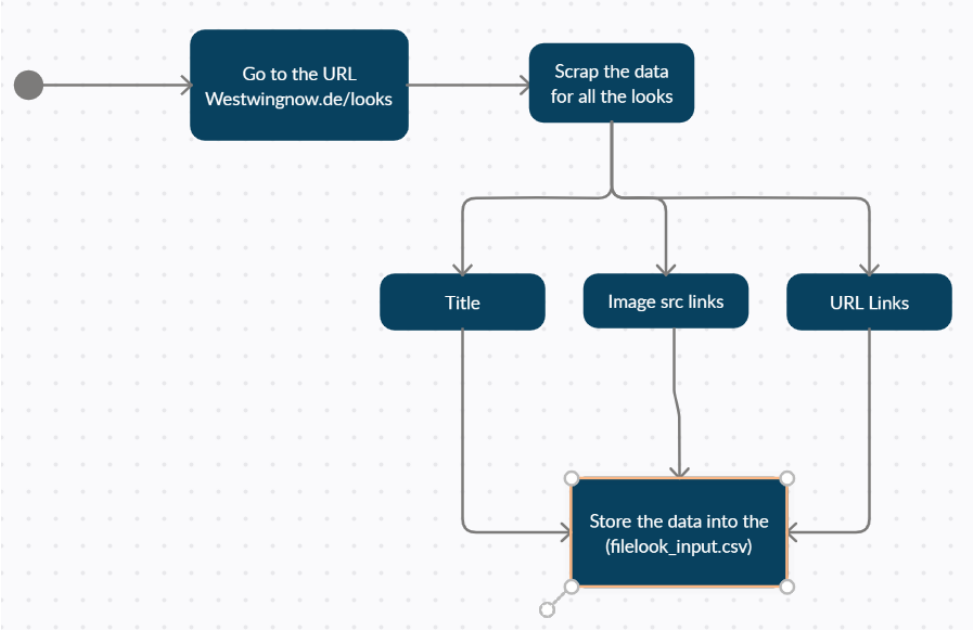
Look Listing Page

Has a list of All existing looks, with an image, title and link

Step 1) Creation of the CSV files (Test Data Generation)

First, it has to navigate to the URL “itstwingnow.de/looks” with the help of selenium (UI automation) it will scrap the data like (Looks title, looks image src links and looks URL links) and store it into the (looks_input.csv).

Note: *This step is only executed once to fill the reference data into files unless any new official change has occurred in the looks to refill the data again. But before that, the looks page must be tested and verified manually to avoid any kind of mistake while making the reference CSV file.*



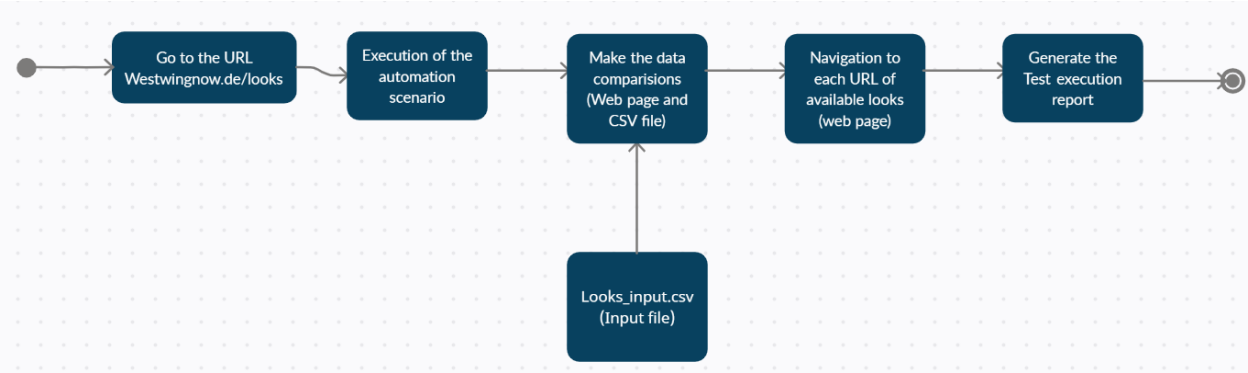
Step 2) Flow of the automation test scenario

In this step, we create the automation scenario to execute for testing of the looks page. First, it has to navigate to the URL “westwingnow.de/looks”. With the help of “foreeach” loop it will compare the data present in the Looks_input.csv file is same as displayed on the web page. It can verify the Title, Image src links and URL links for each of the available look and also check the sequence of the available looks on the web page. After that, it will navigate to each URL of the available look to check either it has navigating to correct page or not AND the navigated page has the same title as shown on the main looks page.

Proper report has been generated for the automation test execution that represents that which component of the of the tested scenario has an issue and related to which look item. By using this automation scenarios, it can check the look page in detail and also quickly as compare to the manual testing.

Automation Scenario Limitation:

The Automation has the limitations to check what is actually shown in the image. So, it cannot verify, either the Image represents the correct theme or not.

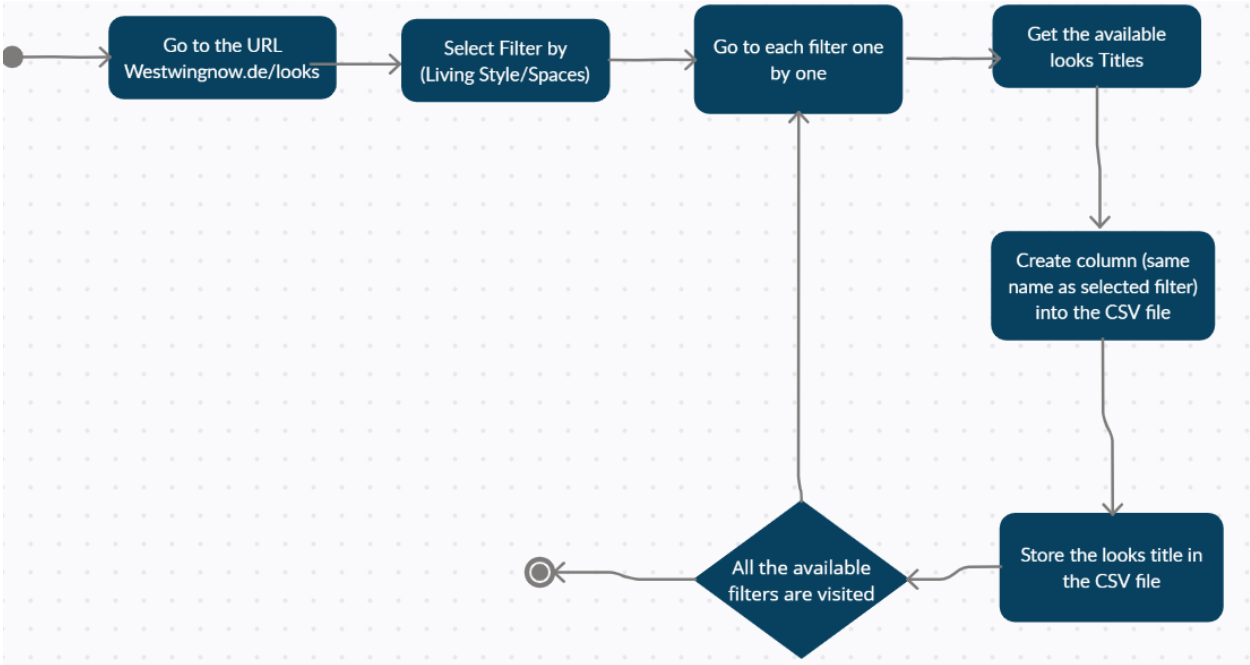


Has an Option to filter by room and style:

Step 1) Creation of the CSV files (Test Data Generation)

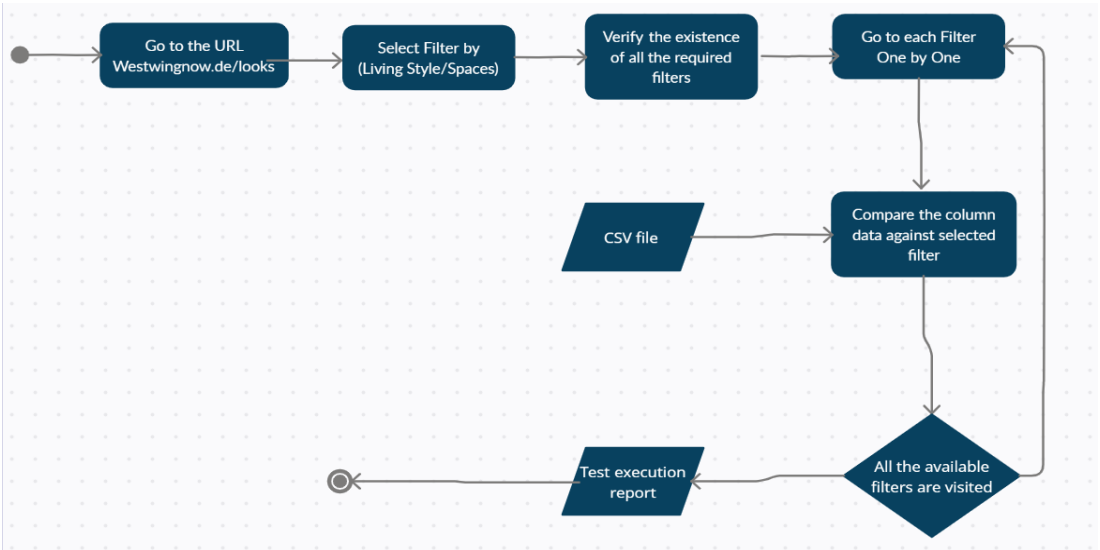
In this step, we have to create 2 CSV files, one for living style and second for the Spaces. When, the automation script is executed it will first go to each filter of the available category and select it (e.g., Glamor, etc) and create the column into relevant CSV file (same name of the column as the name of the filter) and store the looks title into the CSV file column. Repeat the step until all the available filters are covered.

Note: *This step is only executed once to fill the reference data into files unless any new official change has occurred in the looks to refill the data again. But before that, the looks page must be tested and verified manually to avoid any kind of mistake while making the reference CSV file.*



Step 2) Flow of the automation test scenario

It will select go to the URL westwingnow.de/looks, then select the option filter by. First it will check that all the required filters are available, then go each filter one by one. With the help of CSV reference file, the comparisons can be made to check that either the selected filter contains the correct look or not on the tested web page. The steps are repeated until all the filters are checked. In this way all the filters of are correctly tested using the automation scenario.



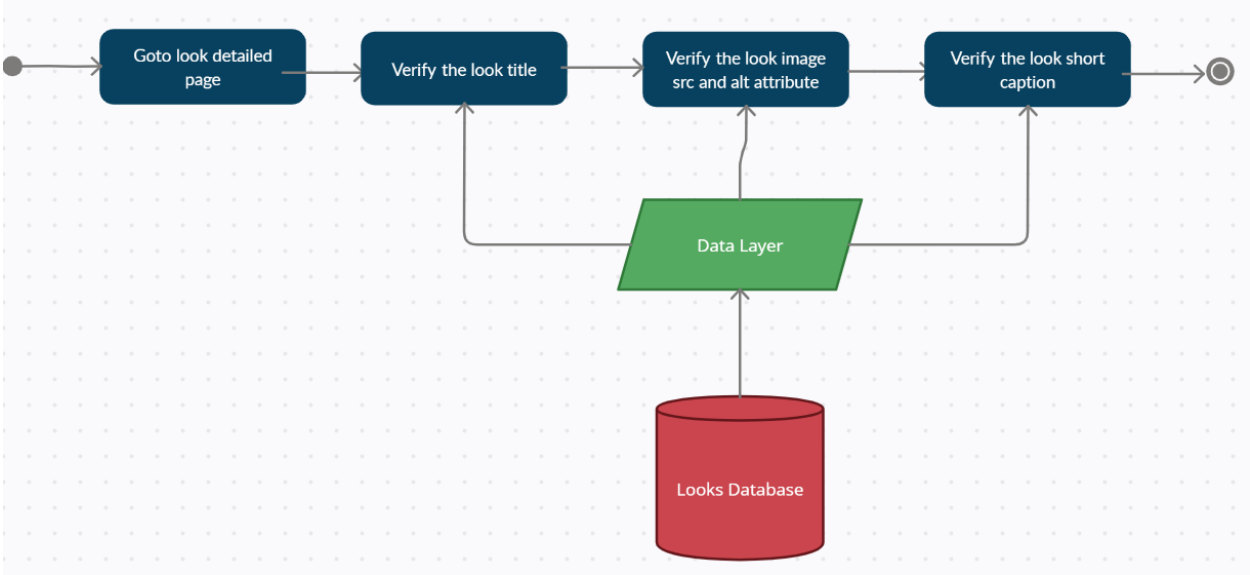
Look Details Page:

Has an Image, Look name and a short caption (Flow of Automation scenario)

We can verify the correct title, Image (src and alt like attributes) and caption on the tested web page by fetching and comparing the required data from the database.

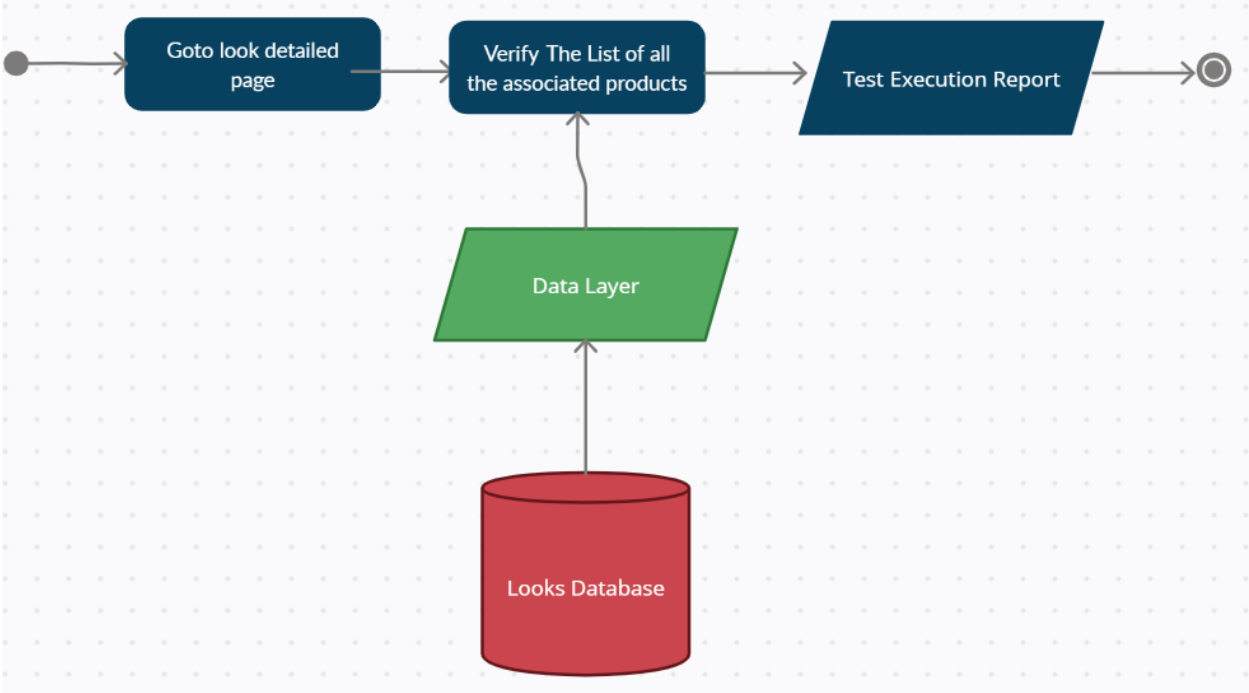
Limitation on Automation Scenarios: If the required data is not present in the DB then it will not be covered in the automation test scenarios (for example, if the short caption is not available in the DB then, in that case, the existence of UI element existence may be checked only without data comparison. But if required data is available and accessible in DB then we can cover data comparison in the automation scenarios).

In the Case of Images, we cannot automate the image verification instead we compare the src link and alt options of the Images with the help of DB.

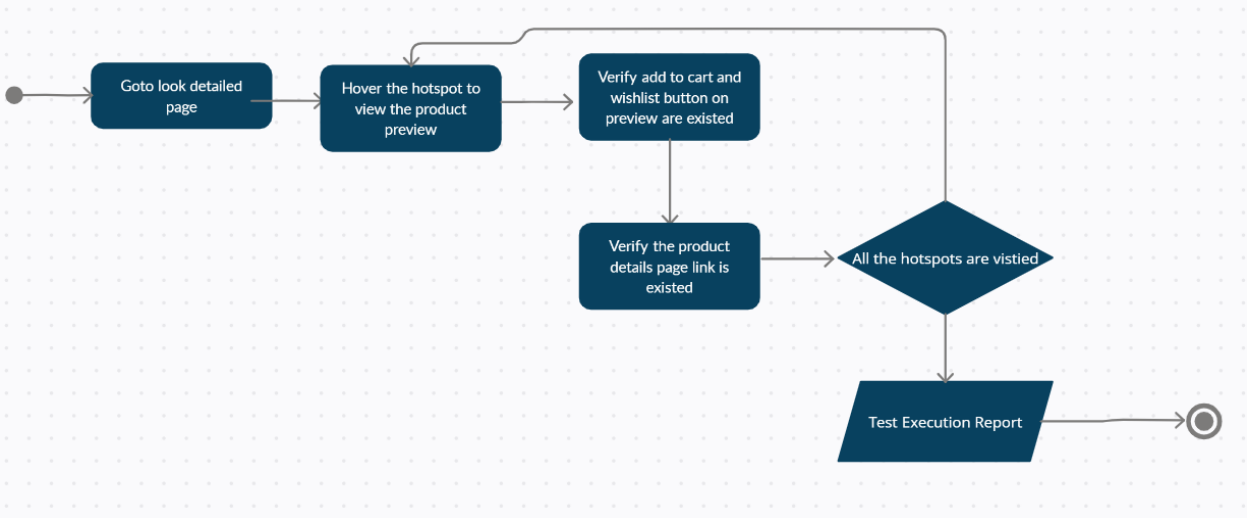


Has a list of product associated with the look: (Flow of Automation scenario)

We can verify the All the associated products by fetching and comparing the data from looks data base with the available product list on the webpage.



Has the Hotspots on the look image: (Flow of Automation scenario)



Has the Navigation to the next and previous look: (Flow of Automation scenario)