AIBurn: Real API Interrogation Addition

Created by	B Brad Wood
Created time	@November 16, 2025 12:51 AM
Category	How It Works Planning
Last edited by	B Brad Wood
Last updated time	@November 16, 2025 12:57 AM
PRD?	☐

📸 Screenshot/Share Features:

1. Share on X (Twitter)

What happens:

- Pre-fills tweet with their actual data:

> I just checked my AI token costs 🤯
>
> Currently spending: $1,247/mo
> Could save: $847/mo by switching to Claude Sonnet
>
> Check yours: aiburn.howstud.io

- Opens Twitter in new tab
- One-click to post
- Instant virality driver

2. Download Screenshot

What it generates:
A beautiful 1200×630px image (perfect for social media) with:

Top Section:

- "AI Token Cost Analysis" headline
- Purple-to-blue gradient background

Current Spending (Left Box):

- "Currently Spending"
- $X,XXX (huge text)
- "per month"

Potential Savings (Right Box - Green):

- "Potential Savings"
- $X,XXX saved
- "Switch to [Model Name]"

Token Usage (Middle):

- Input tokens count
- Output tokens count
- Model(s) used

Footer:

- "aiburn.howstud.io" branding
- "Free AI Token Cost Calculator"

File name: `ai-cost-report-2025-11-16.png`

3. Why This Works:

For Users:

- ✅ Keep personal records (compare month-to-month)
- ✅ Share impressive savings with team/boss

- ✅ Professional-looking report for presentations

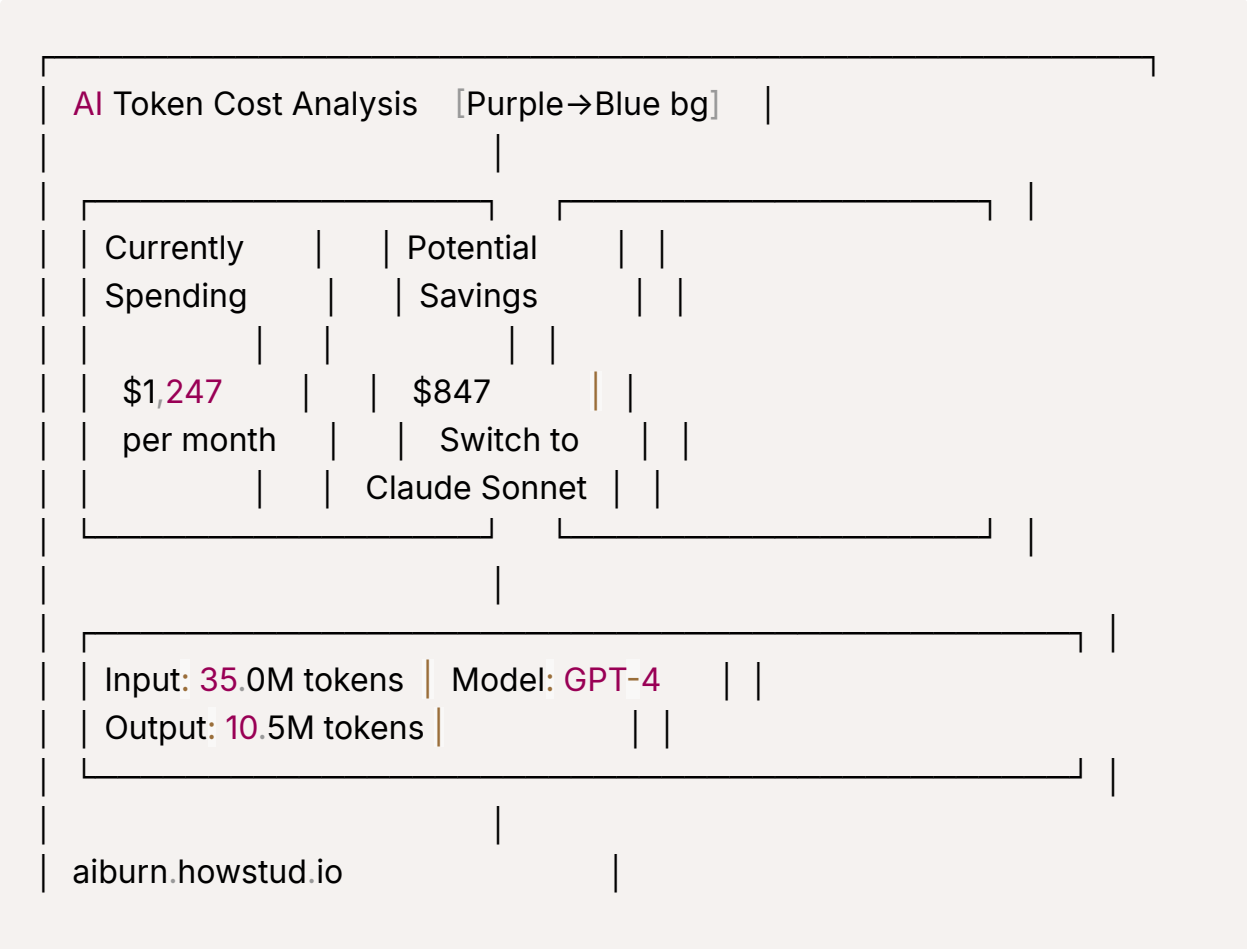
For Virality:

- ✅ Branded image = free advertising when shared
- ✅ Shows real numbers = social proof
- ✅ Eye-catching gradient design = stops scrolls
- ✅ Clear CTA (aiburn.howstud.io)

For Conversions:

- ✅ Users share → their followers see savings → visit tool
- ✅ Every share has your URL embedded
- ✅ Creates FOMO ("I should check mine too")

🎨 The Screenshot Looks Like:



```
| Free AI Token Cost Calculator |
└──────────────────────────────┘
```

How It Works Technically:

1. **HTML5 Canvas API** - Draws the image in browser
2. **No external libraries** - Pure JavaScript
3. **Instant generation** - No backend needed
4. **Auto-downloads** - Saves as PNG to their device
5. **High resolution** - 1200×630 (Twitter/LinkedIn optimal size)

Now We Need: Backend Proxy for API

Let me create the Vercel serverless function to handle the OpenAI API calls (avoid CORS):

File structure:

```
/api
  /usage.js  ← Serverless function
/src
  /App.jsx   ← Our React app
```