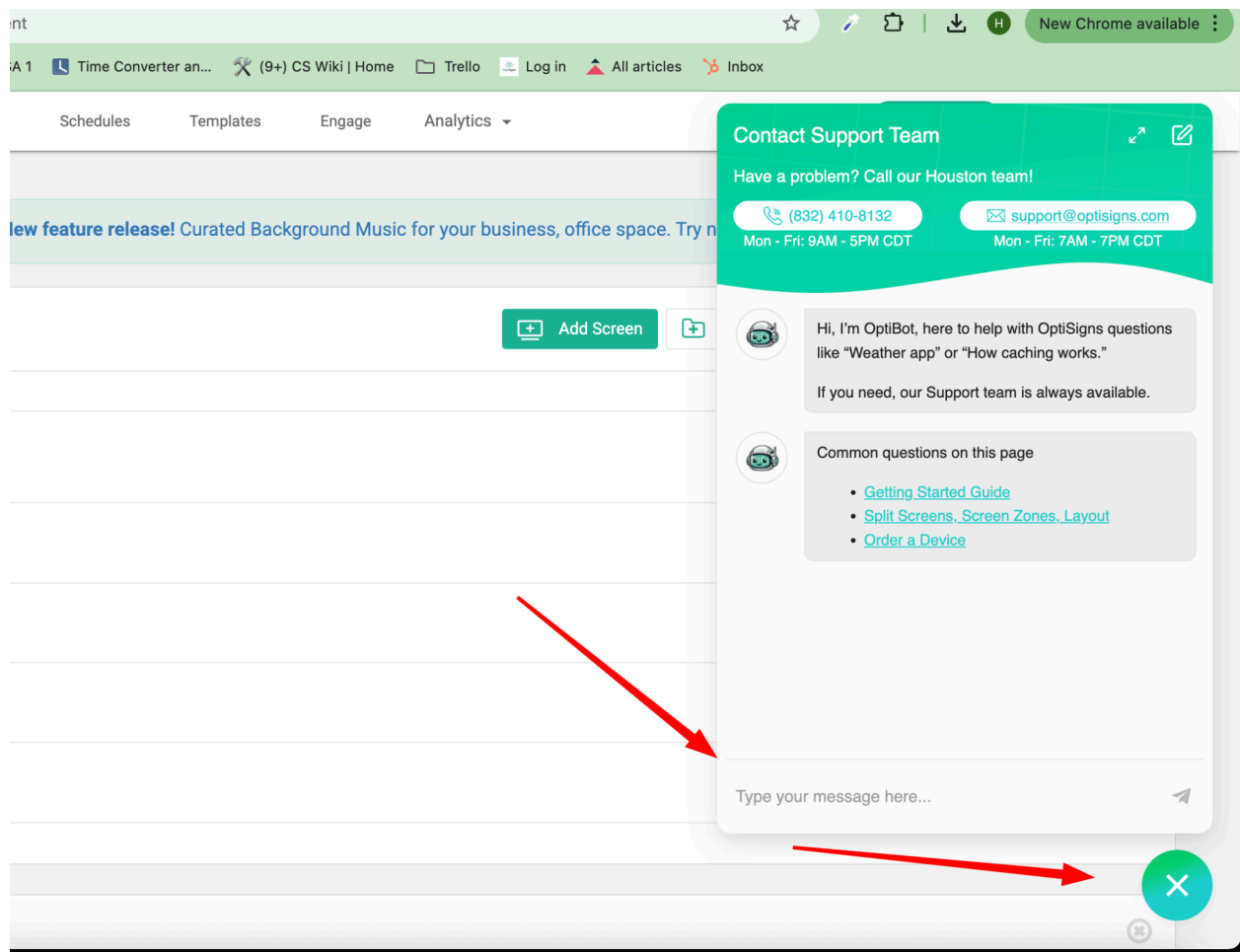


OptiSigns – OptiBot Mini-Clone Take-Home Test

— ship it in ~ 10 focused hours. Everything else is extra credit.

0 . Warm-up (15 min)

1. Create a free **optisigns.com** trial account and chat with **OptiBot** so you know what you're cloning.
2. Open a free **platform.openai.com** account.



1 . Scrape ⇒ Markdown (~3 h)

Goal: prove you can ingest messy web content and normalize it.

- Pull ≥ 30 articles from support.optisigns.com.
- Convert each article to clean **Markdown**.
- Save each file as `<slug>.md` (or whatever scheme you like).
- Preserve relative links, code blocks, and headings. Remove nav/ads.

Hint: you can use [Zendesk API to read the article](#).

2 . Build Assistant & Programmatically Load Vector Store (~2 h)

API upload is mandatory—no UI drag-and-drop here.

Create the Assistant – Easiest way is to use [OpenAI Playground UI](#).

System prompt (verbatim):

You are OptiBot, the customer-support bot for OptiSigns.com.

- Tone: helpful, factual, concise.
- Only answer using the uploaded docs.
- Max 5 bullet points; else link to the doc.
- Cite up to 3 "Article URL:" lines per reply.

1. **Via Python script**, upload Markdown files to OpenAI Vector Store files via OpenAI API.

- [Upload files to OpenAI](#)
- [Attach files to Vector Stores](#)
- Chunking strategy is up to you; just explain it in the README.
- Log how many files and chunks were embedded.

2. **Quick sanity check** – Jump into the Playground, attach the Assistant, and ask: “How do I add a YouTube video?”

Take a screenshot showing a correct answer *with citations*.

You can learn more about the overall [OpenAI Agent here](#).

3 . Deploy Scraper as Daily Job (~2 h)

- Wrap your scraper-uploader in `main.py`.
- **Dockerize** ([Dockerfile](#)).
- Schedule it to run **once per day** on [DigitalOcean Platform](#)
- Job must:
 1. Re-scrape.
 2. Detects new/updated articles (hash, `Last-Modified`, etc.).
 3. Upload only the delta.

Log counts: **added**, **updated**, **skipped**. Provide a link to job logs or last run artefact.

Deliverables

Item	Must Have
GitHub repo	Please DO NOT name your github repo “optisigns” something, just give it cryptic name so future candidate not easy to find when search optisigns Clear commits; no hard-coded keys (use <code>.env.sample</code>).
Dockerfile	<code>docker run -e OPENAI_API_KEY=... main.py</code> runs once and exits 0.
README (≤ 1 page)	• setup• how to run locally• link to daily job logs• screenshot of Playground answer
Screenshot	Assistant correctly answers sample questions with cited URLs.

Grading (70 pts pass bar)

Area	Pts
Scrape & clean quality	25
API-based vector-store upload works	20
Daily job deployment & logs	15
Code clarity + README	10
<i>(Bonus tests)</i>	+5

Set up 1h Project Reviews:

We will discuss

- Your overall concept understanding of the project
- Your approach, solution
- How do you learn something new like this if you haven’t learned it before
- Your thoughts, suggestions on how OptiBots can be improved, what potential challenges you think we will be facing