

JavaScript SDK Guide

Requirements

before starting the integration, ensure you have:

- Access to ADO Technologies' JavaScript SDK files

Integration Steps

1.-Include SDK and Assets: Incorporate the JavaScript SDK and related assets into your web project. This involves linking to the SDK's scripts files.

Ex:

```
<script type="application/javascript"
src="<URL_PROVIDED_BY_INTEGRATION_TEAM>/collector.js"></script>
```

```
<script type="application/javascript"
src="<URL_PROVIDED_BY_INTEGRATION_TEAM>/config.js"></script> // optional
```

2-. initializing the SDK (Optional if you have added the configuration script): if you have not imported the config.js file you must initialize the SDK manually. Then add the following code to do so

Parameter	Type	Description
cid	string	this is your identifier in Ado we will grant you a cid upon POC / going live
baseURL	string	This is our Ado URL - all of the data will be sent into this location (+configuration will be taken from there)

```
(async () => {
  const cid = "YOUR_CID_PROVIDED_BY_ADO"
  const baseURL = "YOUR_DOMAIN_PROVIDED_BY_ADO"
  const options = {cid: cid, baseURL: baseURL, csid: CSID};
  const document.collector = new Collector(options);
  await document.collector.initialize();
```

```
}());
```

3-. Set user ID (userID):

In order to set the userID so we will be able to track users and to detect fraudulent activity based on the user history static and dynamic data we will need to you to set the user id, if you don't want to send us the real user id - it's possible to send Ado the hash of it

- Please use the following setUserID function at the moment you will have the userID
- The user id should be consistent at any time that the same user is log in

```
const userID = bank_user_id || bank_user_id_hashed  
document.collector.setUserID(userID)
```

4-. Set Customer session ID (csid):

If the session ID changes or a new session is created, you'll need to call this function with the new session ID (string)

```
const CSID = bank_session_id  
document.collector.setCsid(CSID);
```

5-. Send Context (sendContext):

Contexts helps us to understand what is happening within the session and what action the user made, we will need you to send a context (String) before a button press / movement to another section of the application

For example if the user enters into transfer money page - we will want you to send us a String "**TRANSFER_MONEY**" with the sendContext function

Parameter	Type	Description
context	string	The activity which the user went through

```
const CONTEXT = "TRANSFER_MONEY"  
document.collector.sendContext(CONTEXT);
```

```
console.log hola mundo
```

Full example implementation

Below is an example HTML structure that demonstrates how to configure the SDK in your web application. This example includes links to the SDK and assets, configuration entries and the emit context button.

```
// index.html
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width = device-width, initial-scale = 1.0, user-scalable = 0, minimal-ui ">
  <title>Demo ADO Collector</title>
  <script type="text/javascript" src="./ComponentsManager.js"></script>
  <script type="application/javascript" src="https://js.test-ado.statsd.io/test-ado/collector.js"></script>
  <script type="application/javascript" src="https://js.test-ado.statsd.io/test-ado/config.js"></script>
</head>

<body>
  <h1>Collector js</h1>
  <h3>Init set user data</h3>
  <div style="display: flex; gap: 10px">
    <div>
      <label for="userID">User id</label>
      <input type="text" id="userID" value="" />
    </div>
    <br />
    <div>
      <label for="CSID">CSID</label>
      <input type="text" id="CSID" value="" />
    </div>
    <button onclick="onSetConfigUser()" id="btnStartCapture">Set</button>
  </div>
  <br />
  <h3>Send context</h3>
  <div>
    <label for="msg-context">Context</label>
    <input type="text" id="msg-context" value="" />
    <button onclick="onEmitContext()" id="emitContext">Emit context</button>
  </div>
</body>
</html>
```

```
</body>
<script>
  function onSetConfigUser () {
    setConfigUser(
      document.getElementById("userID").value,
      document.getElementById("CSID").value
    );
  }
  function onEmitContext() {
    emitContext(document.getElementById("msg-context").value);
  }

</script>
</html>
```

```
// ComponentsManager.js
let collector

async function setConfigUser (userID, CSID) {
  document.collector.setUserID(userID)
  document.collector.setCsid(CSID);
}

async function emitContext(msg) {
  document.collector.sendContext(msg);
}
```

Revision #14

Created 6 May 2024 15:28:33 by Admin

Updated 25 November 2024 14:32:22 by Admin