Create an Omnichannel Recommendations Action

Omnichannel recommendations is an integrated recommendations experience that is consistent across all platforms that you connect to Monetate. Whether a customer is viewing your storefront through a mobile app or your site, Omnichannel recommendations ensure a consistent experience.

You can set up a handler for an Omnichannel recommendations action by 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 experience within Monetate for the methods to reference. Refer to Configure an Omnichannel Recommendations 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

The example code in this article fulfills these conditions and will trigger the Omnichannel experience.

addEvent

This method reports a local event.

```
addEvent(context: EventTypes, events: ContextData): void
```

Parameters:

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

```
personalizationInstance.addEvent(EventTypes.ContextScreenSize, { width: 23, height: 34 });

personalizationInstance.addEvent(EventTypes.ContextIpAddress, {
  ipAddress: "10.0.0.2",
});

personalizationInstance.addEvent(EventTypes.ContextPageView, {
  url: "http://www.monetate.com/index.html",
  pageType: "Home"
});
```

getActionsData

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

```
getActionsData(actionType: ActionType) : Promise<any>
```

Parameters:

- actionType is the type of action you want to request. You can specify multiple actions in an array to handle. (Required)

```
let recData;
personalizationInstance
  .getActionsData(ActionTypes.OmniChannelRecommendation)
  .then(res => {
    recsData = res[0].actions;
  })
  .catch(error => {
    console.warn('Error!', error);
  });
```

Full Code Example

Complete code example blocks are listed below.

```
// Add Context / Events
//-------------------------------------------------------------------------------
personalizationInstance.addEvent(EventTypes.ContextScreenSize, { width: 23, height: 34 });

personalizationInstance.addEvent(EventTypes.ContextIpAddress, {
  ipAddress: "10.0.0.2",
});

personalizationInstance.addEvent(EventTypes.ContextPageView, {
  url: "http://www.monetate.com/index.html",
  pageType: "Home"
});

// Get Actions
// -------------------------------------------------------
let recData;
personalizationInstance
  .getActionsData(ActionTypes.OmniChannelRecommendation)
  .then(res => {
    recsData = res[0].actions;
  })
  .catch(error => {
    console.warn('Error!', error);
  });
```