

Notification SDK - public

Demopage

We have provided really simple demo page, that describes current possibilities of SDK. What the page does is that it initializes SDK, after SDK is initialized demopage prints out the number of currently unread notifications and reacts on new notifications arriving to notification channel by displaying basic information about those. You can inspect source code powering this page at <https://fabric-sandbox.snapshot.travel/using-notifications.js>.

Usage

The SDK takes advantage of javascript adapter that's provided by Keycloak. Consumers of SDK need to add javascript reference to this adapter. Adapter itself is located at `(keycloak-instance)/auth/js/keycloak.js`. Hence for example for staging environment, appropriate import looks like this:

```
<script
src="https://euf-keycloak-sandbox.snapshot.technology/auth/js/keycloak.js"></script>
```

Notification SDK is distributed together with Fabric as a separate javascript bundle. It's available at URL `fabric(-environment).snapshot-travel/dist/snapshot-notifications.js`. It's necessary to reference this file as well. Again for staging environment, appropriate import looks like this:

```
<script
src="https://fabric-sandbox.snapshot.travel/dist/snapshot-notifications.js"></script>
```

Available methods

All methods are attached to newly exposed global object called `SnapshotNotifications`.

Hence calling `init` method means calling

```
window.SnapshotNotifications.init()
```

- `init()`
 - returns Promise
 - this method needs to be called first - it handles SDK internal configuration. SDK cannot work properly without this method being fully finished
- `getUnreadNotificationsCount()`
 - returns Promise
 - this method returns promise that later resolves to the number of currently unread notifications
- `onMessage(message)`
 - callback for newly coming notifications
 - it is triggered when new notification arrives and consumer should be informed about that
 - to use this callback, simply attach to it using

```
window.SnapshotNotifications.onMessage = (message) => { ... do  
whatever you want with notification ... }
```