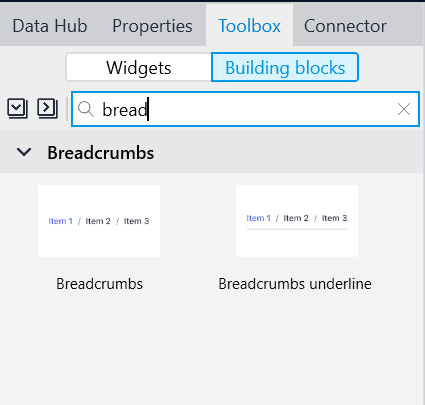
**Breadcrumbs**

This document will teach how you can implement Breadcrumbs for your Mendix apps. In Mendix we have inbuilt building blocks for breadcrumbs. However, those are restricted and need more enhancements based on your project requirements.

Graphical user interface, text, application, Word, email

Description automatically generated

# Overview:

# In our POC we have created a project for Products.

# Products have 3 categories-shoes, Clock, Vase (ENUM)

# Domain model for our Project is below:

# We have created Product Entity for storing distinct types of products

# Breadcrumb Entity is the main entity that we have created for our breadcrumb implementation.

# There are 3 attributes-

# Caption: For page title in the breadcrumbs. We are adding static names in microflow for caption. In below microflow you can see we have created Breadcrumb object and added caption statically.

# 

# 

# Page\_URL: For setting on click functionality and enabling page navigation through Breadcrumbs. We have added '/page name’ in the URL section under page property. And while creating object for breadcrumb we have set the page URL dynamically. In the below images you can see how we have done it.

# 

# 

# Similarly, we have done for all the pages that we have in our project.

# Order: For maintaining Indexes of the page title in Breadcrumbs list.

# Breadcrumb Entity is associated with Account, so for every User there will be a unique ‘breadcrumb’ displayed.

# We have added List View on the custom layout that is used for each page. Data source of that List View is Microflow in which Breadcrumbs List is returned for current user.

# 

# Graphical user interface, text, application, email Description automatically generatedGraphical user interface, text, application, email Description automatically generated

# Inside the list view, a button has been added and On Click action for that button is a Nanoflow:

# Graphical user interface, text, application, email Description automatically generated

# Below is the nanoflow which is called on the “{Caption}>” button. In this nanoflow we have passed a parameter and checks if breadcrumbs list is empty or not. If the list is empty, then we have called a microflow which returns the list of breadcrumbs for current user. Followed by open URL activity for displaying page through the URL. Instead of using Show page activity here we have used show URL because in show page we can only pass static page, but we wanted the page to displayed dynamically.

# Diagram Description automatically generated

# Below is the sub microflow that we are calling in the above nanoflow for updating the list of breadcrumbs by counting the order(index) and checking if it is greater than the current object order, if so then we delete those respective objects from the list with greater order. And then the user will be redirected to the page on which the user has clicked

# Diagram Description automatically generated