

Welcome to the [Creatopy AI](https://creatopy.com/ai) Challenge! In this interview task, we'll evaluate your skills in creating a web application for generating template-based banner ads.

Your goal is to build a user-friendly web application that enables users to effortlessly generate banner ads based on predefined templates. The application should provide a simple and intuitive interface, allowing users to choose from different banner types, generate images, and customise the content of their ads.

## Requirements

Start by creating a new application using the [StackBlitz](https://stackblitz.com/) templates.

The application should be built using [React](https://react.dev/), [TypeScript](https://www.typescriptlang.org/) and [Styled Components](https://styled-components.com/). You can use any other libraries or tools you want, but make sure to justify your choices.

The application should provide three banner templates: Instagram Post (500 x 500px), Twitter Header (1500 x 500px), and Story (1080 x 1920px). Users should be able to select a template and generate a banner ad based on it.

The templates should have the following elements predefined:

* **Text:** The text of the ad. The text should have a title and a description of different sizes.
* **Images:** The images of the ad.
* **Call to Action:** The call to action of the ad. This element should be a clickable button.

All elements in the template should have a position, size and default colour. The application should allow users to change the content of the elements through the generator, but not their position or size.

## API

You should use the [OpenAI API](https://beta.openai.com/docs/api-reference) to generate text and images for your ad. To access the API you’ll need to have an API Key.

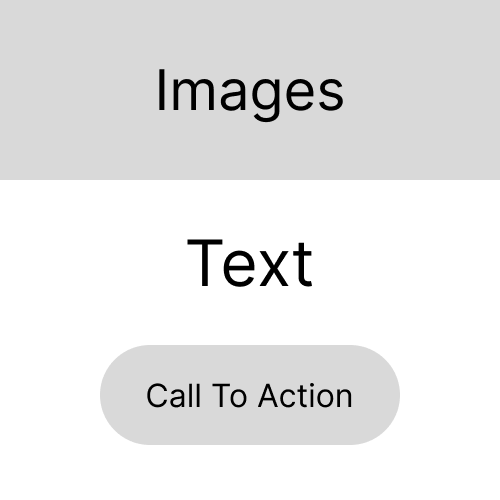
**If you don’t have an API key please contact us (**[**razvan.geangu@creatopy.com**](mailto:razvan.geangu@creatopy.com)**) and we will share with you a unique key for this challenge.**

The application should provide a text input field to describe the ad. When the user clicks the "Generate" button, the application should send a request to the API and display the data into the selected template. The application should allow the user to generate a new ad by clicking the "Generate" button again.

The API should return the following data:

* **Text:** The text of the ad. The text should be generated based on the description provided by the user (title and description).
* **Images:** The images of the ad. The images should be generated based on the description provided by the user.
* **Call to Action:** The call to action of the ad. The call to action should be generated based on the description provided by the user.

Here’s an example of a basic template where the elements are arranged in order:



## Bonus

If you want to go the extra mile, you can implement some of the following features:

* Integrate a file uploader to allow users to upload an image of an ad and generate a new ad based on it. The application should send the image to the API and get the text, colours, images and call to action from the API response.
* Integrate branding elements to enhance the overall look and feel of the ad. You can use the [Brandfetch API](https://brandfetch.com/developers) to retrieve brand assets based on a brand name. When the user describes their ad using a brand name, the application should send a request to the API and display the brand assets into the selected template such as the logo and colours.
* Integrate a colour picker to allow users to choose their preferred colours for the ad.

## Deliverables

Once you're done with the challenge, you need to send us the link to your repository and a short description of your solution. We would like to see the source code and the application hosted and running.

The description should include the following information:

* The features you implemented and why you chose them.
* Challenges you faced and how you overcame them.
* What you would do differently if you had more time.

*Optionally you can also send us a link to a live demo of your application.*

Good luck with the challenge! We look forward to seeing your creative and technically proficient solution.