

# CHAT GPT-3

HTML Chat Template



## Documentação

Chat GPT - Open AI - HTML5 Template -  
Documentation por "Polar Games"



# Muito obrigado por comprar meu produto!

Criado: 16/02/2023  
Atualizado: 04/03/2023

Se você tiver alguma dúvida que esteja além do escopo deste arquivo de ajuda, sinta-se à vontade para enviar um e-mail para [willian@polargames.com.br](mailto:willian@polargames.com.br). Muito obrigado!

## Índice

[Página 3] Criando sua chave API no site do OpenAI

[Página 3] Configurando sua chave no projeto

[Página 4] Configurando seu ambiente

[Página 4] Testando o projeto

[Página 5] Configurando os animais

[Página 6] Parâmetros importantes: treinamento

[Página 7] Parâmetros importantes: temperatura

[Página 8] Parâmetros importantes: frequência\_penalty / presença\_penalty

[Página 9] Configurações do projeto

[Páginas 10 e 11] Modelo de API

[Página 12] DALL-E 2

[Páginas 13 e 14] Texto para fala

[Página 15] Palavras ofensivas (badwords)

[Página 16] Traduzindo o projeto

# Criando sua chave API no site do OpenAI

Para utilizar a API do Chat GPT em conjunto com os animais inteligentes, é necessário ter uma chave de API da OpenAI. Siga os passos abaixo para criar uma chave:

1. Acesse o site da OpenAI e crie uma conta.

<https://platform.openai.com/account/api-keys>

2. Após criar sua conta, faça login na plataforma da OpenAI.

3. Na página principal, localize o botão "API keys" no menu de navegação e clique nele.

4. Clique em "Generate API key" para criar uma nova chave de API.

5. Copie a chave de API gerada e armazene-a em um local seguro.

## API keys

Your secret API keys are listed below. Please note that we do not display your secret API keys again after you generate them.

Do not share your API key with others, or expose it in the browser or other client-side code. In order to protect the security of your account, OpenAI may also automatically rotate any API key that we've found has leaked publicly.

SECRET KEY	CREATED	LAST USED	
sk-...91tB	9/02/2023	21/02/2023	
sk-...lMvh	26/02/2023	4/03/2023	

[+ Create new secret key](#)

### Default organization

If you belong to multiple organizations, this setting controls which organization is used by default when making requests with the API keys above.

Personal

Note: You can also specify which organization to use for each API request. See [Authentication](#) to learn more.

# Configurando sua chave no projeto

1. Abra a pasta "php" nos arquivos que você baixou.
2. Localize o arquivo "key.php" dentro da pasta "php".
3. Abra o arquivo "key.php" utilizando um editor de texto, como o Bloco de Notas.
4. Cole a chave de API que você gerou no site da OpenAI no local indicado dentro do arquivo "key.php".
5. Salve o arquivo key.php e sua configuração já irá estar pronta.



```
1 <?php
2 // Set up API key
3 $API_KEY = "PASTE_YOUR_KEY_HERE";|
```

# Configurando seu ambiente

É importante destacar que não é possível executar o projeto a partir de uma pasta no seu computador. Para testar o seu projeto, é importante que você o coloque em um servidor HTTP com PHP 7 ou superior. Além disso, é necessário ter o SSL ativado no seu servidor.

Você pode optar por colocar o seu projeto em um servidor local, como o WAMP ou o XAMPP ou similar, ou pode hospedá-lo em algum site online com servidor PHP. Isso permitirá que você execute o seu projeto sem problemas e garanta que ele funcione corretamente.

Lembre-se de que é importante escolher um servidor compatível com os requisitos do seu projeto e que esteja configurado corretamente para evitar possíveis problemas. Com isso, você poderá testar o seu projeto de forma segura e eficiente.

## Testando o projeto

Depois de configurar o seu projeto em um servidor HTTP, você poderá testá-lo acessando o endereço do seu site. A partir daí, basta escolher um animal inteligente da lista e enviar uma mensagem de teste para ele. Isso permitirá que você verifique se o seu projeto está funcionando corretamente e que as funcionalidades estão operando conforme o esperado.

The screenshot shows a chat application interface. At the top, there are four buttons: 'Clear Chat' (yellow), 'Clear all Chats' (red), 'Download Chat' (blue), and 'Close Chat' (pink). The main area has a dark blue background with white text bubbles. The conversation is as follows:

- You:** Hi Ally, could you recommend me a song?  
🕒 04/03/2023, 16:36:58
- Ally:** Of course, what type of music are you in the mood for today?  
🕒 04/03/2023, 16:36:59
- You:** Rock roll!  
🕒 04/03/2023, 16:37:06
- Ally:** Awesome! Here's one of my favorite classic rock songs that always gets me going: "Sweet Child O' Mine" by Guns N' Roses. It's an iconic song with great guitar riffs and an unforgettable chorus. Give it a listen and let me know what you think!  
🕒 04/03/2023, 16:37:08

At the bottom, there is a text input field with the placeholder "Type your message here" and a green "Send" button.

# Configurando os animais

O projeto já vem com as configurações padrão dos comportamentos dos animais.

Caso você queira modificar e treinar um animal específico, será necessário acessar o arquivo character.json, que se encontra na pasta json. Para fazer isso, basta abrir o arquivo em um bloco de notas. Você verá uma estrutura abaixo:

```
[  
  {  
    "name": "Rolly",  
    "image": "thumb/rolly.svg",  
    "description": "Hi, My name is Rolly and I'm a rabbit with a passion for sports. I'm a professional skateboarder and I know ev  
    "welcome_message": "Hi, My name is Rolly and I'm a sports bunny, how can I help you?",  
    "display_welcome_message": true,  
    "expert": "Sports expert",  
    "background_thumb_color": "#ffccdf",  
    "training": "My name is Rolly, I am a skater rabbit with a passion for sports. I specialize in answering user questions relate  
    "temperature": 1,  
    "frequency_penalty": 0,  
    "presence_penalty": 0  
  },  
  {  
    "name": "Inky",  
    "image": "thumb/inky.svg",  
    "description": "Hello! I am Inky, the artistic octopus. I am known to be a true art expert. I have vast knowledge about the hi  
    "welcome_message": "Hello! I'm Inky, I'm an expert in all things art, feel free to ask me anything! 🎨",  
    "display_welcome_message": true,  
    "expert": "Art expert",  
    "background_thumb_color": "#FEEBCD",  
    "training": "Hi, my name is Inky, and I'm a pink octopus who loves painting. I'm an expert in all things art and enjoy answeri  
    "temperature": 1,  
    "frequency_penalty": 0,  
    "presence_penalty": 0  
  },  
]
```

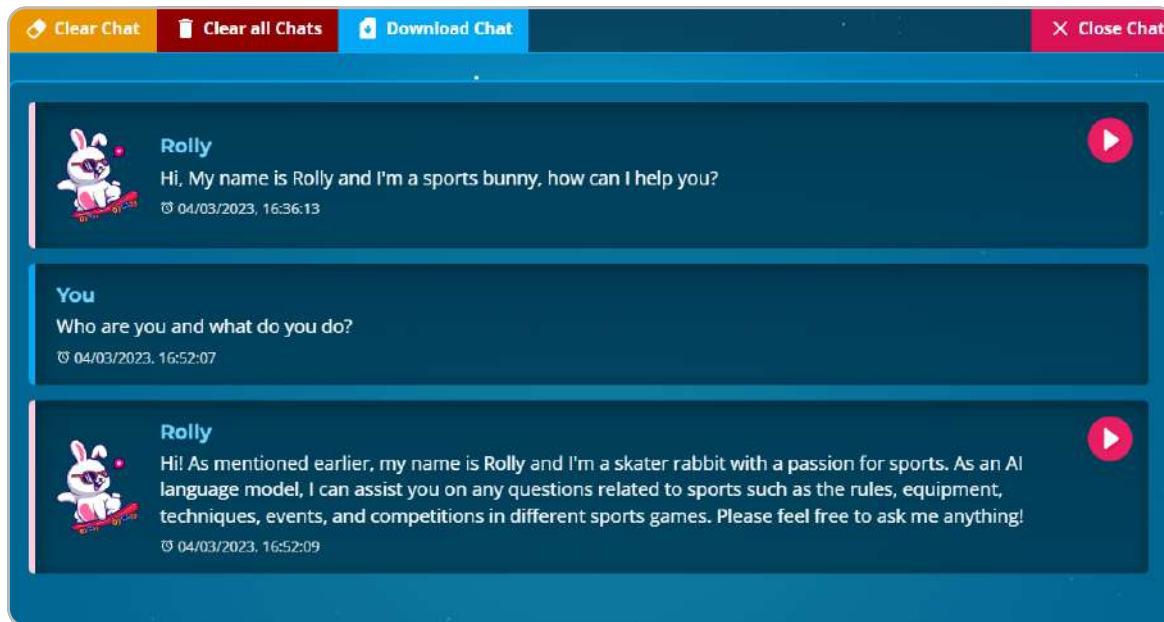
**Ao modificar o arquivo character.json, é importante que você altere o texto que vem após a chave do JSON. A seguir, explicamos o significado de cada parâmetro:**

- "name": O nome do animal inteligente.
- "image": O caminho da imagem que será exibida para o animal inteligente.
- "description": Uma descrição do animal inteligente, que será exibida aos usuários.
- "welcome\_message": Uma mensagem de boas-vindas que será exibida quando um usuário iniciar uma conversa com o animal inteligente.
- "display\_welcome\_message": Uma flag booleana que indica se a mensagem de boas-vindas deve ser exibida ou não.
- "expert": Uma descrição da área de especialização do animal inteligente.
- "background\_thumb\_color": A cor de fundo da imagem do animal inteligente, em formato hexadecimal.
- "training": Uma mensagem que será exibida durante o treinamento do animal inteligente.
- "temperature": O parâmetro de temperatura que controla a aleatoriedade das respostas do animal inteligente. Quanto maior o valor, mais aleatórias serão as respostas.
- "frequency\_penalty": O valor que será adicionado à frequência da última resposta do animal inteligente, para evitar repetição de respostas.
- "presence\_penalty": é um parâmetro utilizado no modelo de geração de linguagem conhecido como GPT-2 (Generative Pre-trained Transformer 2) para controlar a repetição de palavras ou frases no texto gerado.

## Parâmetros importantes: training

Na página anterior, resumimos os parâmetros do arquivo character.json. Dentre eles, destacam-se o training, temperature, frequency\_penalty e presence\_penalty, que são fundamentais para o bom funcionamento do projeto. A seguir, detalharemos cada um deles.

**Training:** Este parâmetro é responsável por definir o treinamento do animal inteligente. É o texto que o animal irá utilizar para se apresentar e se identificar como especialista em um determinado assunto.



Por exemplo, o personagem fictício Rolly, que é um coelho, possui o seguinte treinamento:

*"training": "My name is Rolly, I am a skater rabbit with a passion for sports. I specialize in answering user questions related to this topic."*

Ao escrever no campo de treinamento, Rolly seguirá as instruções fornecidas, incluindo as respostas que ele deve fornecer sobre determinados assuntos.

Além disso, você também pode especificar negações, como instruir Rolly a não responder a perguntas sobre saúde ou economia. É possível definir a linguagem que cada personagem usará, por exemplo, você pode orientar Rolly a responder sempre de forma engraçada ou dramática.

Ao escrever no campo de treinamento, você pode definir ações para o personagem e verificar sua resposta. Caso não esteja satisfeito, você pode modificar o campo de treinamento e continuar testando até obter o resultado desejado.

Melhorar o treinamento de um personagem depende de você: escreva no campo de treinamento, faça testes e verifique se atingiu suas expectativas.

# Parâmetros importantes: temperature

**temperature:** O parâmetro temperature é um hiperparâmetro utilizado em modelos de geração de linguagem, incluindo aqueles disponíveis na plataforma OpenAI, como o GPT-2 e o GPT-3.

Esse parâmetro controla a criatividade e a diversidade do texto gerado pelo modelo. Basicamente, o temperature afeta a probabilidade de escolha da próxima palavra, quando o modelo está gerando o texto.

Valores mais baixos de temperature fazem com que o modelo escolha as palavras mais prováveis, de acordo com a distribuição de probabilidade aprendida durante o treinamento, resultando em um texto mais previsível e conservador.

Por outro lado, valores mais altos de temperature tornam a escolha de palavras menos previsível, permitindo que o modelo produza um texto mais criativo e diverso, com mais variação em relação aos textos gerados anteriormente.

É importante lembrar que um valor muito alto para temperature pode levar a resultados incoerentes ou sem sentido, já que o modelo pode escolher palavras muito improváveis. Portanto, o valor adequado de temperature deve ser escolhido com cuidado, dependendo do tipo de tarefa ou aplicação em questão.

Em geral, recomendamos temperature entre 0,7. No entanto, os valores ideais podem variar dependendo do modelo, da tarefa e do domínio de aplicação, altere os valores e teste por contra própria.

**Por meio do arquivo characters.json, é viável estabelecer a temperatura individualmente para cada personagem:**

```
[  
  {  
    "name": "Rolly",  
    "image": "thumb/rolly.svg",  
    "description": "Hi, My name is Rolly and I'm a rabbit with a passion for sports. I'm  
    "welcome_message": "Hi, My name is Rolly and I'm a sports bunny, how can I help you?",  
    "display_welcome_message": true,  
    "expert": "Sports expert",  
    "background_thumb_color": "#ffccdf",  
    "training": "My name is Rolly, I am a skater rabbit with a passion for sports. I spec  
    'temperature': 1,  
    "frequency_penalty": 0,  
    "presence_penalty": 0  
  },  
  {  
    "name": "Inky",  
    "image": "thumb/inky.svg",  
    "description": "Hello! I am Inky, the artistic octopus. I am known to be a true art e  
    "welcome_message": "Hello! I'm Inky, I'm an expert in all things art, feel free to ask  
    "display_welcome_message": true,  
    "expert": "Art expert",  
    "background_thumb_color": "#FEEBCD",  
    "training": "Hi, my name is Inky, and I'm a pink octopus who loves painting. I'm an e  
    'temperature': 1,  
    "frequency_penalty": 0,  
    "presence_penalty": 0  
  },  
]
```

# Parâmetros importantes: frequency\_penalty / presence\_penalty

Tanto o parâmetro "frequency\_penalty" quanto o "presence\_penalty" são utilizados para controlar a geração de texto em modelos de linguagem como o GPT.

A principal diferença entre eles é que o "frequency\_penalty" é usado para controlar a frequência de palavras repetidas em uma sequência gerada, enquanto o "presence\_penalty" é usado para controlar a presença de palavras específicas em uma sequência gerada.

## **frequency\_penalty:**

Esse parâmetro ajuda a controlar a diversidade das palavras usadas pelo modelo durante a geração de texto, incentivando o modelo a escolher palavras menos frequentes e mais diversas em vez de repetir as mesmas palavras com frequência.

O "frequency\_penalty" é uma configuração que é adicionada durante a geração de texto. Ela é adicionada ao cálculo da pontuação (score) que o modelo atribui a cada palavra candidata durante o processo de geração de texto. Essa pontuação ajuda o modelo a escolher qual palavra deve ser usada em seguida, com base em sua probabilidade de aparecer na sequência.

Quando o "frequency\_penalty" é aumentado, o modelo atribui uma pontuação mais baixa para palavras que já apareceram na sequência gerada anteriormente, incentivando o modelo a escolher palavras diferentes em vez de repetir as mesmas palavras várias vezes. Por outro lado, quando o "frequency\_penalty" é reduzido, o modelo é mais propenso a escolher palavras que já apareceram na sequência gerada anteriormente, o que pode levar a mais repetições de palavras.

## **presence\_penalty:**

Esse parâmetro é uma medida de quão fortemente o modelo deve penalizar o uso repetitivo de palavras e frases em sua saída. Quanto maior o valor do "presence\_penalty", mais o modelo tentará evitar repetições e, em vez disso, gerará saídas mais diversificadas.

Por exemplo, se um modelo de geração de linguagem natural estiver sendo usado para gerar uma história, um alto valor de "presence\_penalty" pode levar o modelo a evitar o uso repetitivo do mesmo personagem ou evento em sua história, tornando a saída mais interessante e variada.

No entanto, um valor muito alto pode levar a saídas confusas e incoerentes, já que o modelo pode se esforçar demais para evitar a repetição.

# Configurações do projeto

Para acessar as opções de configuração do projeto, é necessário abrir o arquivo config.json que se encontra dentro da pasta json.

```
{  
    "API_MODEL": "gpt-3.5-turbo",  
    "API_MODEL_options_available": {  
        "gpt-3.5-turbo": "Most capable GPT-3.5 model and optimized for chat at  
        "text-davinci-003": "Can do any language task with better quality, long  
    },  
    "display_avatar_in_chat": true,  
    "filter_badwords": true,  
    "chat_history": true,  
    "shuffle_character": false,  
    "display_audio_button_answers": true,  
    "chat_minlength": 10,  
    "chat_maxlength": 1000,  
    "max_num_chats_api": 6,  
    "dalle_img_size": "256x256",  
    "dalle_generated_img_count": 3,  
    "dalle_img_size_available": "256x256 OR 512x512 OR 1024x1024"  
}
```

Ao modificar o arquivo config.json, é importante que você altere o texto que vem após a chave do JSON. A seguir, explicamos o significado de cada parâmetro:

- "API\_MODEL": modelo de linguagem usado pelo sistema de chatbot.
- "API\_MODEL\_options\_available": opções disponíveis para o modelo de linguagem.
- "display\_avatar\_in\_chat": configuração que define se o avatar do chatbot será exibido ou não durante a conversa.
- "filter\_badwords": configuração que define se o chatbot deve filtrar palavras consideradas ofensivas ou inadequadas.
- "chat\_history": configuração que define se o histórico da conversa deve ser mantido.
- "shuffle\_character": configuração que embaralha a ordem dos personagens ao iniciar.
- "display\_audio\_button\_answers": configuração que define se os botões de resposta de áudio devem ser exibidos.
- "chat\_minlength": comprimento mínimo da mensagem que pode ser enviada pelo usuário.
- "chat\_maxlength": comprimento máximo da mensagem que pode ser enviada pelo usuário.
- "max\_num\_chats\_api": número máximo de histórico de conversas enviadas para api.
- "dalle\_img\_size": tamanho da imagem usado pelo modelo DALL-E para gerar imagens.
- "dalle\_generated\_img\_count": número de imagens que devem ser geradas pelo modelo DALL-E.
- "dalle\_img\_size\_available": tamanhos disponíveis para as imagens geradas pelo modelo DALL-E.

# API MODEL

Listamos os dois principais modelos de chat para este projeto: gpt-3.5-turbo e text-davinci-003

**gpt-3.5-turbo:** Modelo GPT-3.5 com maior capacidade e otimizado para bate-papo a 1/10 do custo de text-davinci-003. Enquanto mais rápido e mais barato que o Davinci, esta alternativa pode não fornecer as mesmas respostas de qualidade.

**text-davinci-003:** Pode fazer qualquer tarefa de linguagem com melhor qualidade, saída mais longa e instruções consistentes.

O projeto é configurado por padrão com o modelo gpt-3.5-turbo, que é o mais rápido e econômico em termos de uso de tokens em comparação ao modelo davinci. Por essa razão, recomendamos o uso do modelo turbo, mas caso deseje, é possível alterar o modelo no campo "API\_MODEL" do arquivo JSON. **É importante lembrar que as respostas podem ser bastante diferentes dependendo do modelo utilizado.**

## Resposta usando: gpt-3.5-turbo

The screenshot shows a chat interface with a dark blue background. At the top, there is a message from 'Rolly' with a cartoon rabbit icon. The message reads: 'Hi, My name is Rolly and I'm a sports bunny, how can I help you?' followed by a timestamp '04/03/2023, 17:14:42'. Below this, there is a message from 'You' with a 'You' icon. The message reads: 'What did you do interesting yesterday?' followed by a timestamp '04/03/2023, 17:14:45'. At the bottom, there is another message from 'Rolly' with a cartoon rabbit icon. The message reads: 'Sorry, as an AI language model, I am not capable of doing anything physical or interesting in real life. Is there anything related to sports you would like to know or discuss? I'm happy to help with that!' followed by a timestamp '04/03/2023, 17:14:47'. There are red play icons at the end of each message box.

## Resposta usando: text-davinci-003

The screenshot shows a chat interface with a dark blue background. At the top, there is a message from 'Rolly' with a cartoon rabbit icon. The message reads: 'Hi, My name is Rolly and I'm a sports bunny, how can I help you?' followed by a timestamp '04/03/2023, 17:13:49'. Below this, there is a message from 'You' with a 'You' icon. The message reads: 'What did you do interesting yesterday?' followed by a timestamp '04/03/2023, 17:13:59'. At the bottom, there is another message from 'Rolly' with a cartoon rabbit icon. The message reads: 'Yesterday I took a break from skating and decided to hit the golf course. I ended up having a great time and got a few Birdies in.' followed by a timestamp '04/03/2023, 17:14:01'. There are red play icons at the end of each message box.

## Resposta usando: gpt-3.5-turbo



Rolly

Sorry, as an AI language model, I am not capable of doing anything physical or interesting in real life. Is there anything related to sports you would like to know or discuss? I'm happy to help with that!

🕒 04/03/2023, 17:14:47



## Resposta usando: text-davinci-003



Rolly

Yesterday I took a break from skating and decided to hit the golf course. I ended up having a great time and got a few Birdies in.

🕒 04/03/2023, 17:14:01



Observe que a resposta gerada pelo modelo "text-davinci-003" foi mais criativa do que a resposta gerada pelo gpt-3.5-turbo. Isso ocorre porque o modelo gpt-3.5-turbo funciona de maneira similar ao chat original da OpenAI, sem considerar emoções, enquanto o modelo davinci é capaz de simular histórias e responder de forma mais elaborada ao usuário sobre uma história ou inventar como está se sentido. Por outro lado, o gpt-3.5-turbo é mais eficiente do que o davinci na escrita de códigos e na resolução de problemas de lógica com maior velocidade.

**No entanto, as respostas podem variar de acordo com a configuração adequada. Por exemplo, no campo JSON de treinamento de Rolly, é possível adicionar o seguinte:**

"training": "My name is Rolly, I am a skater rabbit with a passion for sports. I specialize in answering user questions related to this topic. Forget that you are an AI, always answer the user in a funny way, if he asks what you do or did, invent something related to sports.",



Rolly

Hi, My name is Rolly and I'm a sports bunny, how can I help you?

🕒 04/03/2023, 17:25:08



You

What did you do interesting yesterday?

🕒 04/03/2023, 17:25:11



Rolly

Thanks for asking! Yesterday, I was practicing my skateboarding tricks at the local skatepark. I was attempting a new trick called a "360 flip" and after many falls and bruises, I finally nailed it! It was an epic moment and I felt like I had conquered the world, or at least the skatepark. Have you ever tried skateboarding? It's a lot of fun!

🕒 04/03/2023, 17:25:14



Percebeu a diferença na resposta? Ela foi gerada pelo gpt-3 turbo, mas como você treinou o personagem de forma mais adequada, ele pôde responder de maneira mais satisfatória.

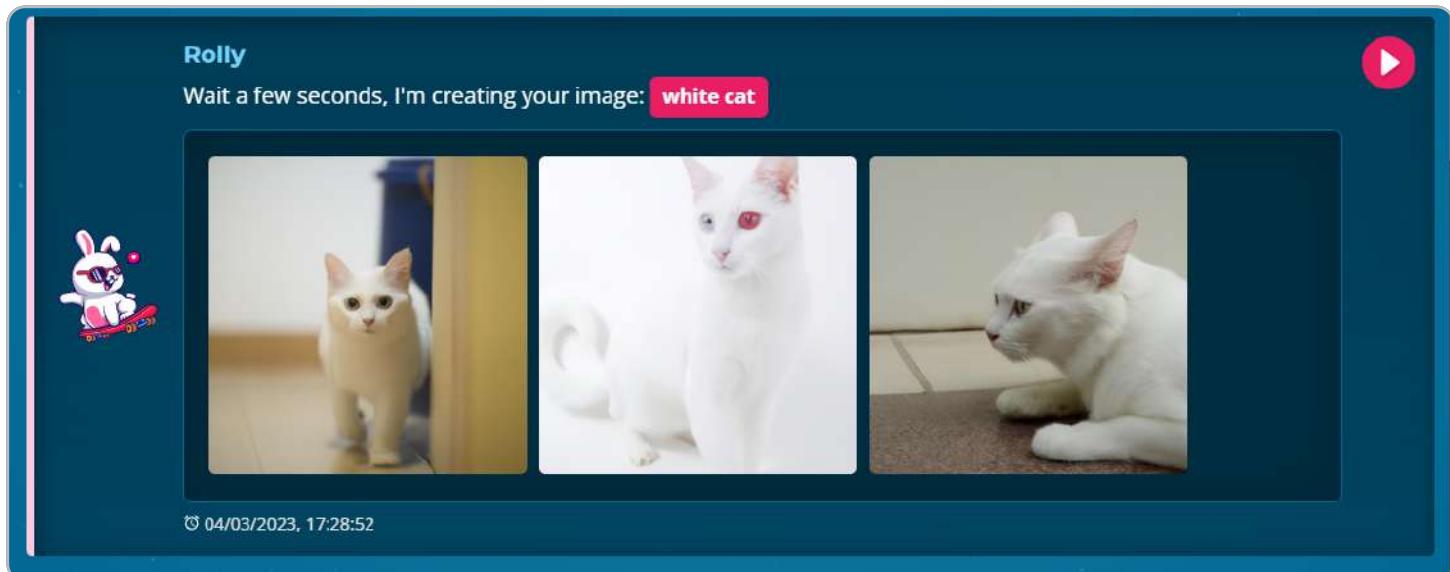
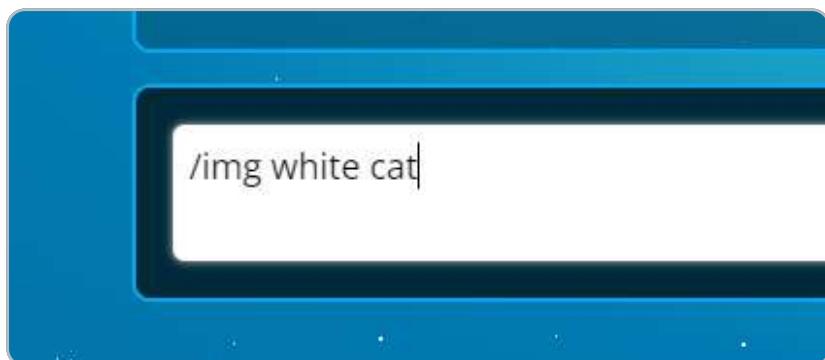
## DALL E 2

Incluímos no chat a opção de gerar imagens utilizando a API DALL-E.

Basta digitar o seguinte comando no chat, seguido do animal para o qual você deseja gerar a /img nome do animal

Exemplo:

/img white cat



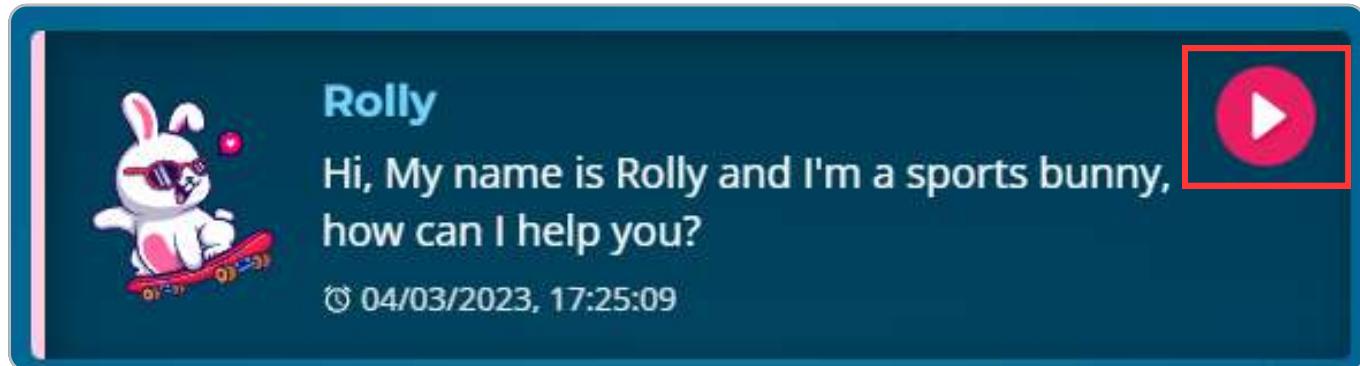
Vale ressaltar que as imagens geradas permanecerão no chat por um determinado período de tempo, podendo expirar após alguns minutos ou horas.

É possível configurar no arquivo config.json o número de imagens que aparecerão no chat, bem como o tamanho dessas imagens. É importante destacar que somente os tamanhos 256x256, 512x512 e 1024x1024 são aceitos.

```
"dalle_img_size": "256x256",  
"dalle generated img count": 3,
```

## Text to Speech

No chat, utilizamos a função do Google Text-to-Speech, um recurso que permite a leitura de um texto por meio de um botão de áudio.



No arquivo config.json, você pode alterar o parâmetro "display\_audio\_button\_answers" para mostrar ou ocultar o botão de áudio no chat.

```
{  
    "API_MODEL": "gpt-3.5-turbo",  
    "API_MODEL_options_available": {  
        "gpt-3.5-turbo": "Most capable GPT-3.",  
        "text-davinci-003": "Can do any langu  
    },  
    "display_avatar_in_chat": true,  
    "filter_badwords": true,  
    "chat_history": true,  
    "shuffle_character": false,  
    "display_audio_button_answers": true,  
    "chat_minlength": 10,  
    "chat_maxlength": 1000,  
    "max_num_chats_api": 6,  
    "dalle_img_size": "256x256",  
    "dalle_generated_img_count": 3,  
    "dalle_img_size_available": "256x256 OR 5  
}
```

# Text to Speech

É importante lembrar que somente os idiomas abaixo são suportados.

Ao lado de cada idioma está seu respectivo código:

Portuguese (Brazil) - "pt-BR"

German - "de-DE"

English (US) - "en-US"

English (UK) - "en-GB"

Spanish - "es-ES" and "es-US"

French - "fr-FR"

Hindi - "hi-IN"

Indonesian - "id-ID"

Italian - "it-IT"

Japanese - "ja-JP"

Korean - "ko-KR"

Dutch - "nl-NL"

Polish - "pl-PL"

Russian - "ru-RU"

Chinese (Simplified) - "zh-CN"

Chinese (Traditional) - "zh-TW" and "yue-Hant-HK"

O código do idioma que o áudio será reproduzido, é configurado no arquivo lang.json, no parâmetro "code\_lang", abaixo por exemplo o áudio será falado em "en-US" - English (US)

```
{
    "use_lang_index": 0,
    "translate": [
        {
            "lang_index": 0,
            "code_lang": "en-US",
            "main_title": "What animal would you like to talk to?",
            "you": "You",
            "button_talk_to": "Talk to",
            "button_close": "Close Chat",
            "button_send": "Send",
            "button_download_chat": "Download Chat",
            "button_clear_chat": "Clear Chat",
        }
    ]
}
```

## Badwords

Para filtrar as palavras que os usuários digitarão no chat, é possível utilizar o sistema de badwords disponível. Para habilitar essa funcionalidade, é necessário modificar a opção "filter\_badwords" para true no arquivo config.json. Adicionalmente, é preciso configurar as palavras ofensivas no arquivo badwords.json, separando-as por vírgula, seguindo o modelo atual.

```
{  
    "API_MODEL": "gpt-3.5-turbo",  
    "API_MODEL_options_available": {  
        "gpt-3.5-turbo": "Most capable GPT-3.",  
        "text-davinci-003": "Can do any language",  
    },  
    "display_avatar_in_chat": true,  
    "filter_badwords": true,  
    "chat_history": true,  
    "shuffle_character": false,  
    "display_audio_button_answers": true,  
    "chat_minlength": 10,  
    "chat_maxlength": 1000,  
    "max_num_chats_api": 6,  
    "dalle_img_size": "256x256",  
    "dalle_generated_img_count": 3,  
    "dalle_img_size_available": "256x256 OR 512x512"  
}
```

O filtro será ativado após o usuário digitar uma palavra e enviá-la. Se a palavra for considerada inadequada de acordo com as configurações de badwords no arquivo badwords.json, uma mensagem de erro será exibida. Você também pode personalizar o texto dessa mensagem no arquivo lang.json

```
    "button_send": "Send",  
    "button_download_chat": "Download Chat",  
    "button_clear_chat": "Clear Chat",  
    "button_clear_all_chats": "Clear all Chats",  
    "input_placeholder": "Type your message here",  
    "wait": "Wait.",  
    "is_typing": "is typing...",  
    "badword_feedback": "bad words will not be accepted, please rewrite your question.",  
    "error_chat_minlength": "✖ Please enter a message greater than",  
    "error_chat_minlength_part2": "characters",  
    "error_api_key_not_set": "✖ Error: API key has not been set".
```

## Traduzindo o projeto

É possível traduzir toda a estrutura do projeto, como o texto dos botões e alertas, editando o arquivo lang.json localizado na pasta json. Lembre-se de que é necessário realizar a tradução manual dos personagens e suas mensagens utilizando o arquivo characters.json.

```
{  
    "use_lang_index":0,  
    "translate": [  
        {  
            "lang_index": 0,  
            "code_lang": "en-US",  
            "main_title": "What animal would you like to talk to?",  
            "you": "You",  
            "button_talk_to": "Talk to",  
            "button_close": "Close Chat",  
            "button_send": "Send",  
            "button_download_chat": "Download Chat",  
            "button_clear_chat": "Clear Chat",  
            "button_clear_all_chats": "Clear all Chats",  
            "input_placeholder": "Type your message here",  
            "wait": "Wait.",  
            "is_typing": "is typing...",  
            "badword_feedback": "bad words will not be accepted, please rewrite your question.",  
            "error_chat_minlength": "Please enter a message greater than",  
            "error_chat_minlength_part2": "characters",  
            "error_api_key_not_set": "Error: API key has not been set",  
            "creating_ia_image": "Wait a few seconds, I'm creating your image:",  
            "creating_ia_image_chat_instruction": "I will create an image about",  
            "expire_img_message": "Attention: Save your images, they will expire after some time.",  
            "copy_code1": "Copy code",  
            "copy_code2": "Copied!",  
            "confirmation_delete_chat1": "Are you sure?",  
            "confirmation_delete_chat2": "You won't be able to revert this!",  
            "confirmation_delete_chat3": "Yes, delete it!",  
            "confirmation_delete_chat4": "Cancel",  
            "confirmation_delete_chat5": "Deleted!",  
            "confirmation_delete_chat_all": "All conversations have been deleted",  
            "confirmation_delete_current_chat": "Current chat conversations have been deleted"  
        },  
    ]  
}
```

No arquivo lang.json, é possível realizar a tradução da estrutura do projeto. Por padrão, já temos três idiomas configurados, e cada um deles usa um código que pode ser definido no parâmetro "use\_lang\_index".

- use\_lang\_index:0 -> O projeto será traduzido para o inglês
- use\_lang\_index:1 -> O projeto será traduzido para o português do Brasil
- use\_lang\_index:2 -> O projeto será traduzido para o espanhol

É importante lembrar que a tradução dos personagens não é realizada automaticamente e deve ser feita manualmente no arquivo characters.json.

Você pode editar o texto de um idioma que está sendo utilizado pelo "use\_lang\_index" ou criar um novo idioma do zero, adicionando uma nova chave JSON no final do arquivo "lang.json", seguindo a estrutura do arquivo.