

Prompt design {source code}

1/ Design Prompt

```
1. const response = await openai.createImage({
2.   prompt: input,
3.   n: 1,
4.   size: "512x512",
5. });
```

2/ entire file : **generateImage.js**

```
1. import { Configuration, OpenAIApi } from "openai";
2.
3. const configuration = new Configuration({
4.   apiKey: process.env.REACT_APP_OPENAI_API_KEY,
5. });
6. const openai = new OpenAIApi(configuration);
7.
8. export default async function (req, res) {
9.   if (!configuration.apiKey) {
10.    res.status(500).json({
11.      error: {
12.        message:
13.          "OpenAI API key not configured, please follow
instructions in README.md",
14.      },
15.    });
16.    return;
17.  }
18.
19.   const input = req.body.input;
20.   if (!input || input === "") {
21.    res.status(400).json({
22.      error: {
23.        message: "Please enter a valid input",
24.      },
25.    });
26.    return;
27.  }
28.
29.   try {
30.    const response = await openai.createImage({
31.      prompt: input,
32.      n: 1,
33.      size: "512x512",
34.    });
35.    console.log(response.data.data[0].url);
36.    const image_url = "";
37.    res.status(200).json({ image_url });
38.  } catch (e) {
39.    console.error(error);
40.    // Consider adjusting the error handling logic for your use
    case
41.    if (error.response) {
42.      console.error(error.response.status, error.response.data);
43.      res.status(error.response.status).json(error.response.data);
44.    } else {
```

```
45.         console.error(`Error with OpenAI API request: $
    {error.message}`);
46.         res.status(500).json({
47.             error: {
48.                 message: error.message,
49.             },
50.         });
51.     }
52. }
53. }
```