Exercise 04 – Identify by Anchor and Create ControlGroup

Objective

By the end of this exercise, you will be able to use the "Identify by Anchor" method to uniquely identify a Module control and create a ControlGroup within a Module.

Why is this important?

Some controls need additional steps to uniquely identify them. Without uniquely identifiable elements, we cannot steer Tosca. ControlGroups make TestCases easier to create and maintain.

Key elements

Identify by Anchor

ControlGroup

Instructions

1.	In the SUT, navigate to the WebShop **Homepage**.	☐
2.	In Tosca, navigate to the **Modules** Section.	☐
3.	Navigate to Module Folder: **WebShop >> In Work >> Products >> Exercise 04 - Product Choice Categories**.	☐
4.	Run XScan. From the Categories list on the left side of the page, select: · **Books** link · **Computers** link · **Electronics** link · **Apparel & Shoes** link · **Digital Downloads** link · **Jewelery** link · **Gift Cards** link Make all controls uniquely identifiable using **Identify by Anchor**. Use the **UL** container that contains the links, and identify it using the properties **Tag** & **OuterHtml**. *Remember: use the filter to see more details of the tree on the screen. *Remember: the anchor control must be uniquely identified for the objects underneath to be uniquely identified.	☐
5.	Within XScan, click on the **Module root** and change the property called **Title** using the wildcard to **Demo Web Shop*** to ensure that the Module will function on all pages.	☐
6.	Rename the Module itself to **WebShop | Product Choice Categories**.	☐
7.	Save and close XScan.	☐
8.	Within this Module, create a **ControlGroup** of the 7 controls just scanned. Name this Control group **Product Categories**.	☐
9.	Save your work.	☐

Hints

» Use **CTRL + Click** to select multiple ModuleAttributes.

» The * symbol acts as a wildcard. In this case, Tosca will perform the action on any window beginning with "Demo Web Shop" (e.g. Demo Web Shop. Books)

Cheat Sheet

Term	Description	Example
Model based test automation	Model based test automation is a software testing technique. During testing, models are created. They contain the technical information and the expected behavior of the System Under Test. These models provide Lego-like "building blocks" that can be combined and reused to create your tests. If your application changes (e.g., a field is added or removed), you just update the model, and the change is automatically propagated to all impacted tests.	
Tricentis Standard Modules	Tricentis Standard Modules are a series of Modules useful for performing common execution tasks within or outside your SUT. Tricentis Tosca provides a subset with default objects that help you to automate your tests and streamline your workflow. Standard Modules are pre-created objects that cannot be customized. As some functions are almost standardized or very much alike, it would be redundant to have to scan them every time you need to use them.	For example, the folder Standard modules contains the following sub-folders: TBox Automation Tools: Modules to perform automation operations.
XScan	Tricentis Tosca XScan can scan various applications. Each time Tricentis Tosca scans visual or non-visual test objects, it creates Modules in Tosca Commander that contain all technical information you needed to steer controls.	
control	A control is an element in your application that the test can interact with. The test ultimately steers these controls, i.e., interacts with them.	For example: a button, a text field, or a link. A test can verify text on a webpage, click buttons, or write something into a text box.
steer	Automated tests steer the controls in your application. That means that they interact with the control as a user would.	For example, you can design the test to click a button, choose an element from a drop-down menu, or enter text in a text field.
Identification Method	In Tosca XScan, you can identify controls in your test application by using multiple identification methods. The identification of controls by their properties is the default identification method in Tosca XScan. However, if your controls cannot be identified uniquely by via their controls, there are additional ways to identify controls.	
Identification by Property	Identifying controls by their properties is the default identification method in Tosca XScan. If a selected control is not uniquely identifiable, you can select additional properties to identify the control or use a different identification method.	

Term	Description	Example
Identify by Anchor	In Tricentis Tosca, you can use technical properties of controls to identify other controls. During automation, Tosca will look first for the anchor control, and then search for the target control in relation to the anchor control. It does this using two possible methods: shortest path or coordinate. With Shortest Path: the system searches each tree level for the control to be identified starting from the anchor control. The search starts directly beneath the control and goes from bottom to top up to the root element. When using the coordinate method, the exact position of the control to be identified is searched via coordinates. This option is highly dependent on the screen resolution. If the initial resolution changed, the control might not be found during test execution. The anchor must be unique.	For example, our web shop logo is a unique item. We could therefore use this element as the reference point (the anchor) to identify the link for “Books”.