Built-in AI APIs for the Web

Thomas Steiner
tomac@google.com

Adding AI to your web app

Imagine you build an app to let users record and organize creative ideas: ideas.example.com 💡.

Features:

- Record ideas by voice (*e.g.*, musical ideas like a tune or a melody, so OS-native dictation isn't an option).
- Enter ideas by keyboard (*e.g.*, a catchy headline for a blog post).
- Sketch ideas with a drawing (*e.g.*, the dimensions of a piece of furniture).
- Grab ideas by photo (*e.g.*, to recall the design of a fabric).

Adding AI to your web app (cont'd)

Possible AI use:

- *"List all my ideas where I hum."*
- *"Based on this headline, write an opening paragraph."*
- *"How much wood do I need given this sketch?"*
- *"Where can I buy a t-shirt with this design?"*
- *"Rewrite this rough idea more formally."*
- *"Share this idea translated to Japanese with my friend 英治."*
- *"Summarize all my ideas from yesterday afternoon."*

Adding AI to your web app (cont'd)

Option 1: Use an AI provider in the cloud ⛅

+ Some of the most powerful models run (exclusively) in the cloud.
+ The app can remain small and doesn't consume much disk space.
+ No device hardware requirements, as all AI inference happens on the server.

- Can get expensive and make cost prediction unclear.
- Requires user data to be sent to the cloud.
- Doesn't work offline.

Adding AI to your web app (cont'd)

Option 2: Download AI models locally and run on the client 💻

+ Privacy-preserving, as all processing happens locally.
+ Works offline once the initial model download has happened.
+ Is free to the developer as all inference happens on the client.

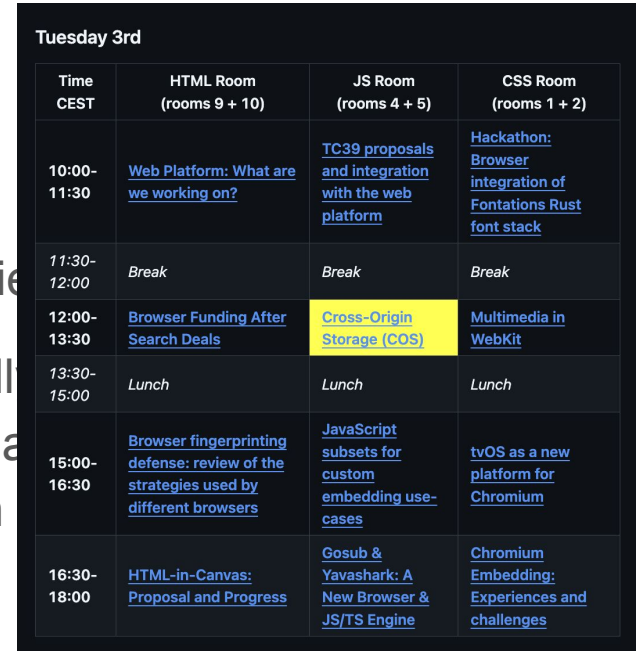
- Model downloads can be prohibitively large, and require lots of disk space.
- May not work on all hardware or be slow.
- Limits what models can be used, no way to protect a model.

Adding AI to your web app (cont'd)

Option 2: Download AI models locally and run on the clie

+ Privacy-preserving, as all processing happens locall
+ Works offline once the initial model download has ha
+ Is free to the developer as all inference happens on

shameless plug



Tuesday 3rd

Time CEST	HTML Room (rooms 9 + 10)	JS Room (rooms 4 + 5)	CSS Room (rooms 1 + 2)
10:00-11:30	Web Platform: What are we working on?	TC39 proposals and integration with the web platform	Hackathon: Browser integration of Fontations Rust font stack
11:30-12:00	*Break*	*Break*	*Break*
12:00-13:30	Browser Funding After Search Deals	Cross-Origin Storage (COS)	Multimedia in WebKit
13:30-15:00	*Lunch*	*Lunch*	*Lunch*
15:00-16:30	Browser fingerprinting defense: review of the strategies used by different browsers	JavaScript subsets for custom embedding use-cases	tvOS as a new platform for Chromium
16:30-18:00	HTML-in-Canvas: Proposal and Progress	Gosub & Yavashark: A New Browser & JS/TS Engine	Chromium Embedding: Experiences and challenges

- Model downloads can be prohibitively large, and require lots of disk space.
- May not work on all hardware or be slow.
- Limits what models can be used, no way to protect a model.

Adding AI to your web app (cont'd)

Proposed option 3: Make a shared AI model available to Web APIs (built-in AI) 🧠

+ Privacy-preserving, as all processing happens locally.
+ Works offline once the initial model download has happened.
+ Is free to the developer as all inference happens on the client.
± Model downloads can be prohibitively large, but it's a one-time operation.

- May not work on all hardware or be slow.
- Limits what models can be used, as you're bound to what the browser ships.

Built-in AI APIs

Web Machine Learning Community Group Charter

- This Charter: https://webmachinelearning.github.io/charter/
- Previous Charter: a4da99c
- Start Date: 2025-05-19
- Last Modified: commits/master

webmachinelearning.github.io/charter/#scope-of-work

Scope of Work

Task-specific APIs

A set of higher-level APIs that:

- Detect the language of the given input text;
- Translate input text to other languages;
- Produce textual summaries of input text;
- Produce new textual material given a writing task;
- Rephrase input text;
- Proofread input text.

Prompt API

A general-purpose API to prompt a language model directly using common prompting techniques to accomplish a variety of tasks, including:

- Classification, tagging, and keyword extraction of arbitrary text;
- Helping users compose text, such as blog posts, reviews, or biographies;
- Summarizing, e.g. of articles, user reviews, or chat logs;
- Generating titles or headlines from article contents;
- Answering questions based on the unstructured contents of a web page;
- Translation between languages.

Translator and Language Detector APIs

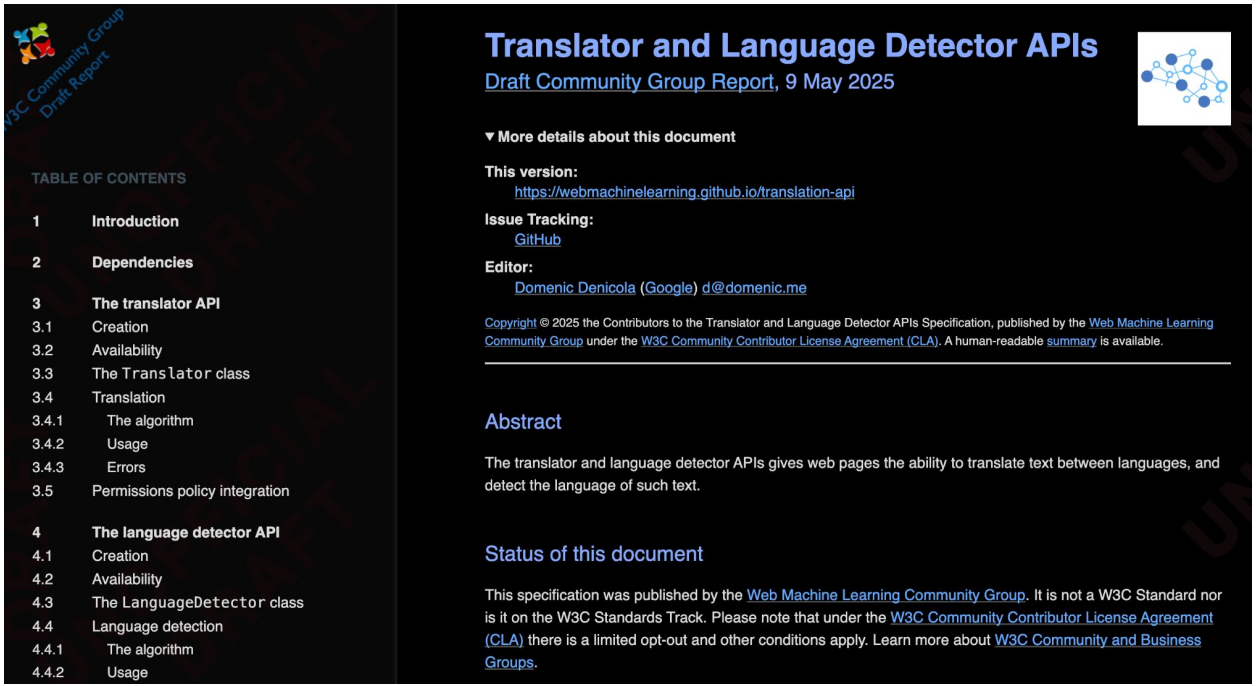


TABLE OF CONTENTS

Translator and Language Detector APIs

Draft Community Group Report, 9 May 2025

▼ More details about this document

This version:
https://webmachinelearning.github.io/translation-api

Issue Tracking:
GitHub

Editor:
Domenic Denicola (Google) d@domenic.me

Abstract

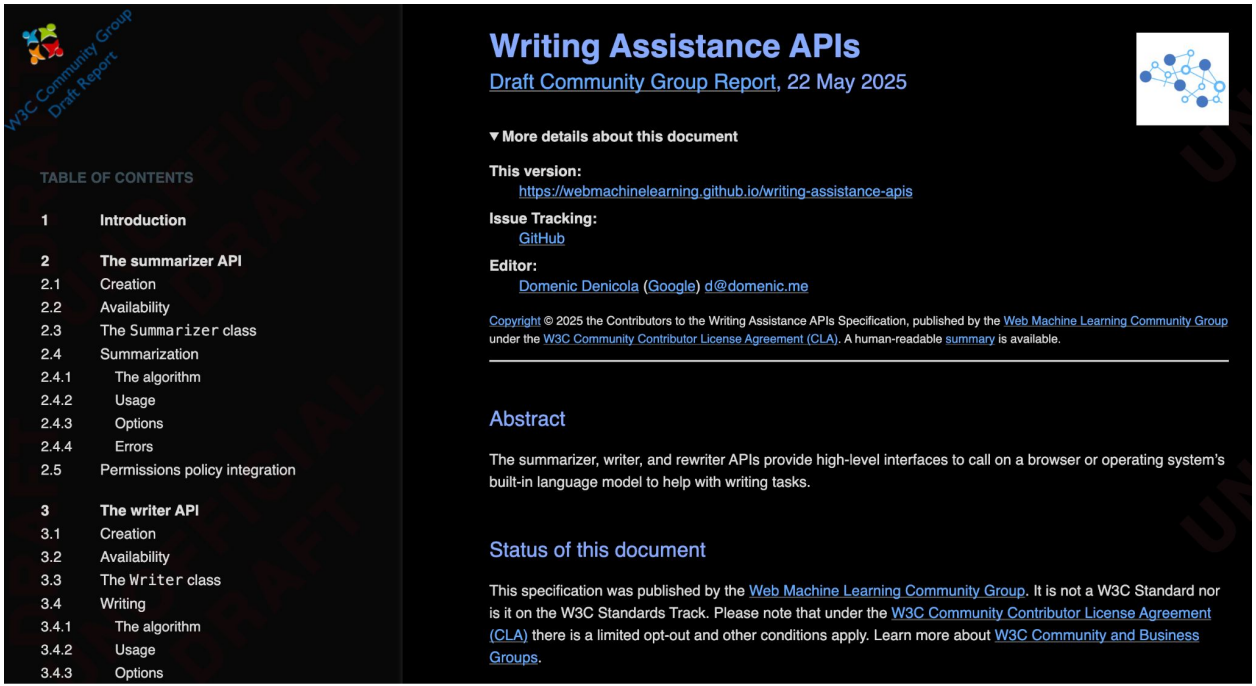
The translator and language detector APIs gives web pages the ability to translate text between languages, and detect the language of such text.

Status of this document

This specification was published by the Web Machine Learning Community Group. It is not a W3C Standard nor is it on the W3C Standards Track. Please note that under the W3C Community Contributor License Agreement (CLA) there is a limited opt-out and other conditions apply. Learn more about W3C Community and Business Groups.

webmachinelearning.github.io/translation-api/

Writing Assistance APIs



W3C Community Group Draft Report

TABLE OF CONTENTS

Writing Assistance APIs

Draft Community Group Report, 22 May 2025

▼ More details about this document

This version:
https://webmachinelearning.github.io/writing-assistance-apis

Issue Tracking:
GitHub

Editor:
Domenic Denicola (Google) d@domenic.me

Abstract

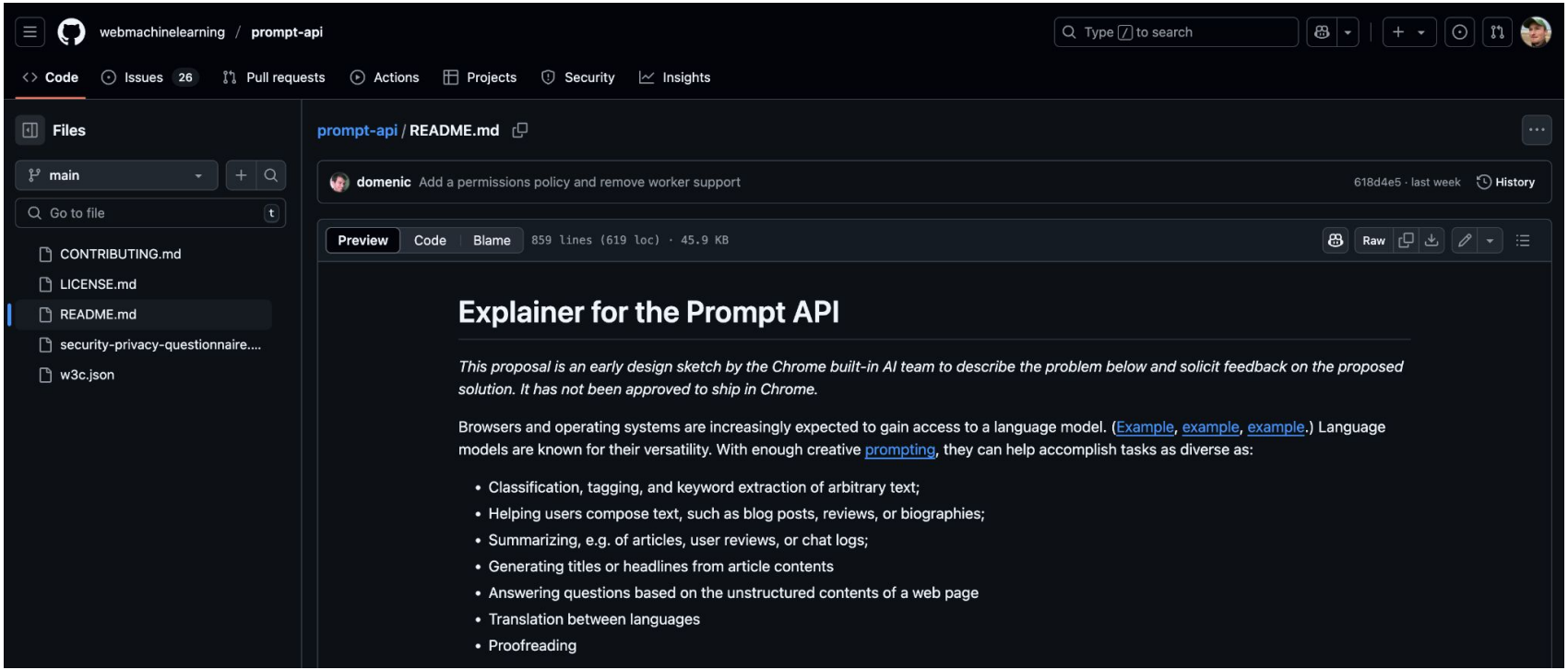
The summarizer, writer, and rewriter APIs provide high-level interfaces to call on a browser or operating system's built-in language model to help with writing tasks.

Status of this document

This specification was published by the Web Machine Learning Community Group. It is not a W3C Standard nor is it on the W3C Standards Track. Please note that under the W3C Community Contributor License Agreement (CLA) there is a limited opt-out and other conditions apply. Learn more about W3C Community and Business Groups.

webmachinelearning.github.io/writing-assistance-apis/

Prompt API



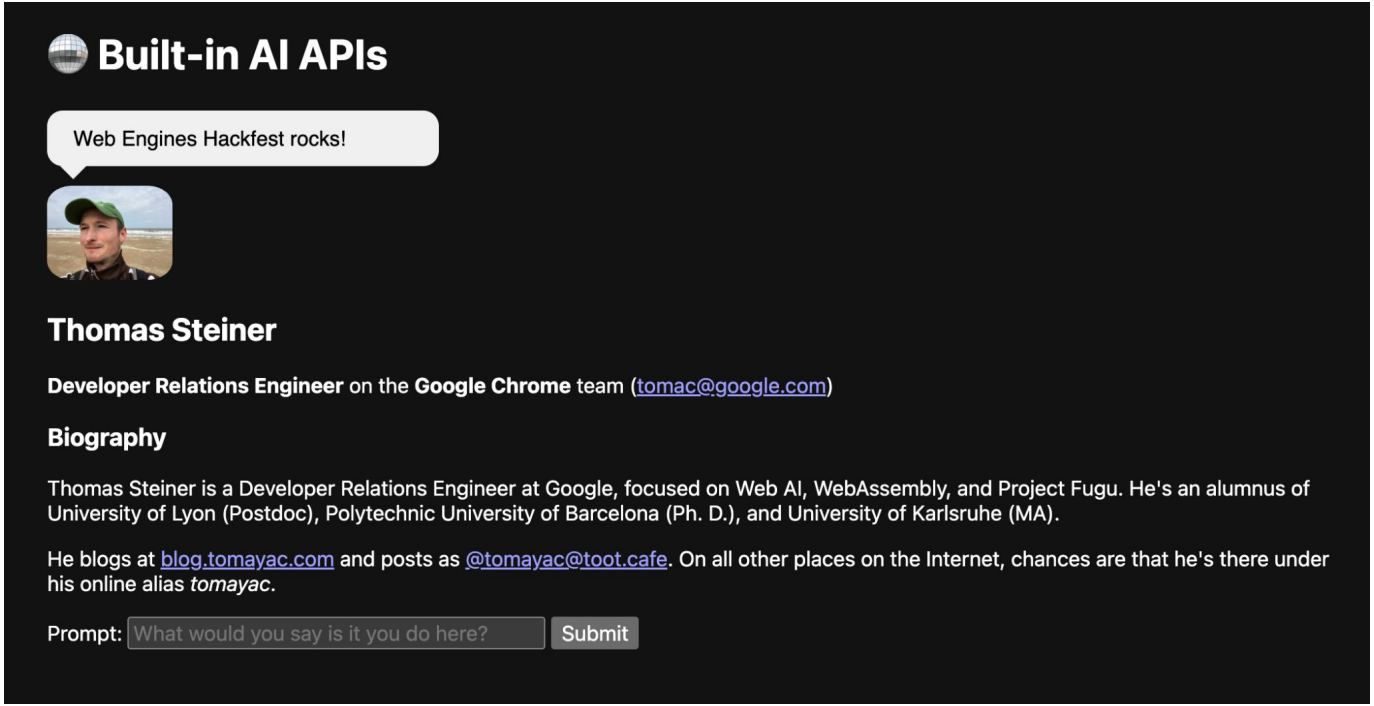
github.com/webmachinelearning/prompt-api/

Adding built-in AI to your web app

Possible AI use:

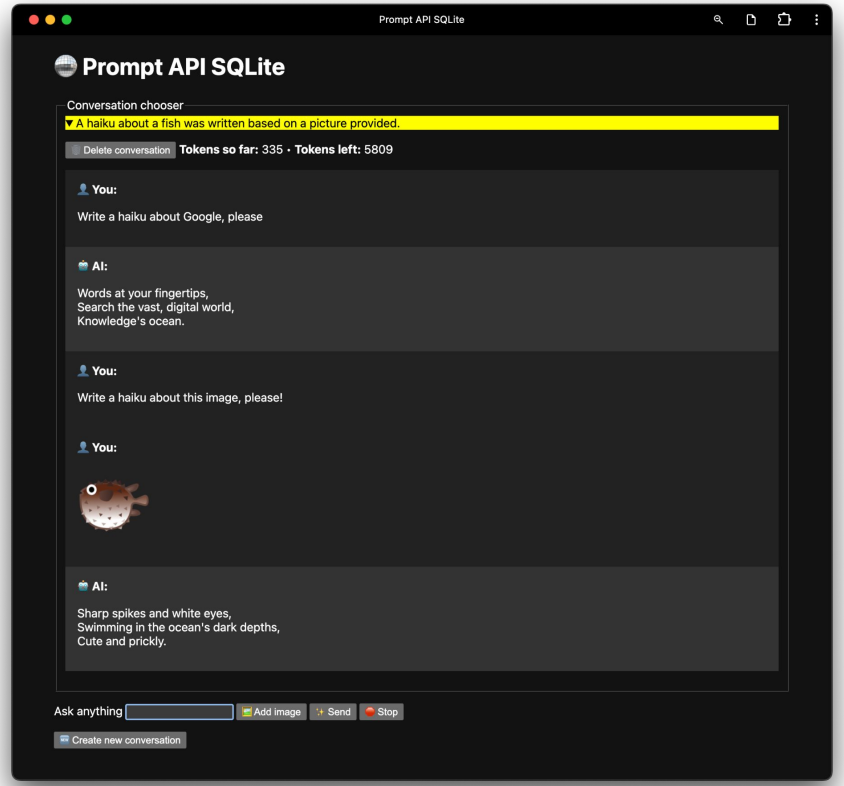
- *"List all my ideas where I hum."* → **Prompt API with audio input**
- *"Based on this headline, write an opening paragraph."* → **Writer API**
- *"How much wood do I need given this sketch?"* → **Prompt API w/ image input**
- *"Where can I buy a t-shirt with this design?"* → **Prompt API w/ image input**
- *"Rewrite this rough idea more formally."* → **Rewriter API**
- *"Share this idea translated to Japanese with my friend 英治."* → **Translator API**
- *"Summarize all my ideas from yesterday afternoon."* → **Summarizer API**

Live demo of the APIs



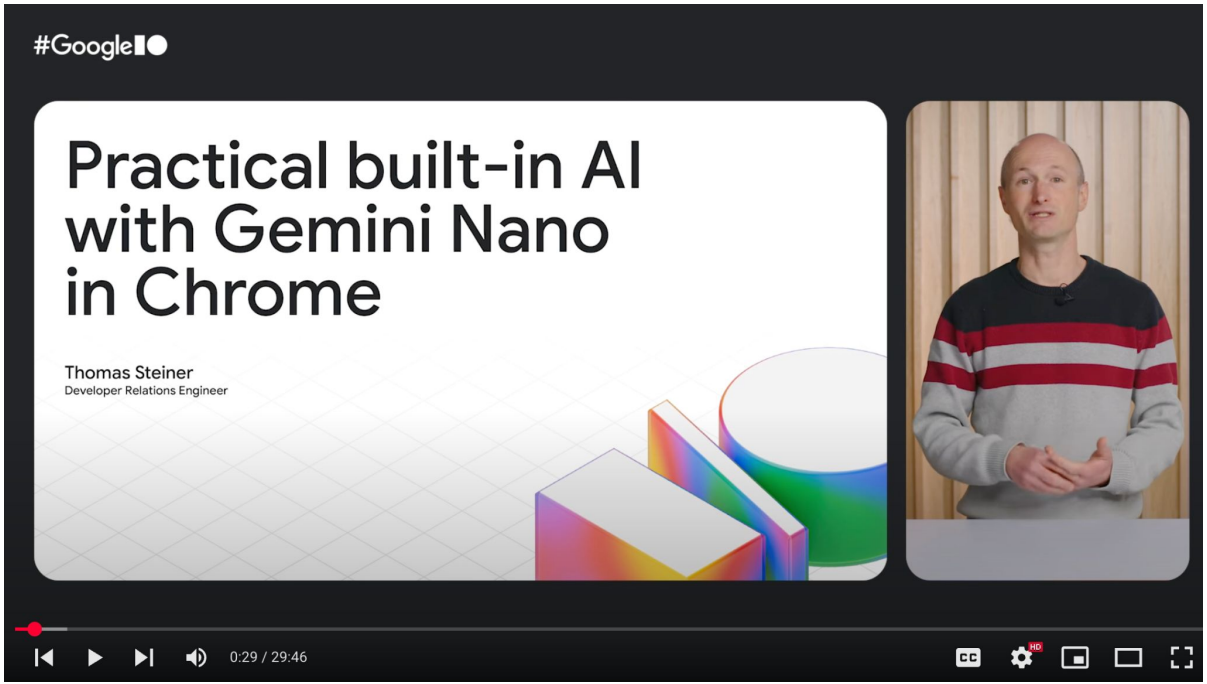
web-engines-hackfest-rocks.glitch.me/

Live demo of the APIs (cont’d)



tomayac.github.io/prompt-api-sqlite

Example partner implementations

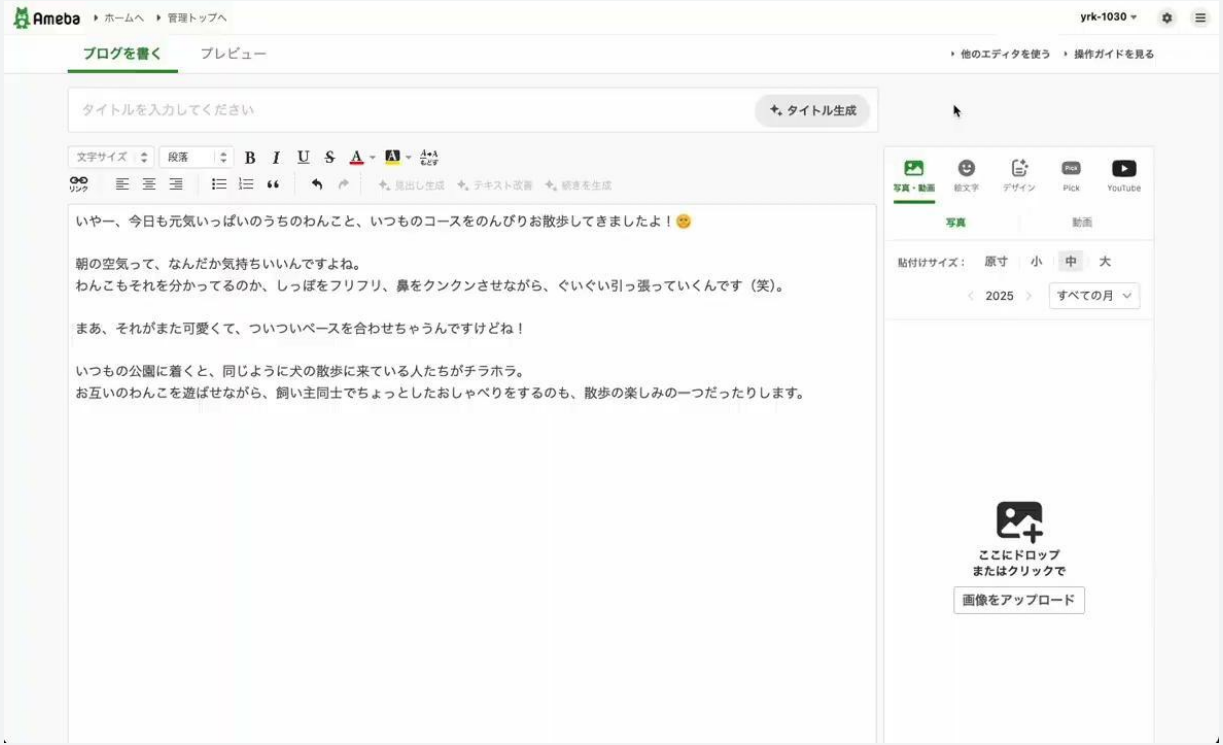


youtu.be/CjpZCWYrSxM

CyberAgent

Blogger helper function extension for blog writers powered by the **Prompt API for Chrome Extensions.**

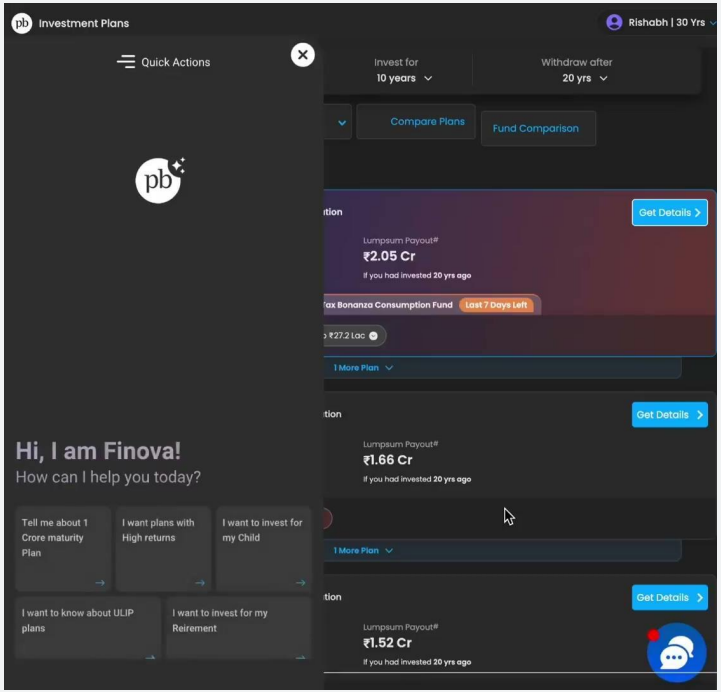
- Generating blog titles and headings.
- Improving and editing sentences.
- Drafting subsequent paragraphs.



policybazaar.com®

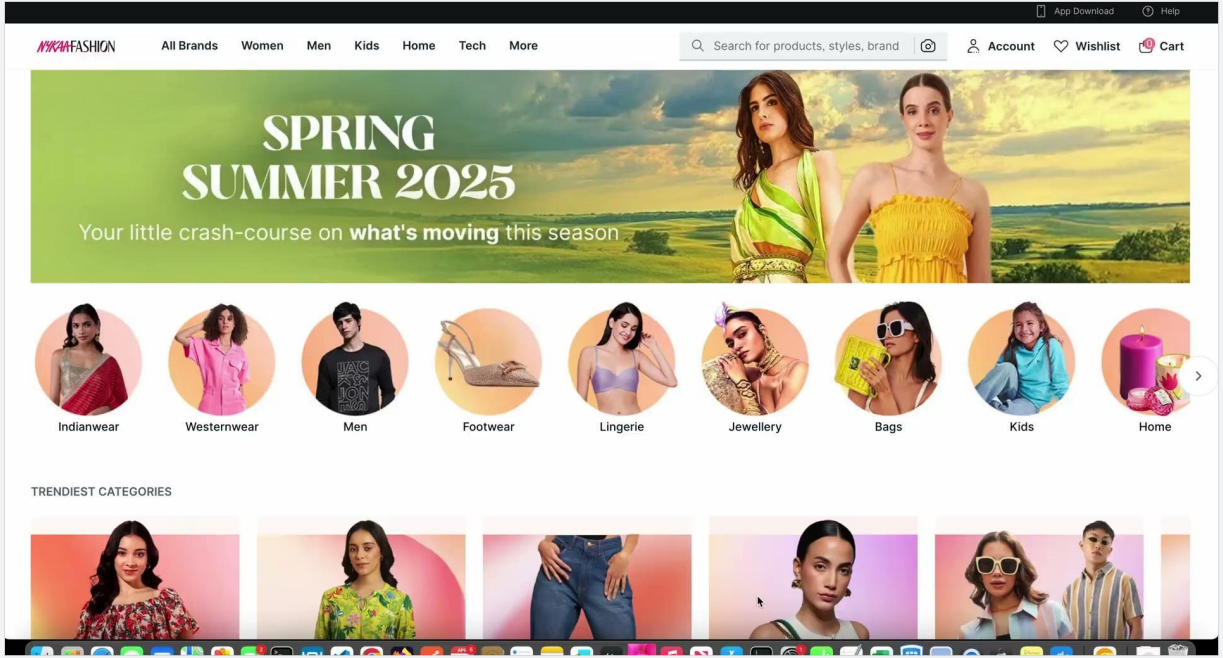
HAR FAMILY HOGI INSURED

Policybazaar used the **Language Detector** API to detect the user's input language and the **Translator API** to translate messages to and from English. This enables efficient and accessible insurance assistance to users in their native language.



NYKAA FASHION

Nykaa, India's leading beauty and lifestyle destination, tested the **Prompt API with image input** to let users run visual search sessions to find fitting outfits.

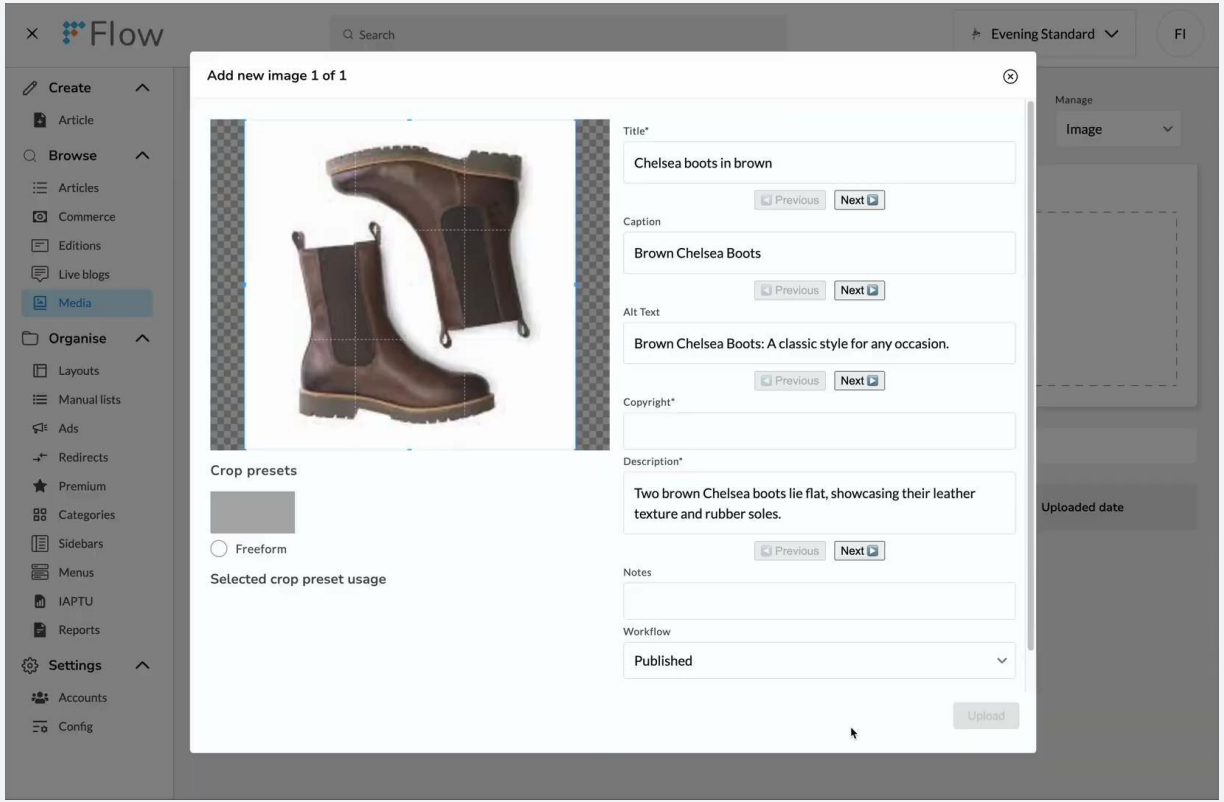


CHEFKOCH

Chefkoch streamlined the recipe publishing process for their users by integrating the **Prompt API with image input.** The API functions as client-side OCR, converting handwritten or photographed recipes into structured JSON output that can then be edited on the Chefkoch site.

BrightSites

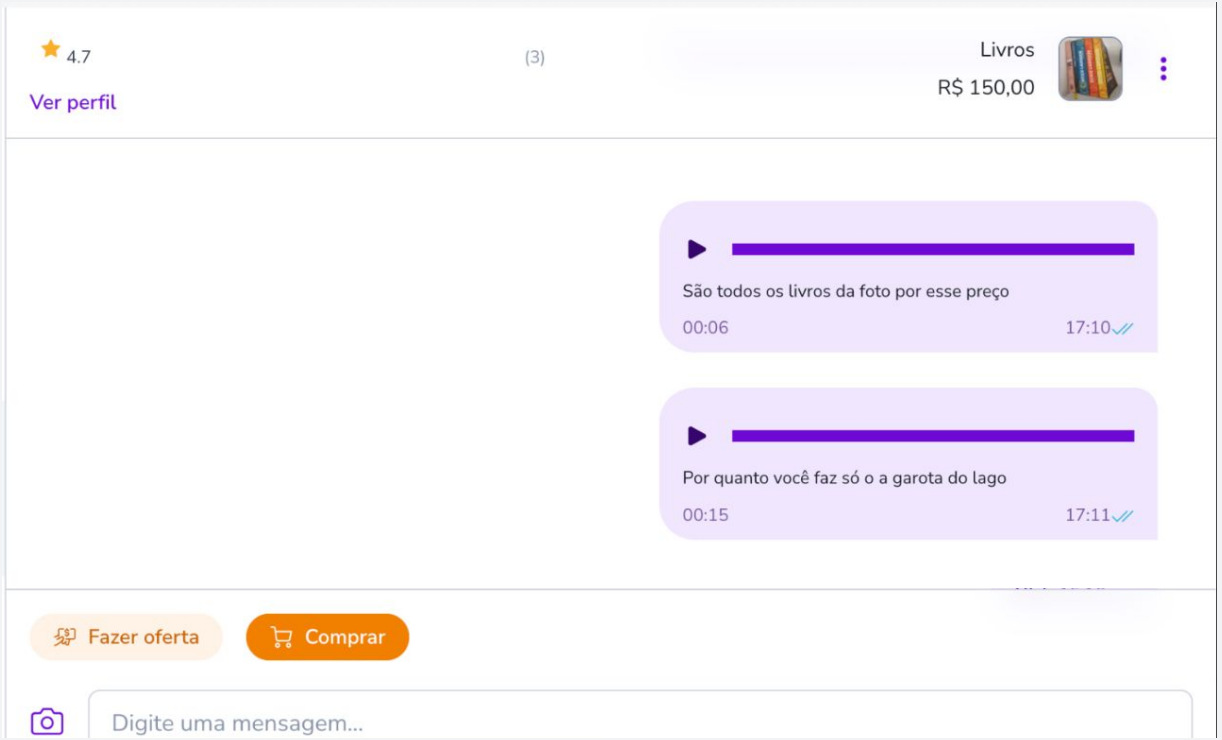
BrightSites' integration of the **Prompt API with image input** empowers journalists to automatically generate titles, captions, alt text, and descriptions directly from images within their CMS.



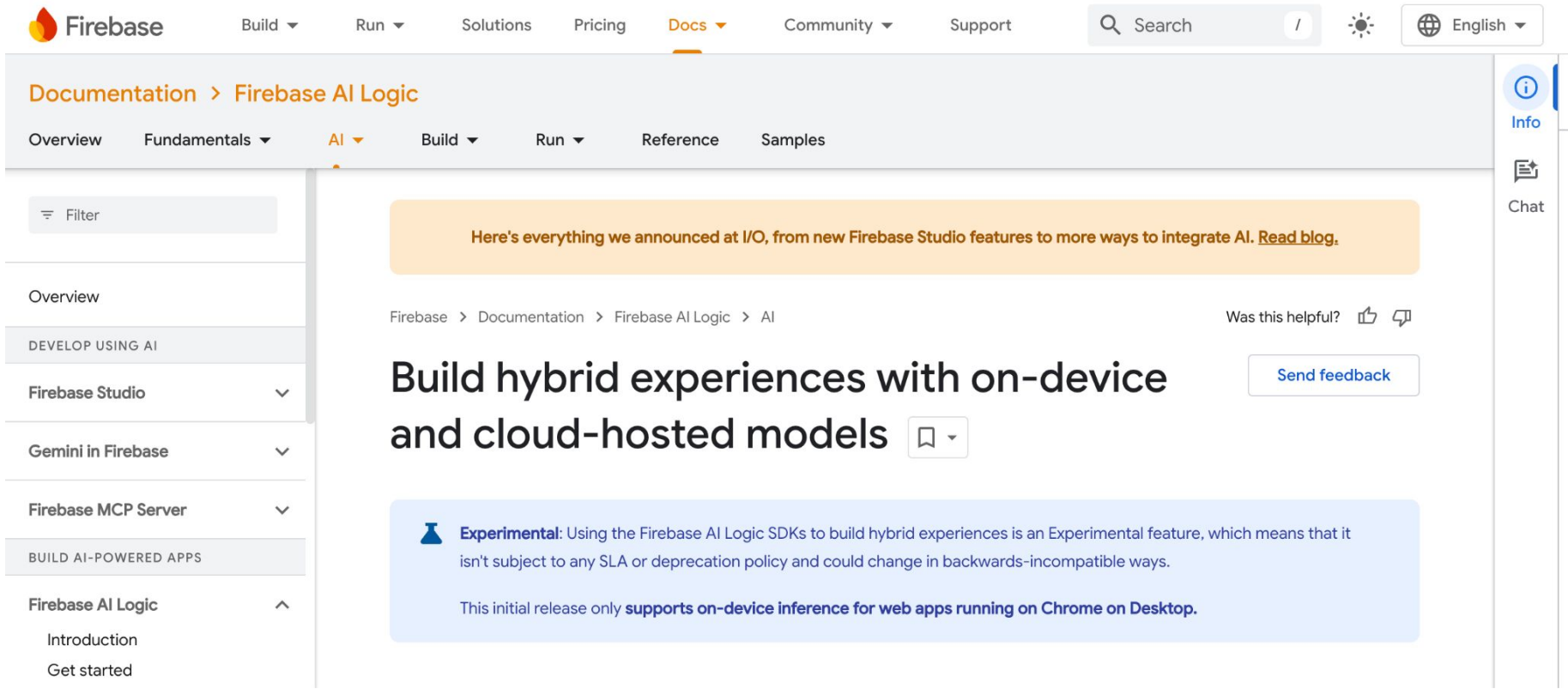
grupo OLX

olx zap VivaReal

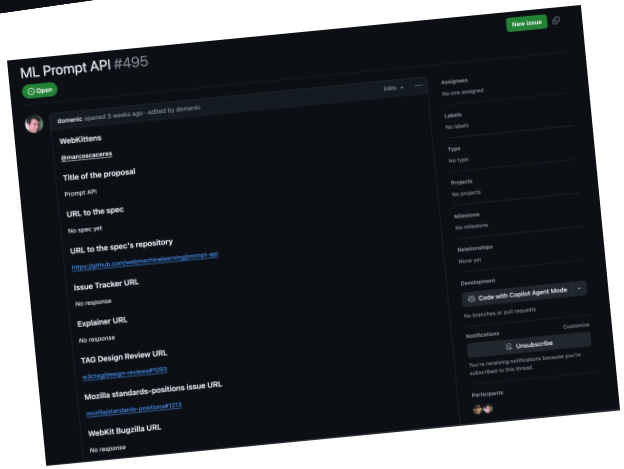
OLX used the **Prompt API with audio input** to transcribe audio messages, enhancing accessibility and enabling them to be read in more places and less time than the original audio.

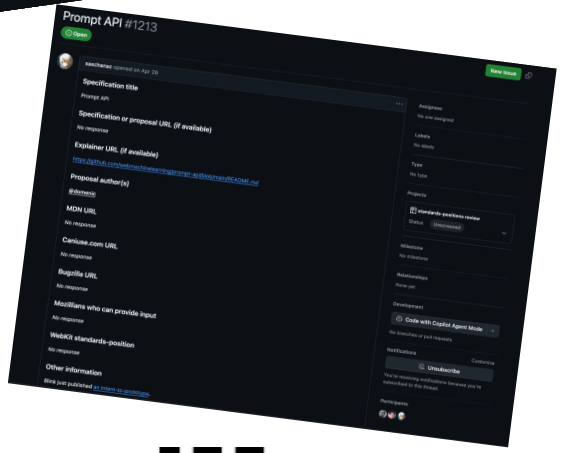


Firebase AI Logic



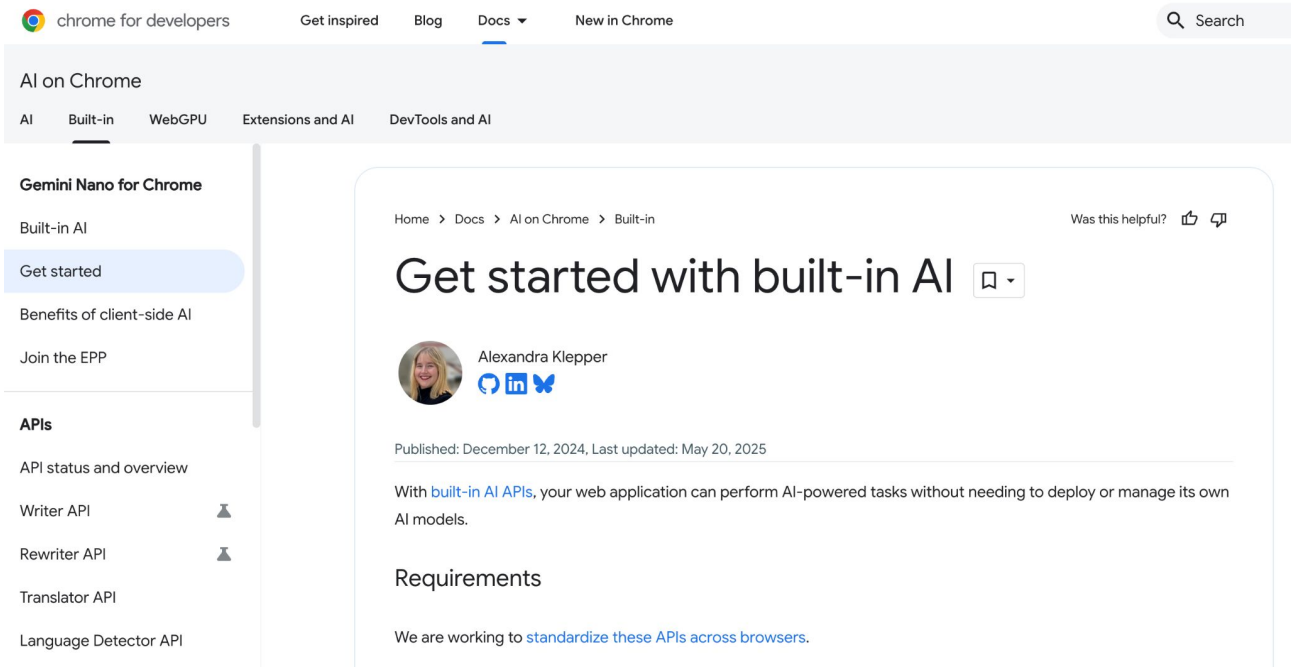
Seeking feedback





Mozilla

Learn more about Chrome’s implementation



Thank you

Thomas Steiner
tomac@google.com