



Screen Interactions

Buttons and Links



Topics

- Interacting with the app
 - Link Widget
 - Button Widget
- Navigate Screen Lifecycle
- Submit Screen Lifecycle
 - Submit to the current Screen
- Download Widget

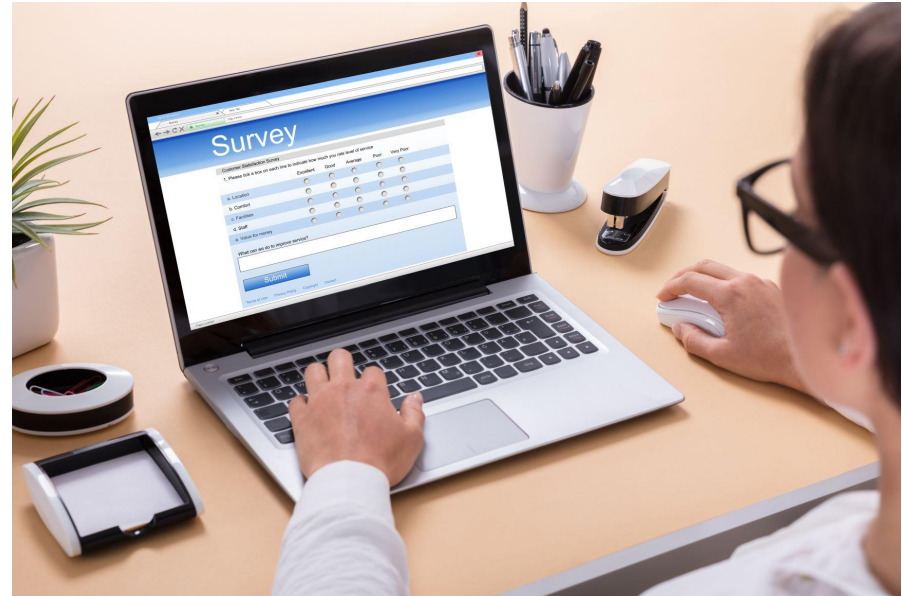
Interacting with the app

When using web applications, end-users don't just type URLs and wait for the result

It is very common to have other forms of interactions

- Following links
- Filling in forms
- Clicking buttons

Each time the user **navigates** to a new page, or **submits** data to the server, a new request / response cycle starts



Links and Buttons

Links and Buttons have similar behaviors

Links can enclose other widgets

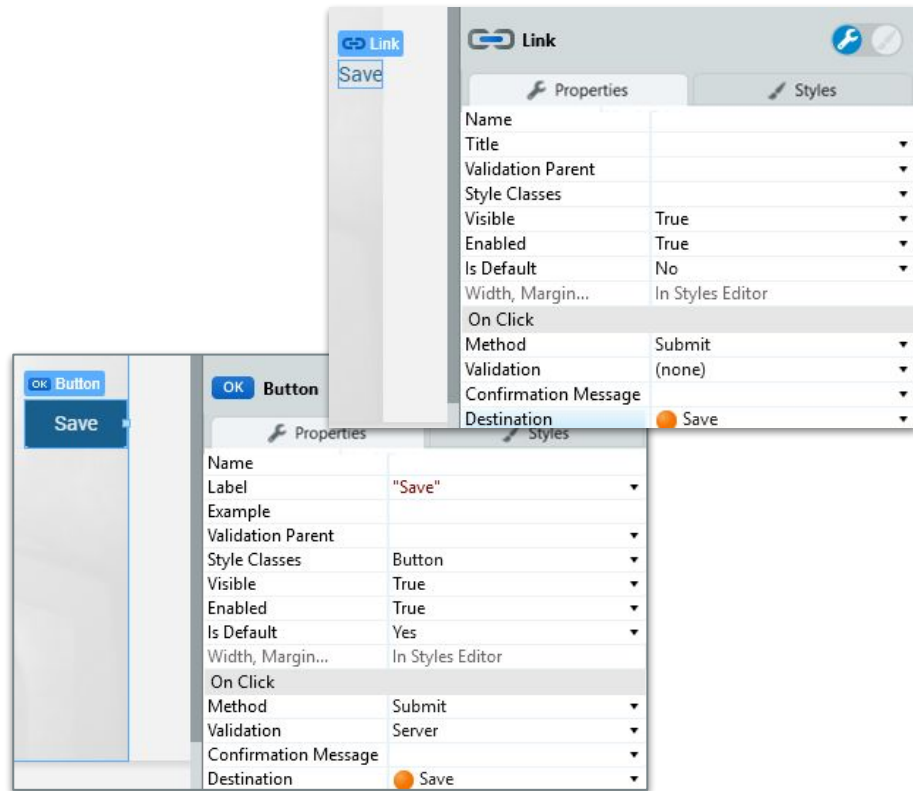
- Visually more flexible than Buttons

On Click: Either submits a request to call an Action or navigates to a Screen

Method property determines how data is submitted to the Server

Destination property can be set to a:

- Screen Action
- Destination



Navigate

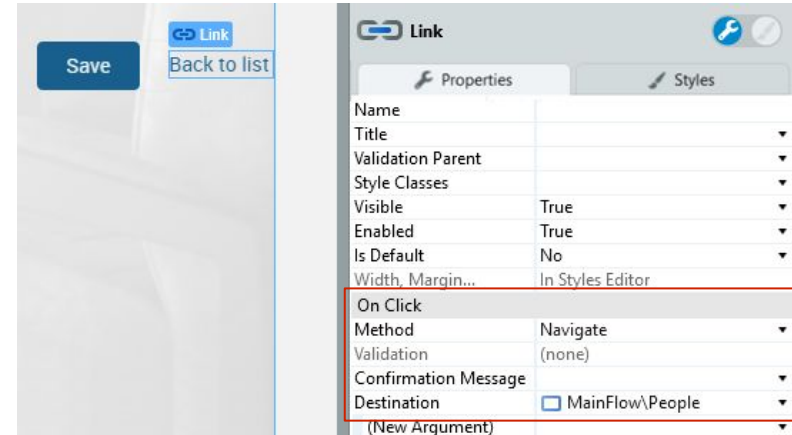
Uses HTTP GET Method

- Requests a specific resource
- Does not submit data to the server
 - Parameters are sent and displayed in the URL
 - Form Input values are not sent in the request

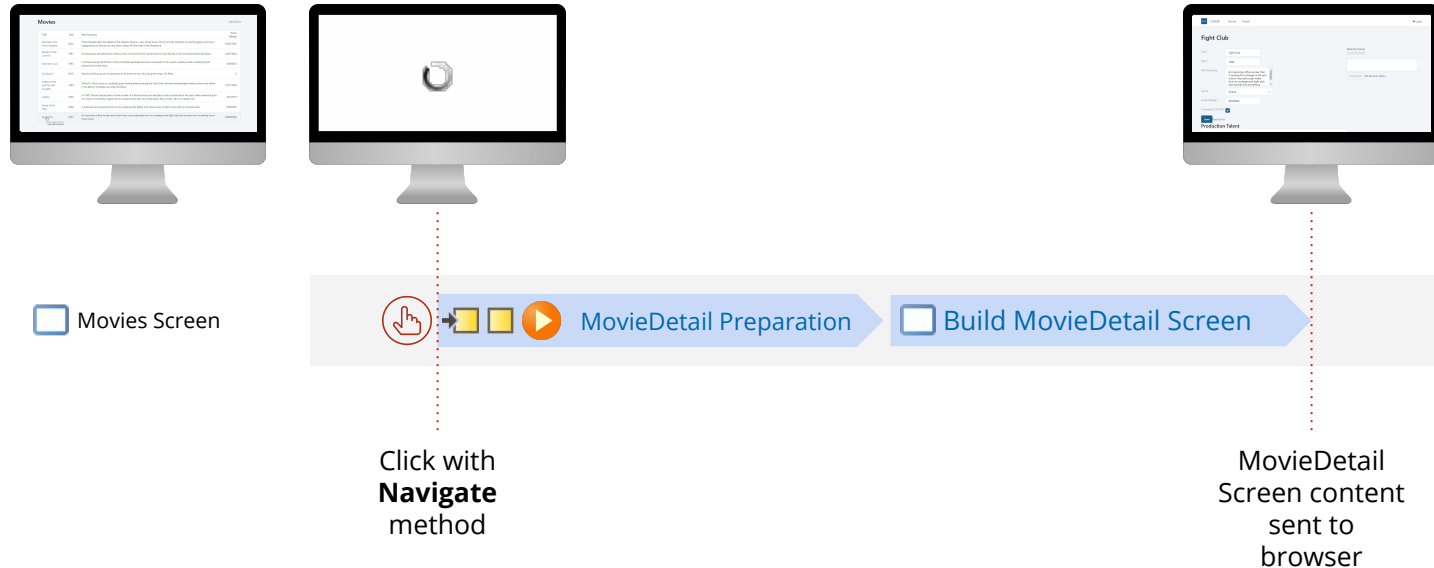
Browser displays the Destination URL

Link / Button Destination can be:

- Target Screen
- External URL



Navigate Screen Lifecycle



Submit

Uses HTTP POST Method

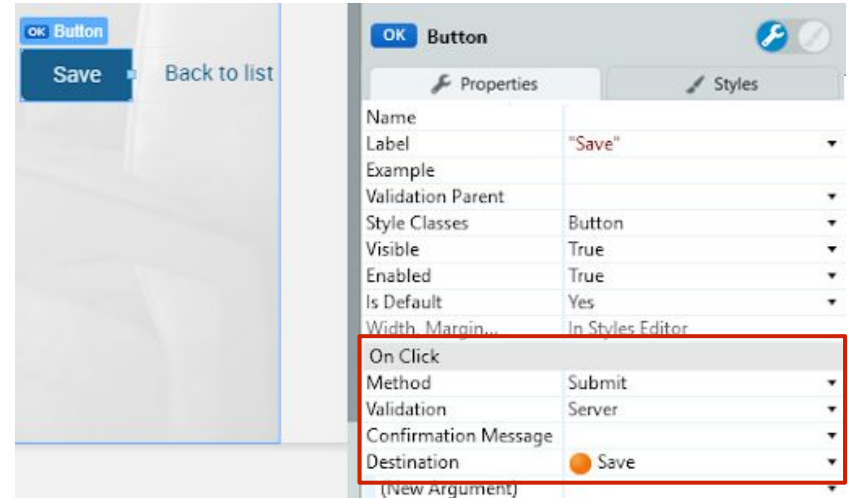
Submits data to the server in the request

- Upload a file
- Send Form Input fields data

Browser shows URL of the Screen that handles the submit request

Link / Button Destination can be:

- **Screen Action** (submits to current Screen)
- Target Screen
- External URL



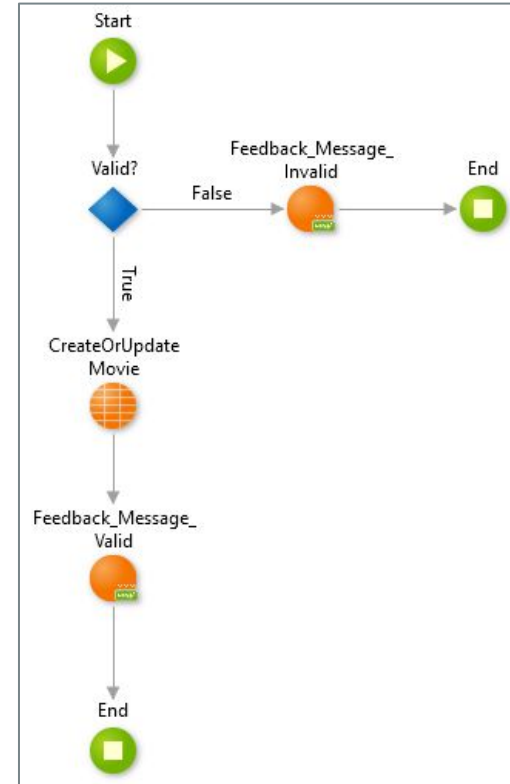
Submitting to a Screen Action

With the Submit Method, the Destination can be a Screen Action

- Runs when the Button / Link is clicked

A Screen Action has a flow of statements that are executed in order

- Has access to all elements in the Screen scope
 - Widgets
 - Preparation elements
 - Screen Variables
- The last node of the Action has impact on the Screen lifecycle

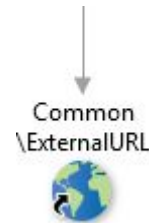
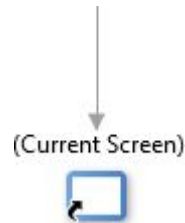
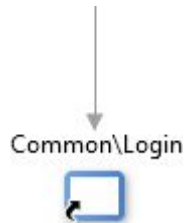
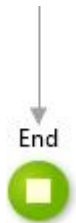


Finishing the Screen Action

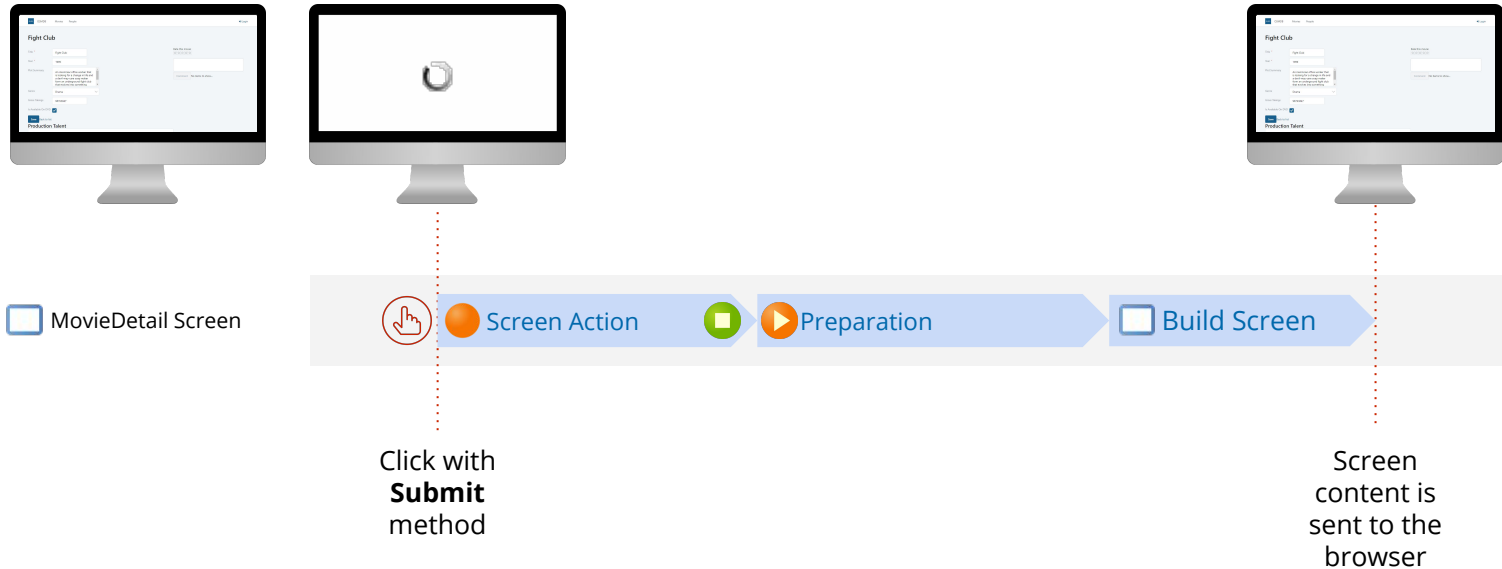
The last node of a Screen Action flow greatly influences the lifecycle

There are several possibilities

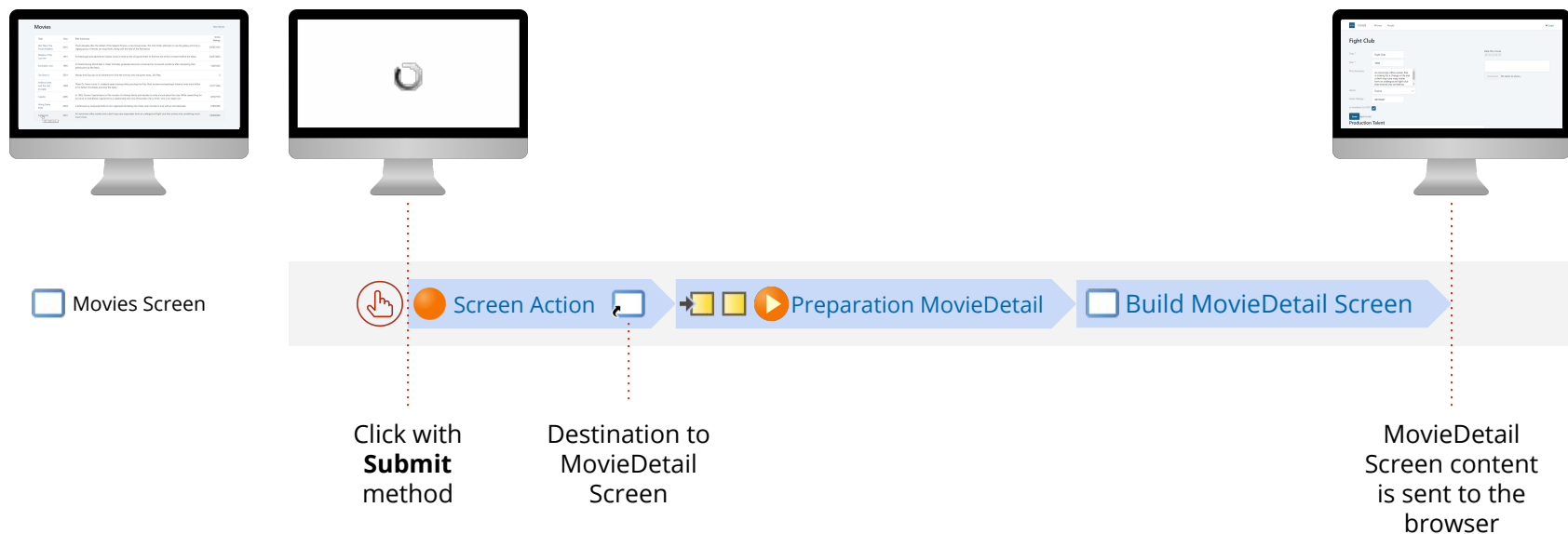
- Stay on the same Screen
 - End, Download (of a file)
- Navigate to a Screen
 - Destination, Current Screen, External URL
- Raise an Exception



Submit Screen Lifecycle: End



Submit Screen Lifecycle: Destination



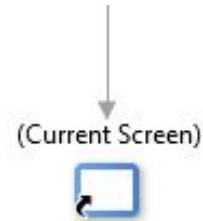
Submit Method: End vs. (Current Screen)

The application stays in the same Screen in both cases:

- Preparation of the Screen runs
- Entire Screen will be rebuilt

What's the difference?

- End Node
 - Values of Screen Variables are preserved
- Destination (Current Screen)
 - Values of Screen Variables are reset to their default values



Download

Allows the end-user to download a file

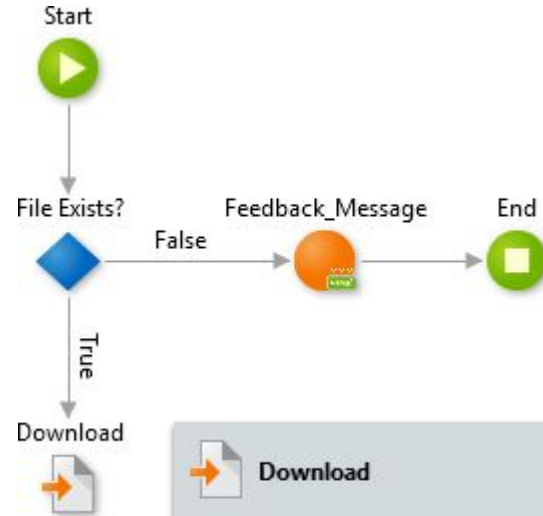
- Can be saved


Download node properties

- File Content: Expect binary data
- File Name: Text of the file name
- Save to Disk: Or just display the file

Ends the Action Flow

- There can be no other statements after it
- Preparation does not run again



|  Download | |
|--|---|
| File Content | GetOrders.List.Current.OrdersInvoice.Invoice ▼ |
| File Name | GetOrders.List.Current.OrdersInvoice.Filename ▼ |
| Mime-Type | "application/octet-stream" ▼ |
| Save to Disk | Yes ▼ |

Summary

- Interacting with the app
 - Link Widget
 - Button Widget
- Navigate Screen Lifecycle
- Submit Screen Lifecycle
 - Submit to the current Screen
- Download Widget



Screen Interaction
Thank You!