Create Social Proof Actions

Social proof is a demonstration that other people have purchased or found value in a product or service. This demonstration helps the product or service stand out from others and makes the customer more likely to purchase it. These demonstrations often reflect existing customer behavior. For example, labelling a product as a best seller is a social proof demonstration.

You can set up a handler for Omnichannel Social Proof actions using two methods. The `addEvents` method defines the events that can trigger the action. The `getActionsData` method is then used as the trigger and requests the decision based on the defined events. `getActionsData` then returns a JSON object containing recommendations data that you can then handle in code.

Prerequisites

You must first create an Omnichannel Social Proof action within Monetate for the methods to reference. Refer to Configure an Omnichannel Social Proof Action for instructions.

Make note of the WHO settings, as these correspond to events your code listens for. The example experience in this article uses the following WHO settings:

- IP address is 1.0.0.2
- Screen height is at least 500 pixels and screen width is at least 300 pixels

Also make note of the Social Proof type you are creating.

The example code in this article fulfills these conditions and will trigger the Social Proof action.

addEvent

This method records a local event in the defined context.

```
addEvent(context: <ContextEnum>, event: <MEvent?>)
```

Parameters:

- `context` is name of the event. (Required)
- `event` is the event data. (Required)

You can use this method multiple times to add all the necessary events for an experience you might want to trigger. The example code uses multiple method calls to fulfill the experience requirements.

Code Example

```
objPersonalization.addEvent(context: .ScreenSize, event: ScreenSize(height: 1000, width: 400))obj
```

Depending on the Social Proof type you want to use, you must use add certain events in addition to these. Refer

to the code examples below for these events.

getActionsData

This method sends the defined events to Monetate to trigger an experience. If the events fulfill the WHO settings and the other required conditions of an experience, that experience is triggered. A JSON object containing the experience response is then returned.

```
getActionsData(requestId: <String>, includeReporting: <Bool>, arrActionTypes: <[String]>
```

Parameters:

- requestID is the request ID for the API.
- actionType is the type of action you want to request. You can specify multiple actions in an array to handle.
- includeReporting indicates whether the response will have impression reporting data.

Code Example

```
objPersonalization.getActionsData( requestId: "123456 ", includeReporting: false,  arrActionTypes:
```

Viewed Product Code Example

This Social Proof type displays the most viewed products. It requires the ContextPageView event to be added.

```
// Add Context / Events//----------------------------------------objPersonalization.add
```

Carted Product Code Example

This Social Proof type displays the products most frequently added to a cart. You can use it for a single product or multiple products. Single products requires the ContextProductDetailView event to be added. Multiple products requires the ProductThumbnailView event to be added.

Single Product

```
// Add Context / Events//---------------------------------------objPersonalization.add
```

Multiple Products

```
// Add Context / Events//---------------------------------------objPersonalization.add
```

Purchased Product Code Example

This Social Proof type displays the products most frequently purchased. You can use it for a single product or

multiple products. Single products requires the ContextProductDetailView event to be added. Multiple products requires the ProductThumbnailView event to be added.

Single Product

```
// Add Context / Events//--------------------------------------------------objPersonalization.add
```

Multiple Products

```
// Add Context / Events//--------------------------------------------------objPersonalization.add
```

Recommended Product Code Example

This Social Proof type uses a recommendation strategy to determine and display a product the customer is most likely to purchase. It requires the ContextProductDetailView event to be added.

```
// Add Context / Events//--------------------------------------------------objPersonalization.add
```

Inventory Code Example

This Social Proof type displays products based on your current inventory, which may indicate a popular product due to limited inventory count. You can use it for a single product or multiple products. Single products requires the ContextProductDetailView event to be added. Multiple products requires the ProductThumbnailView event to be added.

Single Product

```
// Add Context / Events//--------------------------------------------------objPersonalization.add
```

Multiple Products

```
// Add Context / Events//--------------------------------------------------objPersonalization.add
```