



## 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

4. Follow the instructions in the GitHub repos: [iOS SDK](#) and [Android SDK](#).

5. Once the Segment library is integrated with your app, toggle Flurry on in your Segment UI.

*Note: Flurry does not always display data in real time. We've seen that it can take anywhere from a few hours to a few days for certain types of data to sync with Flurry.*

## Adding the dependency

To install the Segment-Flurry integration, simply add this line to your gradle file:

```
implementation 'com.segment.analytics.kotlin.destinations:flurry:<latest_version>'
```

Or the following for Kotlin DSL

```
implementation('com.segment.analytics.kotlin.destinations:flurry:<latest_version>')
```

## Using the Plugin in your App

Open the file where you setup and configure the Analytics-Kotlin library. Add this plugin to the list of imports.

```
import com.segment.analytics.kotlin.destinations.flurry.FlurryDestination
```

Just under your Analytics-Kotlin library setup, call `analytics.add(plugin = ...)` to add an instance of the plugin to the Analytics timeline.

```
analytics = Analytics("<YOUR WRITE KEY>", applicationContext) {  
    this.flushAt = 3  
    this.trackApplicationLifecycleEvents = true  
}  
analytics.add(plugin = FlurryDestination())
```

Your events will now begin to flow to Flurry in device mode.

## Screen

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

An example Screen call would look like:

```
analytics.screen("ScreenName", buildJsonObject {  
    put("productSlug", "example-product-123")  
});
```

*Note: When you toggle the Screen Tracks As Events option on in your Flurry Segment UI - we will treat screen calls as events when sending them to Flurry.*

## Identify

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

An example Identify call would look like:

```
analytics.identify("user-123", buildJsonObject {  
    put("username", "MisterWhiskers")  
    put("email", "hello@test.com")  
    put("plan", "premium")  
});
```

When you call `identify`, we'll set the user ID in Flurry, and set any special Flurry traits you provide, such as gender, OR age.

## Track

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

An example Track call would look like:

```
analytics.track("View Product", buildJsonObject {  
    put("productId", 123)  
    put("productName" "Striped trousers")  
});
```

This page was last modified: 20 Jun 2023

---

## 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](#)

