



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

# Journey JavaScript SDK

Types of tracked data

---

## Contents

- [1 Track viewed pages](#)
- [2 Track custom events](#)
- [3 Guidelines for custom event names](#)

---

Learn how to track different types of data using Journey SDK.

The Journey JavaScript SDK lets you customize how you collect tracking data for your website. The most basic form of tracking is page view tracking. For page view tracking, Genesys Predictive Engagement records each page a visitor visits. You can also use the Journey JavaScript SDK to record custom visitor activities such as button clicks. For more information see [Track custom events](#).

## Track viewed pages

The `pageview` method tracks the webpages that your visitors view. To send a pageview, call the `ac` function and pass `pageview` as the first argument.

```
ac('pageview');
```

### Important

By default, the Genesys Predictive Engagement tracking snippet contains the `ac` function.

## Track custom events

The `record` method allows you to track custom events, usually as a result of a person interacting with an element or control in your website. For example, the click of a button.

The `record` method takes two parameters:

- The name of the event to record as a string. **Note:** See [Guidelines for custom event names](#).
- An (optional) key-value hash of properties for the event.
- An (optional) callback function that invokes when the request is completed.
- An (optional) callback timeout, in milliseconds, to configure how long to wait when the event takes too long to complete. This option is useful to capture events (such as file downloads) before navigating to a different page/URL, ensuring that the visitor is always redirected.

```
ac('record', 'section_opened');
```

You can also provide extra metadata with your custom event:

```
ac('record', 'button_clicked', {
```

---

```
    companyName: 'Acme Inc,',
    employees: '100-500'
  });
```

The following code sample shows how to record a file download and navigate to the file URL `/files/pricing.pdf` when capturing the event, or after 400 milliseconds:

```
ac('record', 'file.downloaded', {
  name: 'Pricing',
  fileType: 'pdf'
}, function () {
  navigateTo('/files/pricing.pdf');
}, 400);
```

## Guidelines for custom event names

### Important

Event names:

- Must be less than 255 characters; we recommend keeping them short and descriptive
- Can contain any combination of alphanumeric characters, underscores, or hyphens only

The suggested format for custom event names is: *object-delimiter-action*, where *object* is the object that was interacted with, and *action* is the type of interaction that occurred.

Examples of custom event names include:

- "detected\_errors"
- "section\_failed"
- "section\_submitted"
- "product\_added"
- "product\_removed"
- "address\_selected"