Service worker notifications

Push notifications are a compelling way to engage users. Through the power of service workers, notifications can be delivered to a device even when your application is not in focus.

The Angular service worker enables the display of push notifications and the handling of notification click events.

When using the Angular service worker, push notification interactions are handled using the `SwPush` service. To learn more about the browser APIs involved see Push API and Using the Notifications API.

Prerequisites We recommend you have a basic understanding of the following:

- Getting Started with Service Workers.

Notification payload

Invoke push notifications by pushing a message with a valid payload. See `SwPush` for guidance.

In Chrome, you can test push notifications without a backend. Open Devtools -> Application -> Service Workers and use the `Push` input to send a JSON notification payload.

Notification click handling

The default behavior for the `notificationclick` event is to close the notification and notify `SwPush.notificationClicks`.

You can specify an additional operation to be executed on `notificationclick` by adding an `onActionClick` property to the `data` object, and providing a `default` entry. This is especially useful for when there are no open clients when a notification is clicked.

```
{
  "notification": {
    "title": "New Notification!",
    "data": {
      "onActionClick": {
        "default": {"operation": "openWindow", "url": "foo"}
      }
    }
  }
}
```

Operations

The Angular service worker supports the following operations:

- `openWindow`: Opens a new tab at the specified URL, which is resolved relative to the service worker scope.

- `focusLastFocusedOrOpen`: Focuses the last focused client. If there is no client open, then it opens a new tab at the specified URL, which is resolved relative to the service worker scope.

- `navigateLastFocusedOrOpen`: Focuses the last focused client and navigates it to the specified URL, which is resolved relative to the service worker scope. If there is no client open, then it opens a new tab at the specified URL.

If an `onActionClick` item does not define a `url`, then the service worker's registration scope is used.

Actions

Actions offer a way to customize how the user can interact with a notification.

Using the `actions` property, you can define a set of available actions. Each action is represented as an action button that the user can click to interact with the notification.

In addition, using the `onActionClick` property on the `data` object, you can tie each action to an operation to be performed when the corresponding action button is clicked:

```
{
  "notification": {
    "title": "New Notification!",
    "actions": [
      {"action": "foo", "title": "Open new tab"},
      {"action": "bar", "title": "Focus last"},
      {"action": "baz", "title": "Navigate last"},
      {"action": "qux", "title": "Just notify existing clients"}
    ],
    "data": {
      "onActionClick": {
        "default": {"operation": "openWindow"},
        "foo": {"operation": "openWindow", "url": "/absolute/path"},
        "bar": {"operation": "focusLastFocusedOrOpen", "url": "relative/path"},
        "baz": {"operation": "navigateLastFocusedOrOpen", "url": "https://other.domain.com/'
      }
    }
  }
}
```

If an action does not have a corresponding `onActionClick` entry, then the notification is closed and `SwPush.notificationClicks` is notified on existing clients.

More on Angular service workers

You might also be interested in the following:

- Service Worker in Production.