



Getting Started

- What is Segment?
- [How Segment Works](#)
- Getting Started Guide
- A Basic Segment Installation
- Planning a Full Installation
- A Full Segment Installation
- Sending Data to Destinations
- Testing and Debugging
- What's Next
- Use Cases

Guides

Connections

Unify

Engage

Privacy

Protocols

Segment App

API

Partners

Glossary

Config API

Help

This documentation is maintained by the Segment community, please share your feedback, [contact our developer support team](#).

Getting Started

- From the Segment web app, click **Catalog**.
- Search for "Podsights" in the Catalog, select it, and choose which of your sources to connect the destination to.
- Visit your [Podsights dashboard](#) and navigate to Manage > Pixels. Copy your Pixel ID which will be your Segment "API Key".
- Drop the Pixel ID in the "API Key" field in your Segment Settings UI.

Once you start sending data to the Podsights' Destination it will take up to 20 minutes to appear in the Podsights pixel debugger.

Page

If you're not familiar with the Segment Specs, take a look to understand what the [Page method](#) does. An

example call would look like:

```
analytics.page()
```

Page calls will be sent to Podsights as a **view** event.

Podsights is an attribution platform, and as such, we need more context about the visitor than just a User ID. Analytics.js [automatically collects context fields](#). Podsights requires certain context fields and properties for page calls. Below is an example of a raw JSON payload that contains the minimum requirements.

```
{
  "type": "page",
  "context": {
    "ip": "1.2.3.4",
    "userAgent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_5) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/75.0.3770.100 Safari/537.36"
  },
  "properties": {
    "referrer": "",
    "url": "https://podsights.com/"
  },
  "timestamp": "2019-07-19T23:56:59.716Z",
  "userId": "3212"
}
```

For page events Podsights requires a **context** object that contains a **userAgent** and an **ip** field and a **properties** object that contains a **referrer** and a **url** field. As you can see in the page event's raw JSON payload above.

The **context** and **properties** object are required, along with the fields in them. If you're using Segment server-side you must send these attributes. Otherwise Podsights will return a **400 HTTP Error**.

Track

If you're not familiar with the Segment Specs, take a look to understand what the [Track method](#) does. An example call would look like:

```
analytics.track('Order Completed', {
  order_id: '50314b8e9bcf000000000000',
  total: 27.50,
  coupon: 'hasbros',
  currency: 'USD',
});
```

Track calls will be mapped to Podsights events. Podsights' support the following from the Segment Spec:

- [Signed Up](#) as lead
- [Product Viewed](#) as product
- [Product Added](#) as addto cart
- [Checkout Started](#) as checkout
- [Order Completed](#) as purchase

For track events Podsights requires a **context** object that contains a **userAgent** and an **ip** Podsights also requires a **page** object that contains a **referrer** and a **url** field. Analytics.js [automatically collects context fields](#). Podsights requires certain context fields for track calls. Below is an example of a raw JSON payload that contains the minimum requirements.

```
{
  "type": "track",
  "context": {
    "ip": "1.2.3.4",
    "userAgent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_5) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/75.0.3770.100 Safari/537.36",
    "page": {
      "url": "https://podsights.com/",
      "referrer": ""
    }
  },
  "event": "Test Event Name",
  "userId": "test-user-xip99",
  "timestamp": "2019-04-08T01:19:38.931Z",
  "properties": {}
}
```

The `context` and `page` object are required, along with the fields in them. If you're using Segment server-side you must send these attributes. Otherwise Podsights will return a 400 HTTP Error.

Server

Podsights does not support server-side events out of the box, but you can send server-side events if you follow the requirements of `page` and `track` events outlined in the sections for each call.

Engage

You can send computed traits and audiences generated using [Engage](#) to this destination as a **user property**. To learn more about Engage, schedule a [demo](#).

For user-property destinations, an [identify](#) call is sent to the destination for each user being added and removed. The property name is the snake_cased version of the audience name, with a true/false value to indicate membership. For example, when a user first completes an order in the last 30 days, Engage sends an Identify call with the property `order_completed_last_30days: true`. When the user no longer satisfies this condition (for example, it's been more than 30 days since their last order), Engage sets that value to `false`.

When you first create an audience, Engage sends an Identify call for every user in that audience. Later audience syncs only send updates for users whose membership has changed since the last sync.



Real-time to batch destination sync frequency

Real-time audience syncs to Podsights may take six or more hours for the initial sync to complete. Upon completion, a sync frequency of two to three hours is expected.

Settings

Segment lets you change these destination settings from the Segment app without having to touch any code.

SETTING	DESCRIPTION
API Key (required)	<code>string</code> . Visit Manage > Pixels and enter your Pixel ID as your API Key.

This page was last modified: 08 Jan 2024

Need support?

Questions? Problems? Need more info? Contact Segment Support for assistance!

[Visit our Support page](#)

Help improve these docs!

 [Edit this page](#)

[+ Request docs change](#)

Was this page helpful?

 [Yes](#)

 [No](#)

Get started with Segment

Segment is the easiest way to integrate your websites & mobile apps data to over 300 analytics and growth tools.

Your work e-mail

[Request Demo](#)

or

[Create free account](#)

© 2025 Segment.io, Inc.

[Privacy](#)

[Terms](#)

[Website Data Collection Preferences](#)

