How the Engineer Behind Coding Kitty Built a Full Video Publishing Engine on Buffer's API



Sameer Ali is the kind of developer who, when faced with a repetitive task, builds an app to fix it. He's also a creator, publishing coding content on Instagram, TikTok, and YouTube under the name coding.kitty — and his build-an-app approach was exactly how he removed almost all the admin from his social presence.

Previously, every finished video meant 15–20 minutes of busywork: downloading, re-uploading to each platform, rewriting captions, setting metadata, scheduling, and manually

updating his project board. His custom desktop app — the aptly named coding.kitty engine — completely changed that. It handles the entire production pipeline from ideation and scripting through to subtitling, scheduling, and analytics.

But when it came to actually getting the videos onto Instagram, TikTok, and YouTube, even Sameer decided not to build that part himself. He turned to Buffer's API instead.

Why a full-stack engineer chose *not* to build it himself

Sameer could have used Instagram, TikTok, and YouTube's native APIs. But that would have meant three separate OAuth flows, three different upload mechanisms, three sets of rate limits, and a custom scheduler service to keep it all running on time. He described it as building a whole product on top of the product he was already building.

Instead, Buffer handles all of it.

One GraphQL API, one auth flow, and Sameer can pass YouTube titles, privacy settings, categories, Instagram Reel vs. post type, first comments, and TikTok titles through a single mutation. He had the full integration working in a few days.

"I want a guarantee that my posts will be posted at the specified time," Sameer says. "Buffer handles the scheduling part reliably, and I can see everything in a calendar view."

He also liked that the GraphQL schema covered everything he needed in one place: create, delete, fetch posts, fetch channels — without him having to stitch together separate endpoints.

From 1 YouTube video to multiple posts in

2 minutes

As a long-time creator and marketer, I’m can’t get over how efficient Sameer’s app + Buffer API workflow is.

It kicks off when a video gets marked “ready to schedule” in Jira. The coding.kitty engine picks it up automatically, downloads the subtitled video from the Jira attachment, and automatically resizes it to fit Instagram Reels’ vertical format. The video uploads to Sameer’s cloud storage and becomes accessible via a public URL.

From there, Sameer picks the target platforms, generates a platform-specific caption (his built-in AI knows the character limits and conventions for each platform, so he’s not rewriting the exact same message three times), and scrubs through the video to select a thumbnail frame.

Then coding.kitty hits Buffer’s CreatePost GraphQL mutation that includes the video URL, caption, thumbnail, and all platform-specific metadata. Buffer fetches the video from R2, queues it for publishing, and the Jira ticket auto-transitions to the next column.

Two minutes, done.

A calendar and smart post scheduling

If that weren’t enough, Sameer built a couple more handy features into his app. For starters, his own calendar view.

coding.kitty also pulls scheduled posts back from Buffer to display its own calendar view. Sameer can spot gaps in his schedule, avoid posting conflicts, and reschedule content by dragging posts around — all without leaving his app.

And perhaps my favorite bonus feature: An AI agent that makes all the decisions around publishing.

Because coding.kitty has access to recent posts, and the full queue of videos ready to go, Sameer can hand off the publishing decision entirely to the agent. It checks what's been posted recently, picks the right content, platform, and timing, and schedules it through the Buffer API.

Pretty cool, right?

Try it yourself

Sameer built a desktop app. You might build a CLI tool, a Slack bot, or an n8n workflow. Whatever you build, you can focus on that — the publishing piece is already solved.

Buffer's API is now available. You can start building today.

https://buffer.com/resources/coding-kitty-case-study/