Quick Start

Start building now.

Quick Start

Currently we offer both transactions webhooks (listening to transactions for a set of accounts) and account change webhooks (listen to account changes for a set of accounts). Helius lets you interact with webhooks in three ways.

Helius UI (Recommended)

If you'd rather not bother with code and want additional methods for viewing logs and sending test webhook events, the Helius UI is what you need. The UI can be accessed at dev.helius.xyz .

Helius REST API

If you're not working with Typescript or Javascript, you'll need to interact with our webhooks through REST: API Reference

Helius SDK

The easiest (and most fun) way to interact with Helius webhooks is to use the official SDK: [GitHub - helius-labs/helius-sdk GitHub](GitHub) The SDK contains abstractions to enhance what webhooks have to offer, including the ability to create collection webhooks (i.e., a webhook that tracks all the NFTs within an NFT collection)!

Example Uses

* Bots * * When an NFT is listed on marketplace X, trigger an "nft buy" action. * * When a margin position is unhealthy, trigger a "liquidation" action. * Monitoring & Alerts * * When a program emits a certain log, trigger PagerDuty integration. * * When a token account balance changes by more than X%, use Dialect to communicate a warning action. * Event-driven Indexing * * When any transaction occurs for a given program, send it directly to your database or backend. * Notifications & Activity Tracking * * When a transfer occurs from wallet X to wallet Y — send a Slack notification or email. * Analytics & Logs * * When event X happens, send it to an ETL pipeline or persist it directly on Helius to view trends over time. * Workflow Automation * * When event X happens, trigger any set of actions. [Webhooks & Websockets -Previous What are Webhooks? Next- Webhooks & Websockets API Reference](Webhooks) Last modified2mo ago