twitch.json

misinformation

Page ID: 85

Source: https://dev.twitch.tv/docs/extensions/designing/

['authentic']

Designing Extensions | Twitch

Developers Asset 20 Products Showcase Support Blog Docs Tutorials Log in with Twitch Contents Overview Twitch API EventSub Chat & Chatbots Authentication Organizations Drops Embedding Twitch Extensions Get started Required Technical Background Designing Extensions Introduction General Guidelines Panel Extension Guidelines Video-Overlay Extensions Mobile Guidelines Pop Out Building Extensions Using the Twitch API in an Extension Front End Life Cycle Management Monetization A/B Testing Using Google Analytics Increase Feedback Load Testing Extensions Submission Best Practices Guidelines & Policies Reference Insights & Analytics Mobile Deep Links PubSub Video Broadcast Twitch CLI Changelog Product Lifecycle Designing Extensions Introduction

If you are just getting started with Extensions, use the recommendations in this guide to put your best foot forward for streamers.

If you already shipped an extension, use this guide to take your extension to the next level; e.g., with other extension types, such as panel or video overlay.

General Guidelines Extensions provide new ways to interact on Twitch, so we provide guidelines to help ensure your experiences are harmonious with the overall Twitch experience.

The guidelines in this section apply to all extension types.

Guidelines for specific extension types appear in Panel Extension Guidelines, Video Overlay Extension Guidelines, Mobile Guidelines, and Pop Out Guidelines.

Terminology

['trust']

For example, a Hearthstone stream has most of its action in the middle of the board, so occupying the top right of the video will ensure your extension does not interfere with the game’s content.

However, if your extension augments the information about the game and may require placement closer to the content on video, make sure it can be collapsed, to ensure you limit its interference with the content/game.

Fail Gracefully and Take Feedback

Well With potentially hundreds of thousands of users interacting with your extension every day, there is a chance your extension will have issues.

Even relatively simple Extensions (like those that display the gear a streamer uses to create a broadcast) can have hundreds of failure points.

Just like any app, anticipating general failures helps build user trust.

Apps should handle 404 or 500 errors by providing ways for viewers to provide feedback and leverage some other part of your extension.

Apps that fail to retrieve data also should allow users to provide feedback.

Streamers control the placement and use of extensions, so your extension should be prepared to accept feedback directly.

If you don’t provide support, Twitch may find it necessary to remove your extension from the directory.

Choose the Right Extension Type Throughout this guide, we provide guidelines for using different types of Extensions, but in general, consider: Content — How does your extension expand stream content?

Does it relate to a game, or does it integrate with other apps (like those that play music)?