Front End Developer Intern Challenge

Fun with GPT-3

GPT-3 is a powerful AI model created by OpenAI. It can process plain text prompts and produce outputs that are hard to distinguish from human writing. Check out some examples of what it can do! GPT-3 can be accessed through a public API that includes a generous amount of free credits.

The Challenge

You will write an app that sends plain text prompts to the OpenAI API and displays the results in a list.

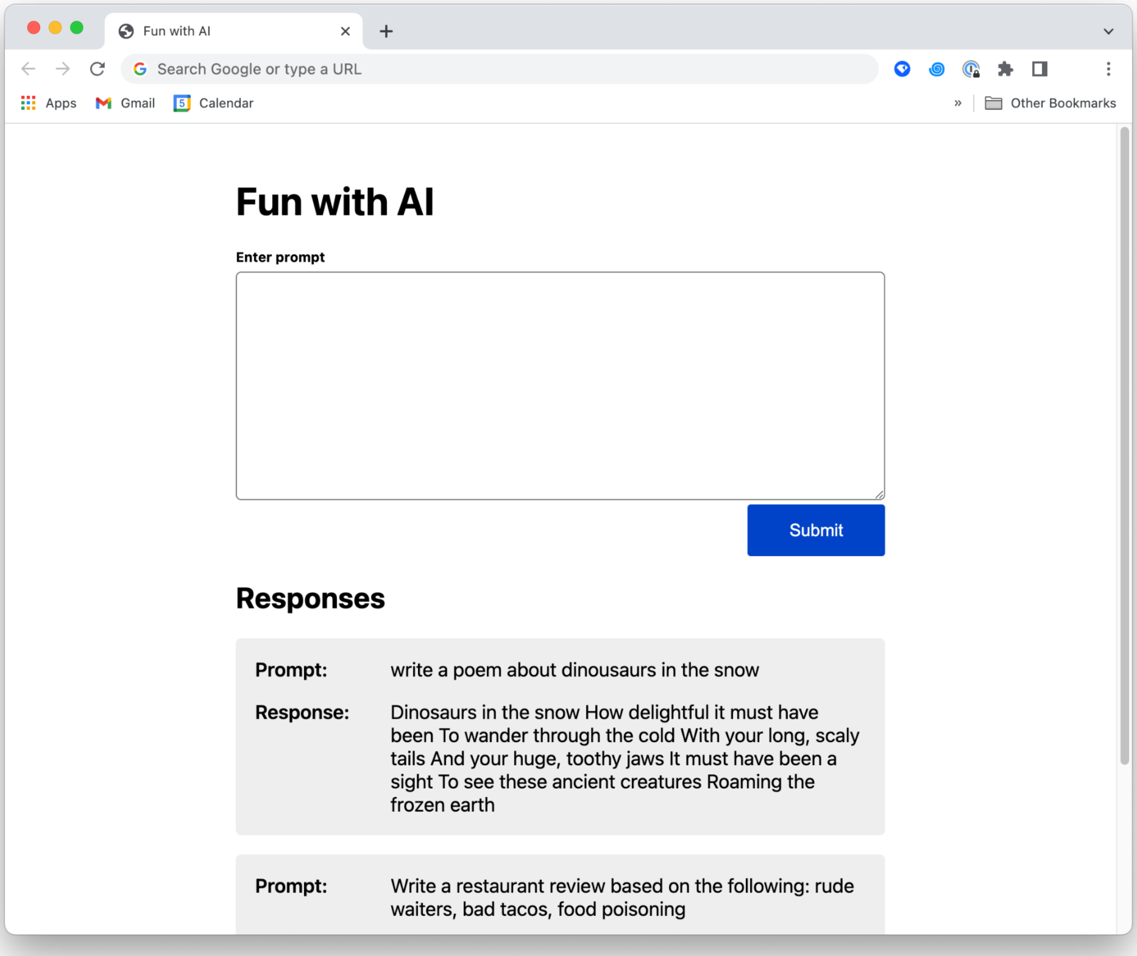
We'd like your app to have a simple-to-use interface that includes the following:

- A form for entering text prompts
- Submitting the form sends the prompt to the OpenAI API
- Results are displayed in a list, sorted from newest to oldest. Each result should include the original prompt and a response from the API.

Technical requirements

1. Results should come from OpenAI's completions API, for which you'll need a free API key (no credit card required). Detailed signup instructions are included below.
    - We've provided screenshots below of demo apps we built using the OpenAI API.
    - We recommend using the "**text-curie-001**" AI engine which is a good balance between speed, cost, and accuracy (example code below)
    - You are free to use any front end framework/component library you like (or none at all!)
2. Each result should include at least the original prompt you entered and the response from the API.
3. Responses should be stored in order of newest to oldest.
4. The HTML that ends up being served client-side should be accessible and semantic (MDN reference)

Example screenshot



Fun with AI

Enter prompt

Responses

Prompt: write a poem about dinousaurs in the snow

Response: Dinosaurs in the snow How delightful it must have been To wander through the cold With your long, scaly tails And your huge, toothy jaws It must have been a sight To see these ancient creatures Roaming the frozen earth

Prompt: Write a restaurant review based on the following: rude waiters, bad tacos, food poisoning

Extras

There is a lot to be improved on here, you can polish the required features by crafting a nicer design, or improve the app by adding new features! Choose something that you feel best showcases your passion and skills.

If you need inspiration, here are examples of what you can work on. If you work on these ideas, **we recommend choosing only one or two**.

- Save responses if the user leaves or reloads the page
- Let the user choose the AI engine from a select box
- Add some presets for the user to quickly send a good prompt

- Make the app more specific to a single purpose. See the OpenAI Examples and prompt design docs for inspiration.

Submission

Please submit your application via “Apply Now” and make sure you include:

- A link to your hosted code so we can test it (Free hosting available via: Cloudflare Pages, CodeSandbox, Github pages, Netlify and Heroku)
  - Quick start guides for each hosting site have been linked above
- A link to your Github repository containing the code
- Any other notes you'd like us to consider alongside the page

---

OpenAI sign up instructions

1. Go to https://beta.openai.com/signup
2. Enter your email address and password
3. Verify your email address
4. Verify your phone number by entering the code that is sent to you via SMS
5. Once logged in, go to https://beta.openai.com/account/api-keys to get your secret API key

Once you have your secret API key you can send requests from your application. Please note that you have $18.00 of free credits - consider using a less expensive model such as “**text-curie-001**” to ensure that you don’t exceed the limit.

Important: You should never expose API keys on the client side in production code because anyone on the internet will have access to your secret key and can make requests with your account. It is however acceptable for the purpose of this application and testing since it won’t be widely available and we don’t expect you to have a backend server.

Also note that you cannot commit your secret API key to your git repository. OpenAI will disable your API key if you do this. You should use environment variables or a key manager to avoid committing your secret API key. Check the documentation for your hosting provider for how to do this.

Example JS code for sending requests

```
const data = {
 prompt: "Write a poem about a dog wearing skis",
 temperature: 0.5,
 max_tokens: 64,
 top_p: 1.0,
 frequency_penalty: 0.0,
 presence_penalty: 0.0,
};

fetch("https://api.openai.com/v1/engines/text-curie-001/completions", {
 method: "POST",
 headers: {
   "Content-Type": "application/json",
   Authorization: `Bearer ${process.env.OPENAI_SECRET}`,
 },
 body: JSON.stringify(data),
});
```